electronics

Special Issue Reprint

Deep Learning in Image Processing and Pattern Recognition

Edited by
Yuji Iwahori, Aili Wang and Haibin Wu

mdpi.com/journal/electronics

MDPI

Deep Learning in Image Processing and Pattern Recognition

Deep Learning in Image Processing and Pattern Recognition

Guest Editors

Yuji Iwahori
Aili Wang
Haibin Wu

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Guest Editors

Yuji Iwahori
Department of Computer Science
Chubu University
Kasugai
Japan

Aili Wang
Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application
Harbin University of Science and Technology
Harbin
China

Haibin Wu
Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application
Harbin University of Science and Technology
Harbin
China

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Electronics* (ISSN 2079-9292), freely accessible at: www.mdpi.com/journal/electronics/special_issues/2XZYCHZVH5.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-4371-8 (Hbk)
ISBN 978-3-7258-4372-5 (PDF)
https://doi.org/10.3390/books978-3-7258-4372-5

© 2025 by the authors. Articles in this book are Open Access and distributed under the Creative Commons Attribution (CC BY) license. The book as a whole is distributed by MDPI under the terms and conditions of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) license (https://creativecommons.org/licenses/by-nc-nd/4.0/).

Contents

About the Editors . . . . . **ix**

Aili Wang, Haibin Wu and Yuji Iwahori
Deep Learning in Image Processing and Pattern Recognition
Reprinted from: *Electronics* **2025**, *14*, 1942, https://doi.org/10.3390/electronics14101942 . . . . . **1**

Qing-Dao-Er-Ji Ren, Lele Wang, Zerui Ma and Saheya Barintag
Offline Mongolian Handwriting Recognition Based on Data Augmentation and Improved ECA-Net
Reprinted from: *Electronics* **2024**, *13*, 835, https://doi.org/10.3390/electronics13050835 . . . . . **9**

Yi Jiang, Kexin Yang, Jinlin Zhu and Li Qin
YOLO-Rlepose: Improved YOLO Based on Swin Transformer and Rle-Oks Loss for Multi-Person Pose Estimation
Reprinted from: *Electronics* **2024**, *13*, 563, https://doi.org/10.3390/electronics13030563 . . . . . **27**

Mingji Yang, Xinbo Tong and Haisong Chen
Detection of Small Lesions on Grape Leaves Based on Improved YOLOv7
Reprinted from: *Electronics* **2024**, *13*, 464, https://doi.org/10.3390/electronics13020464 . . . . . **43**

Fengqi Yan, Shaokun Li, Zhiguo Zhou and Yonggang Shi
A Residual Network with Efficient Transformer for Lightweight Image Super-Resolution
Reprinted from: *Electronics* **2024**, *13*, 194, https://doi.org/10.3390/electronics13010194 . . . . . **65**

Xiaohong Qian, Lifeng Xie, Ning Ye, Renlong Le and Shengying Yang
Pixel-Level Degradation for Text Image Super-Resolution and Recognition
Reprinted from: *Electronics* **2023**, *12*, 4546, https://doi.org/10.3390/electronics12214546 . . . . . **82**

Yong Qin, Zhenye Pan and Chenhao Shao
Defect Detection Method of Phosphor in Glass Based on Improved YOLO5 Algorithm
Reprinted from: *Electronics* **2023**, *12*, 3917, https://doi.org/10.3390/electronics12183917 . . . . . **98**

Rimeh Daoudi, Aymen Mouelhi, Moez Bouchouicha, Eric Moreau and Mounir Sayadi
Fully Automatic Approach for Smoke Tracking Based on Deep Image Quality Enhancement and Adaptive Level Set Model
Reprinted from: *Electronics* **2023**, *12*, 3888, https://doi.org/10.3390/electronics12183888 . . . . . **110**

Xiaoyu Yu, Fuchao Li, Yan Liu and Aili Wang
A Multi-Stage Adaptive Copy-Paste Data Augmentation Algorithm Based on Model Training Preferences
Reprinted from: *Electronics* **2023**, *12*, 3695, https://doi.org/10.3390/electronics12173695 . . . . . **127**

Yingluo Song, Aili Wang, Yan Zhao, Haibin Wu and Yuji Iwahori
Multi-Scale Spatial–Spectral Attention-Based Neural Architecture Search for Hyperspectral Image Classification
Reprinted from: *Electronics* **2023**, *12*, 3641, https://doi.org/10.3390/electronics12173641 . . . . . **144**

Jingfang Wei, Haibin Wu, Qing Wu, Yuji Iwahori, Xiaoyu Yu and Aili Wang
Gaze Estimation Method Combining Facial Feature Extractor with Pyramid Squeeze Attention Mechanism
Reprinted from: *Electronics* **2023**, *12*, 3104, https://doi.org/10.3390/electronics12143104 . . . . . **166**

Thanasit Rithanasophon, Kitsaphon Thitisiriwech, Pittipol Kantavat, Boonserm Kijsirikul, Yuji Iwahori, Shinji Fukui, et al.
Quality of Life Prediction on Walking Scenes Using Deep Neural Networks and Performance Improvement Using Knowledge Distillation
Reprinted from: *Electronics* **2023**, *12*, 2907, https://doi.org/10.3390/electronics12132907 . . . . . **186**

Tian Han, Zhu Zhang, Mingyuan Ren, Changchun Dong, Xiaolin Jiang and Quansheng Zhuang
Text Emotion Recognition Based on XLNet-BiGRU-Att
Reprinted from: *Electronics* **2023**, *12*, 2704, https://doi.org/10.3390/electronics12122704 . . . . . **211**

Xiaoyu Yu, Yan Zhang, Haibin Wu and Aili Wang
An Improved Unscented Kalman Filtering Combined with Feature Triangle for Head Position Tracking
Reprinted from: *Electronics* **2023**, *12*, 2665, https://doi.org/10.3390/electronics12122665 . . . . . **226**

Xiaoyu Yu, Wenjun Yuan and Aili Wang
X-ray Security Inspection Image Dangerous Goods Detection Algorithm Based on Improved YOLOv4
Reprinted from: *Electronics* **2023**, *12*, 2644, https://doi.org/10.3390/electronics12122644 . . . . . **248**

Jingwei Dong, Ziqi Zhao and Tongxin Wang
Crowd Counting by Multi-Scale Dilated Convolution Networks
Reprinted from: *Electronics* **2023**, *12*, 2624, https://doi.org/10.3390/electronics12122624 . . . . . **265**

Jingwei Dong, Yushun Zhang and Lingye Fan
A Multi-View Face Expression Recognition Method Based on DenseNet and GAN
Reprinted from: *Electronics* **2023**, *12*, 2527, https://doi.org/10.3390/electronics12112527 . . . . . **281**

Tian Han, Zhu Zhang, Mingyuan Ren, Changchun Dong, Xiaolin Jiang and Quansheng Zhuang
Speech Emotion Recognition Based on Deep Residual Shrinkage Network
Reprinted from: *Electronics* **2023**, *12*, 2512, https://doi.org/10.3390/electronics12112512 . . . . . **298**

Bohao Li, Kai Xie, Xuepeng Zeng, Mingxuan Cao, Chang Wen, Jianbiao He and Wei Zhang
Anomalous Behavior Detection with Spatiotemporal Interaction and Autoencoder Enhancement
Reprinted from: *Electronics* **2023**, *12*, 2438, https://doi.org/10.3390/electronics12112438 . . . . . **320**

Qiong Wu, Hai Huang, Xinmiao Lu, Jiaxing Qu, Juntao Gu and Cunfang Yang
E-ReInForMIF Routing Algorithm Based on Energy Selection and Erasure Code Tolerance Machine
Reprinted from: *Electronics* **2023**, *12*, 2408, https://doi.org/10.3390/electronics12112408 . . . . . **339**

Xuelai Fang and Xiangchu Feng
Domain-Aware Adaptive Logarithmic Transformation
Reprinted from: *Electronics* **2023**, *12*, 1318, https://doi.org/10.3390/electronics12061318 . . . . . **352**

Xinmiao Lu, Cunfang Yang, Qiong Wu, Jiaxu Wang, Yuhan Wei, Liyu Zhang, et al.
Improved Reconstruction Algorithm of Wireless Sensor Network Based on BFGS Quasi-Newton Method
Reprinted from: *Electronics* **2023**, *12*, 1267, https://doi.org/10.3390/electronics12061267 . . . . . **368**

Xing Zi, Kunal Chaturvedi, Ali Braytee, Jun Li and Mukesh Prasad
Detecting Human Falls in Poor Lighting: Object Detection and Tracking Approach for Indoor Safety
Reprinted from: *Electronics* **2023**, *12*, 1259, https://doi.org/10.3390/electronics12051259 . . . . . **383**

Lanfei Zhao, Shijun Li and Jun Wang
Deep Signal-Dependent Denoising Noise Algorithm
Reprinted from: *Electronics* **2023**, *12*, 1201, https://doi.org/10.3390/electronics12051201 . . . . . **395**

Xue Wang, Qian Chen and Xiaoyang Yu
Research on Spectrum Prediction Technology Based on B-LTF
Reprinted from: *Electronics* **2023**, *12*, 247, https://doi.org/10.3390/electronics12010247 . . . . . **412**

Xinmiao Lu, Cunfang Yang, Qiong Wu, Jiaxu Wang, Zihan Lu, Shuai Sun, et al.
Research on Analog Circuit Soft Fault Diagnosis Method Based on Mathematical Morphology Fractal Dimension
Reprinted from: *Electronics* **2023**, *12*, 184, https://doi.org/10.3390/electronics12010184 . . . . . **431**

Xinmiao Lu, Zihan Lu, Qiong Wu, Jiaxu Wang, Cunfang Yang, Shuai Sun, et al.
Soft Fault Diagnosis of Analog Circuit Based on EEMD and Improved MF-DFA
Reprinted from: *Electronics* **2023**, *12*, 114, https://doi.org/10.3390/electronics12010114 . . . . . **452**

Lanfei Zhao, Guoqing Li and Jun Wang
Tone Mapping Method Based on the Least Squares Method
Reprinted from: *Electronics* **2023**, *12*, 31, https://doi.org/10.3390/electronics12010031 . . . . . **468**

Lanfei Zhao, Ruiyang Sun and Jun Wang
Three-Stage Tone Mapping Algorithm
Reprinted from: *Electronics* **2022**, *11*, 4072, https://doi.org/10.3390/electronics11244072 . . . . . **482**

Han Ye, Xiaodong Zhu, Chengyang Liu, Linlin Yang and Aili Wang
Furniture Image Classification Based on Depthwise Group Over-Parameterized Convolution
Reprinted from: *Electronics* **2022**, *11*, 3889, https://doi.org/10.3390/electronics11233889 . . . . . **495**

Kosuke Honda, Masaki Kurematsu, Hamido Fujita, and Ali Selamat
Multi-Task Learning for Scene Text Image Super-Resolution with Multiple Transformers
Reprinted from: *Electronics* **2022**, *11*, 3813, https://doi.org/10.3390/electronics11223813 . . . . . **507**

Mingzhu Liu, Junyu Chen and Xiaofei Han
Research on Retinex Algorithm Combining with Attention Mechanism for Image Enhancement
Reprinted from: *Electronics* **2022**, *11*, 3695, https://doi.org/10.3390/electronics11223695 . . . . . **525**

Xinmiao Lu, Yuhan Wei, Qiong Wu, Cunfang Yang, Dongyuan Li, Liyu Zhang and Ying Zhou
A Coverage Hole Patching Algorithm for Heterogeneous Wireless Sensor Networks
Reprinted from: *Electronics* **2022**, *11*, 3563, https://doi.org/10.3390/electronics11213563 . . . . . **538**

Mingzhu Liu, Ben Li and Wei Zhang
Research on Small Acceptance Domain Text Detection Algorithm Based on Attention Mechanism and Hybrid Feature Pyramid
Reprinted from: *Electronics* **2022**, *11*, 3559, https://doi.org/10.3390/electronics11213559 . . . . . **547**

Guodong Li, Xvan Qin, He Liu, Kaiyuan Jiang and Aili Wang
Modulation Recognition of Digital Signal Using Graph Feature and Improved K-Means
Reprinted from: *Electronics* **2022**, *11*, 3298, https://doi.org/10.3390/electronics11203298 . . . . . **562**

Wenbo Jiang, Tong Ren and Qianhua Fu
Deep Learning in the Phase Extraction of Electronic Speckle Pattern Interferometry
Reprinted from: *Electronics* **2024**, *13*, 418, https://doi.org/10.3390/electronics13020418 . . . . . **575**

About the Editors

Yuji Iwahori

Yuji Iwahori (Member, IEEE) received his B.S. degree from Nagoya Institute of Technology, in 1983, and his M.S. and Ph.D. degrees from the Department of Electrical and Electronics, Tokyo Institute of Technology, in 1985 and 1988, respectively. In 1988, he joined the Educational Centre for Information Processing, Nagoya Institute of Technology, as a Research Associate, where he became a Professor at the Centre for Information and Media Studies, in 2002. From 1991 to 2024, he was a Visiting Researcher at Computer Science Department, UBC. From 2010 to 2024, he held a position as a Research Collaborator with IIT Guwahati and at the Department of Computer Engineering, Chulalongkorn University from 2014 to 2024. In 2004, he joined Chubu University as a Professor. He acted as the Department Head of Computer Science, the Head of the Graduate Computer Science Course, and the Vice-Dean of the College of Engineering. He became an Honorary Faculty Member of IIT Guwahati in 2020. His research interests include computer vision, biomedical image processing, deep learning, and the application of artificial intelligence. He was a recipient of the KES 2008 Best Paper Award and the KES 2013 Best Paper Award from KES International. Unfortunately, he passed away on 24th December, 2024.

Aili Wang

Aili Wang (Member, IEEE) received her B.S., M.S., and Ph.D. degrees in information and signal processing from Harbin Institute of Technology, Harbin, China, in 2002, 2004, and 2008, respectively. She has been a Visiting Professor in the research of 3D polyp reconstruction with the Computer Science Laboratory, Chubu University, Kasugai, Japan, in 2014. She is the author of two books and more than 160 articles. Her research interests include deep learning-based space intelligent remote sensing and image processing.

Haibin Wu

Haibin Wu received his B.S. and M.S. degrees from the Harbin Institute of Technology, Harbin, China, in 2000 and 2002, respectively, and his Ph.D. degree in Measuring and Testing Technologies and Instruments from Harbin University of Science and Technology, Harbin, in 2008. From 2014 to 2015, he was a Visiting Scholar with the Robot Perception and Action Laboratory, University of South Florida, Tampa, FL, USA. He is the author of three books, more than 120 articles, and the creator of more than 20 inventions. His research interests include robotic vision, visual measuring and image processing, medical virtual reality, and photoelectric testing.

electronics

MDPI

Editorial

Deep Learning in Image Processing and Pattern Recognition

Aili Wang [1,*], Haibin Wu [1] and Yuji Iwahori [2]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; woo@hrbust.edu.cn
[2] Department of Computer Science, Chubu University, 1200 Matsumoto-cho, Kasugai 487-8501, Japan; iwahori@isc.chubu.ac.jp
* Correspondence: aili925@hrbust.edu.cn

1. Image Preprocessing Field

The current field shows a trend of multi-dimensional fusion [1], the use of lightweight convolutional self-encoder and generative adversarial network in denoising, super-resolution tasks beyond the traditional methods, and multimodal fusion technology through the integration of visible/infrared/depth map data to enhance feature extraction. In future, it is necessary to build a quantum entanglement parallel denoising system, develop neural radiation field three-dimensional dynamic reconstruction technology, and integrate optoelectronic hardware design to guarantee data security [2].

Image preprocessing techniques focus on improving data representation and quality optimization, and the following research articles included in this Special Issue drive innovation in image preprocessing methods in cross-modal scenarios. For image enhancement and dynamic scene analysis, "Fully Automatic Approach for Smoke Tracking Based on Deep Image Quality Enhancement and Adaptive Level Set Model" suppresses noise and reconstructs the image through a convolutional self-encoder, combining HSV thresholding and dynamic energy features to achieve smoke tracking. "Deep Signal-Dependent Denoising Noise Algorithm" combines Retinex enhancement and reparameterized convolution to achieve 30 fps real-time inference in low-illumination fall detection. In the field of HDR tone mapping, "Tone Mapping Method Based on the Least Squares Method" and "Three-Stage Tone Mapping Algorithm" are based on the improved light estimation of the Retinex model and the three-stage compression mechanism, respectively, to balance the dynamic range and detail retention. In terms of image enhancement, "Research on Retinex Algorithm Combining with Attention Mechanism for Image Enhancement" optimizes the decomposition and illumination reconstruction network of Enhance-Net through the attention mechanism to alleviate the noise and color distortion problems of Retinex-Net.

2. Feature and Image Selection

A self-supervised and comparative learning framework significantly reduces the dependence on labeled data [3], and the attention mechanism is combined with reinforcement learning to optimize dynamic sampling. In future, it is necessary to build a self-supervised contrast collaboration framework, develop Transformer–dynamic convolution hybrid architecture [4], and strengthen cross-scale modeling and interpretability.

Feature extraction and image selection techniques, on the other hand, focus on improving data representation, and the following articles in this Special Issue achieve the optimization of feature representation in specific areas. These techniques serve more in the preprocessing stage, providing quality data input for subsequent classification and detection tasks. "Detection of Small Lesions on Grape Leaves Based on Improved YOLOv7"

proposes an improved YOLOv7 model for the problems of high leakage rate and inaccurate localization in the detection of small lesions on grape leaves. In "Defect Detection Method of Phosphor in Glass Based on Improved YOLO5 Algorithm", the YOLOv5 algorithm is improved to detect small defects in PiG materials. "Crowd Counting by Multi-Scale Dilated Convolution Networks" proposes a multi-scale hollow convolution network to solve the problem of the uneven distribution of dense crowds and head scale difference. "Research on Small Acceptance Domain Text Detection Algorithm Based on Attention Mechanism and Hybrid Feature Pyramid" proposes a fusion of attention mechanism and hybrid feature pyramid to address the issue of insufficient feature extraction for small text detection in complex scenes. The lightweight network architecture is embedded with a hybrid attention module, and a hybrid feature pyramid is constructed to integrate shallow details and deep semantic features across layers, and combined with a bidirectional long short-term memory network to strengthen contextual modeling.

3. Image Processing and Pattern Recognition

Vision Transformer dominates image classification and segmentation through the self-attention mechanism, and dynamic sparse attention improves real-time analysis capabilities [5]. In future, we need to design a multimodal synergy framework, develop a physical embedding model to integrate a priori knowledge such as light field equations, and combine it with dynamic pruning to balance performance [6].

Image processing pattern recognition technology is directly aimed at target detection, classification, and segmentation tasks, and solves practical challenges such as occlusion and small targets through algorithm optimization. The following articles in this Special Issue reflect task-specific algorithm improvements; aiming at the frontiers of multimodal perception and intelligent computing, this multi-disciplinary research proposes innovative solutions. "Offline Mongolian Handwriting Recognition Based on Data Augmentation and Improved ECA-Net" improves the recognition capability of complex concatenated characters by combining the dual data augmentation strategy of random erasure and horizontal waveform transformation, combined with an improved channel attention mechanism, enhancing complex concatenated character recognition. In the field of target detection, "YOLO-Rlepose: Improved YOLO Based on Swin Transformer and Rle-Oks Loss for Multi-Person Pose Estimation" integrates Swin Transformer and Rle-Oks loss function to solve the occlusion interference problem in multi-person pose estimation. This problem is also solved by "A Residual Network with Efficient Transformer for Lightweight Image Super-Resolution" and "Pixel-Level Degradation for Text Image Super-Resolution and Recognition" which used residual network and Rle-Oks loss function, respectively.

In terms of line-of-sight and pose tracking, "Gaze Estimation Method Combining Facial Feature Extractor with Pyramid Squeeze Attention Mechanism" combines pyramid squeeze attention to improve environmental robustness, whereas "An Improved Unscented Kalman Filtering Combined with Feature Triangle for Head Position Tracking" improves the head tracking accuracy in surgical scenes using geometric feature triangles. In the field of multimodal recognition, "A Multi-View Face Expression Recognition Method Based on DenseNet and GAN" adopts lightweight DSC-DenseNet and a dual-path LD-GAN model, while "Speech Emotion Recognition Based on Deep Residual Shrinkage Network" combines Deep Residual Shrinkage Network with bidirectional GRU to extract speech emotion features. Finally, "Anomalous Behavior Detection with Spatiotemporal Interaction and Autoencoder Enhancement" proposes a joint framework of spatiotemporal interaction graph convolution and confidence-enhanced autoencoder.

4. Image Processing in Intelligent Transportation

In the field of intelligent transportation, multi-sensor fusion is used to build high-precision 3D environment models [7], event cameras help to break through the traditional frame rate limitations, and federated learning is employed to optimize global traffic prediction. In future, we need to develop an impulse neural network to drive heterogeneous data alignment, construct a meta-learning cross-domain adaptive framework, and establish a privacy security sharing mechanism [8].

The intelligent transportation field relies on technologies such as monocular depth estimation and lightweight CNN models to promote the implementation of autonomous driving and traffic management. The following articles in this Special Issue promote the development of technology in this field. "Quality of Life Prediction on Walking Scenes Using Deep Neural Networks and Performance Improvement Using Knowledge Distillation" proposes a deep learning and knowledge distillation-based walking scenes quality-of-life prediction method to replace traditional high-cost questionnaires. The authors extract 15 types of scene features such as pedestrian density and road width using YOLOv4 target detection and DDRNet-23-slim semantic segmentation, and construct a deep model integrating one-dimensional convolution, bidirectional LSTM, and gated recurrent units to capture spatiotemporal features in order to infer QoL scores.

The articles "YOLO-Rlepose: Improved YOLO Based on Swin Transformer and RLE-Oks Loss for Multi-Person Pose Estimation" and "Detecting Human Falls in Poor Lighting: Object Detection and Tracking Approach for Indoor Safety", mentioned in the previous section, also make contributions to this field.

5. Hyperspectral Image Processing

End-to-end models are employed to realize the accurate classification of agricultural pests and diseases, whereas physically driven unmixing algorithms can break through the hardware acquisition limit [9]. In future, it is necessary to build a hybrid architecture of self-attention–deformable convolution, develop a semi-supervised generation framework to integrate spectral unmixing theory, and promote hardware collaborative optimization [10].

The optimization of neural network architecture design and training strategy constitutes another important direction. The following articles in this Special Issue balance the efficiency and performance of the model and optimize the network for specific scenarios. "Multi-Scale Spatial-Spectral Attention-Based Neural Architecture Search for Hyperspectral Image Classification" designs a multi-scale attention-extended search space, integrating different scale convolution and attention modules such as CBAM and SE to strengthen spatial feature extraction; a slow–fast learning paradigm is introduced to optimize the architecture search efficiency by combining group intelligence; and the Lion optimizer is used to update the parameters by symbolic operations to reduce the memory and training costs.

6. Biomedical Image Processing

In biomedicine, multimodal fusion technology improves tumor localization accuracy, self-supervised models reduce the cost of medical diagnosis [11], and adversarial networks solve the problem of long-tailed distribution. In future, we need to develop a three-dimensional cross-modal alignment framework, integrate adversarial generation and domain adaptation techniques, and revolutionize the paradigm of medical image analysis [12].

The field of biomedical image processing is supported by medical imaging-specific algorithms to improve diagnostic accuracy. "Fully Automatic Approach for Smoke Tracking Based on Deep Image Quality Enhancement and Adaptive Level Set Model" and "Speech

Emotion Recognition Based on Deep Residual Shrinkage Network" provide technical support in this context.

7. Advances, Challenges, and Research Trends in Deep Learning for Image Processing and Pattern Recognitions

Graph convolutional networks realize the real-time detection of abnormal events, and federated learning promotes cross-border security compliance [13]. In future, we need to couple privacy computing and edge reasoning, deepen spatiotemporal graph convolutional cross-modal parsing, and build a multimodal meta-knowledge base [14].

Image processing techniques for intelligent surveillance scenarios focus on human behavior analysis and security applications, and the following articles in this Special Issue reflect the importance of scenario-based algorithm design. In "X-ray Security Inspection Image Dangerous Goods Detection Algorithm Based on Improved YOLOv4", the deformable convolutional reconstruction PANet module is introduced to dynamically adjust the sensing field to align the features and solve the problem of misalignment in complex shape detection; the Focal-EIoU loss function is used to balance the weights of difficult and easy samples to accelerate convergence; and the confidence of overlapping frames is dynamically adjusted with Soft-NMS to reduce leakage detection.

Moreover, the previously mentioned articles "Fully Automatic Approach for Smoke Tracking Based on Deep Image Quality Enhancement and Adaptive Level Set Model", "Quality of Life Prediction on Walking Scenes Using Deep Neural Networks and Performance Improvement Using Knowledge Distillation", "Anomalous Behavior Detection with Spatiotemporal Interaction and Autoencoder Enhancement", and "Detecting Human Falls in Poor Lighting: Object Detection and Tracking Approach for Indoor Safety" all contribute to this field to varying degrees.

8. Deep Learning for Image Processing

Dynamic inference technology balances the efficiency of denoising tasks, whereas model compression technology adapts to embedded devices [15]. In future, it is necessary to integrate adaptive computing flow and heterogeneous compression frameworks [16] and construct a causal graph network to improve the robustness of industrial algorithms.

Deep learning-based end-to-end image processing techniques focus on image generation, reconstruction, and editing, and the following articles in this Special Issue provide technical support in this area. Aiming at the problem of target detection and text processing optimization, "A Multi-Stage Adaptive Copy-Paste Data Augmentation Algorithm Based on Model Training Preferences" proposes a staged adaptive copy-paste augmentation strategy to dynamically adjust the tail category sample size to balance the detection task category bias. "Text Emotion Recognition Based on XLNet-BiGRU-Attention" fuses XLNet two-stream self-attention and BiGRU-attention mechanisms to improve the efficiency of text emotion feature extraction through semantic–positional cooperative coding.

In the field of wireless sensor networks, "E-ReInForMIF Routing Algorithm Based on Energy Selection and Erasure Code Tolerance Machine" combines erasure code redundancy coding and energy equalization routing to enhance the fault tolerance of data transmission, whereas "Improved Reconstruction Algorithm of Wireless Sensor Network Based on BFGSQuasi-Newton Method" optimizes compressed sensory reconstruction using the L-BFGS quasi-Newton method and guarantees the convergence of the algorithm through Wolfe line search. On the other hand, "Research on Spectrum Prediction Technology Based on B-LTF" constructs a hybrid BP-LSTM model, which utilizes BP to extract nonlinear features and LSTM to capture temporal dependencies, so as to enhance the accuracy of dynamic spectrum prediction.

In image processing, "Multi-Task Learning for Scene Text Image Super-Resolution with Multiple Transformers" designs a multi-task Transformer architecture to optimize the super-scoring and denoising tasks through joint feature sharing and enhancement modules, and combines positional coding to cope with text deformation. In network coverage optimization, "A Coverage Hole Patching Algorithm for Heterogeneous Wireless Sensor Networks" proposes a mobile node scheduling strategy based on outer circle localization to dynamically fill the coverage holes in heterogeneous networks according to vulnerability size priority.

Moreover, the previously mentioned articles "Offline Mongolian Handwriting Recognition Based on Data Augmentation and Improved ECA-Net", "YOLO-Rlepose: Improved YOLO Based on Swin Transformer and Rle-Oks Loss for Multi-Person Pose Estimation", "Detection of Small Lesions on Grape Leaves Based on Improved YOLOv7", "A Residual Network with Efficient Transformer for Lightweight Image Super-Resolution", "Pixel-Level Degradation for Text Image Super-Resolution and Recognition", "Multi-Scale Spatial–Spectral Attention-Based Neural Architecture Search for Hyperspectral Image Classification", and "Deep Learning in the Phase Extraction of Electronic Speckle Pattern Interferometry" all contribute to this field to varying degrees.

9. AI-Based Image Processing, Understanding, Recognition, Compression, and Reconstruction

Generative AI expands the boundaries of creative design, neural compression technology saves 40% of the bit rate [17], and 3D reconstruction technology empowers digital twins. In future, it is necessary to develop a diffusion model-driven reconstruction framework and promote the lightweight Transformer architecture to reduce computational costs [18].

AI-driven full-flow image processing technology integrates compression, analysis, and generation, and the following articles in this Special Issue provide technical support in this area. "Domain-Aware Adaptive Logarithmic Transformation" proposes a method called AdaLogT to solve the problem of oversaturation and detail loss caused by ignoring the differences in image properties in traditional tone mapping. A sub-domain objective function is constructed by the parameter p: for the luminance domain algorithm, the exposure bias and histogram skewness are jointly optimized, and the optimal p-value is solved using the trilateration method; for the gradient domain algorithm, the texture information is maximized based on the exponential mean and local variance; and the DNN algorithm fuses the luminance-gradient domain features to balance the global and local information.

Taken together, the 15 articles mentioned above, "A Residual Network with Efficient Transformer for Lightweight Image Super-Resolution", "Pixel-Level Degradation for Text Image Super-Resolution and Recognition", "A Multi-Stage Adaptive Copy-Paste Data Augmentation Algorithm Based on Model Training Preferences", "Quality of Life Prediction on Walking Scenes Using Deep Neural Networks and Performance Improvement Using Knowledge Distillation", "X-ray Security Inspection Image Dangerous Goods Detection Algorithm Based on Improved YOLOv4", "E-ReInForMIF Routing Algorithm Based on Energy Selection and Erasure Code Tolerance Machine", "Improved Reconstruction Algorithm of Wireless Sensor Network Based on BFGSQuasi-Newton Method", "Research on Spectrum Prediction Technology Based on B-LTF", "Tone Mapping Method Based on the Least Squares Method", "Three-Stage Tone Mapping Algorithm", "Multi-Task Learning for Scene Text Image Super-Resolution with Multiple Transformers", "Research on Retinex Algorithm Combining with Attention Mechanism for Image Enhancement", "A Coverage Hole Patching Algorithm for Heterogeneous Wireless Sensor Networks", "Modulation Recognition of Digital Signal Using Graph Feature and Improved K-Means", and "Deep

Learning in the Phase Extraction of Electronic Speckle Pattern Interferometry", provide a complete solution from data preprocessing to application.

Author Contributions: Conceptualization, A.W., H.W. and Y.I.; writing—original draft preparation, A.W. and H.W.; writing—review and editing, A.W. and H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflicts of interest.

List of Contributions:

1. Ren, Q.-D.-E.-J.; Wang, L.; Ma, Z.; Barintag, S. Offline Mongolian Handwriting Recognition Based on Data Augmentation and Improved ECA-Net. *Electronics* **2024**, *13*, 835. https://doi.org/10.3390/electronics13050835.
2. Jiang, Y.; Yang, K.; Zhu, J.; Qin, L. YOLO-Rlepose: Improved YOLO Based on Swin Transformer and Rle-Oks Loss for Multi-Person Pose Estimation. *Electronics* **2024**, *13*, 563. https://doi.org/10.3390/electronics13030563.
3. Yang, M.; Tong, X.; Chen, H. Detection of Small Lesions on Grape Leaves Based on Improved YOLOv7. *Electronics* **2024**, *13*, 464. https://doi.org/10.3390/electronics13020464.
4. Yan, F.; Li, S.; Zhou, Z.; Shi, Y. A Residual Network with Efficient Transformer for Lightweight Image Super-Resolution. *Electronics* **2024**, *13*, 194. https://doi.org/10.3390/electronics13010194.
5. Qian, X.; Xie, L.; Ye, N.; Le, R.; Yang, S. Pixel-Level Degradation for Text Image Super-Resolution and Recognition. *Electronics* **2023**, *12*, 4546. https://doi.org/10.3390/electronics12214546.
6. Qin, Y.; Pan, Z.; Shao, C. Defect Detection Method of Phosphor in Glass Based on Improved YOLO5 Algorithm. *Electronics* **2023**, *12*, 3917. https://doi.org/10.3390/electronics12183917.
7. Daoudi, R.; Mouelhi, A.; Bouchouicha, M.; Moreau, E.; Sayadi, M. Fully Automatic Approach for Smoke Tracking Based on Deep Image Quality Enhancement and Adaptive Level Set Model. *Electronics* **2023**, *12*, 3888. https://doi.org/10.3390/electronics12183888.
8. Yu, X.; Li, F.; Liu, Y.; Wang, A. A Multi-Stage Adaptive Copy-Paste Data Augmentation Algorithm Based on Model Training Preferences. *Electronics* **2023**, *12*, 3695. https://doi.org/10.3390/electronics12173695.
9. Song, Y.; Wang, A.; Zhao, Y.; Wu, H.; Iwahori, Y. Multi-Scale Spatial–Spectral Attention-Based Neural Architecture Search for Hyperspectral Image Classification. *Electronics* **2023**, *12*, 3641. https://doi.org/10.3390/electronics12173641.
10. Wei, J.; Wu, H.; Wu, Q.; Iwahori, Y.; Yu, X.; Wang, A. Gaze Estimation Method Combining Facial Feature Extractor with Pyramid Squeeze Attention Mechanism. *Electronics* **2023**, *12*, 3104. https://doi.org/10.3390/electronics12143104.
11. Rithanasophon, T.; Thitisiriwech, K.; Kantavat, P.; Kijsirikul, B.; Iwahori, Y.; Fukui, S.; Nakamura, K.; Hayashi, Y. Quality of Life Prediction on Walking Scenes Using Deep Neural Networks and Performance Improvement Using Knowledge Distillation. *Electronics* **2023**, *12*, 2907. https://doi.org/10.3390/electronics12132907.
12. Han, T.; Zhang, Z.; Ren, M.; Dong, C.; Jiang, X.; Zhuang, Q. Text Emotion Recognition Based on XLNet-BiGRU-Att. *Electronics* **2023**, *12*, 2704. https://doi.org/10.3390/electronics12122704.
13. Yu, X.; Zhang, Y.; Wu, H.; Wang, A. An Improved Unscented Kalman Filtering Combined with Feature Triangle for Head Position Tracking. *Electronics* **2023**, *12*, 2665. https://doi.org/10.3390/electronics12122665.
14. Yu, X.; Yuan, W.; Wang, A. X-ray Security Inspection Image Dangerous Goods Detection Algorithm Based on Improved YOLOv4. *Electronics* **2023**, *12*, 2644. https://doi.org/10.3390/electronics12122644.
15. Dong, J.; Zhao, Z.; Wang, T. Crowd Counting by Multi-Scale Dilated Convolution Networks. *Electronics* **2023**, *12*, 2624. https://doi.org/10.3390/electronics12122624.
16. Dong, J.; Zhang, Y.; Fan, L. A Multi-View Face Expression Recognition Method Based on DenseNet and GAN. *Electronics* **2023**, *12*, 2527. https://doi.org/10.3390/electronics12112527.

17. Han, T.; Zhang, Z.; Ren, M.; Dong, C.; Jiang, X.; Zhuang, Q. Speech Emotion Recognition Based on Deep Residual Shrinkage Network. *Electronics* **2023**, *12*, 2512. https://doi.org/10.3390/electronics12112512.
18. Li, B.; Xie, K.; Zeng, X.; Cao, M.; Wen, C.; He, J.; Zhang, W. Anomalous Behavior Detection with Spatiotemporal Interaction and Autoencoder Enhancement. *Electronics* **2023**, *12*, 2438. https://doi.org/10.3390/electronics12112438.
19. Wu, Q.; Huang, H.; Lu, X.; Qu, J.; Gu, J.; Yang, C. E-ReInForMIF Routing Algorithm Based on Energy Selection and Erasure Code Tolerance Machine. *Electronics* **2023**, *12*, 2408. https://doi.org/10.3390/electronics12112408.
20. Fang, X.; Feng, X. Domain-Aware Adaptive Logarithmic Transformation. *Electronics* **2023**, *12*, 1318. https://doi.org/10.3390/electronics12061318.
21. Lu, X.; Yang, C.; Wu, Q.; Wang, J.; Wei, Y.; Zhang, L.; Li, D.; Zhao, L. Improved Reconstruction Algorithm of Wireless Sensor Network Based on BFGS Quasi-Newton Method. *Electronics* **2023**, *12*, 1267. https://doi.org/10.3390/electronics12061267.
22. Zi, X.; Chaturvedi, K.; Braytee, A.; Li, J.; Prasad, M. Detecting Human Falls in Poor Lighting: Object Detection and Tracking Approach for Indoor Safety. *Electronics* **2023**, *12*, 1259. https://doi.org/10.3390/electronics12051259.
23. Zhao, L.; Li, S.; Wang, J. Deep Signal-Dependent Denoising Noise Algorithm. *Electronics* **2023**, *12*, 1201. https://doi.org/10.3390/electronics12051201.
24. Wang, X.; Chen, Q.; Yu, X. Research on Spectrum Prediction Technology Based on B-LTF. *Electronics* **2023**, *12*, 247. https://doi.org/10.3390/electronics12010247.
25. Lu, X.; Yang, C.; Wu, Q.; Wang, J.; Lu, Z.; Sun, S.; Liu, K.; Shao, D. Research on Analog Circuit Soft Fault Diagnosis Method Based on Mathematical Morphology Fractal Dimension. *Electronics* **2023**, *12*, 184. https://doi.org/10.3390/electronics12010184.
26. Lu, X.; Lu, Z.; Wu, Q.; Wang, J.; Yang, C.; Sun, S.; Shao, D.; Liu, K. Soft Fault Diagnosis of Analog Circuit Based on EEMD and Improved MF-DFA. *Electronics* **2023**, *12*, 114. https://doi.org/10.3390/electronics12010114.
27. Zhao, L.; Li, G.; Wang, J. Tone Mapping Method Based on the Least Squares Method. *Electronics* **2023**, *12*, 31. https://doi.org/10.3390/electronics12010031.
28. Zhao, L.; Sun, R.; Wang, J. Three-Stage Tone Mapping Algorithm. *Electronics* **2022**, *11*, 4072. https://doi.org/10.3390/electronics11244072.
29. Ye, H.; Zhu, X.; Liu, C.; Yang, L.; Wang, A. Furniture Image Classification Based on Depthwise Group Over-Parameterized Convolution. *Electronics* **2022**, *11*, 3889. https://doi.org/10.3390/electronics11233889.
30. Honda, K.; Kurematsu, M.; Fujita, H.; Selamat, A. Multi-Task Learning for Scene Text Image Super-Resolution with Multiple Transformers. *Electronics* **2022**, *11*, 3813. https://doi.org/10.3390/electronics11223813.
31. Liu, M.; Chen, J.; Han, X. Research on Retinex Algorithm Combining with Attention Mechanism for Image Enhancement. *Electronics* **2022**, *11*, 3695. https://doi.org/10.3390/electronics11223695.
32. Lu, X.; Wei, Y.; Wu, Q.; Yang, C.; Li, D.; Zhang, L.; Zhou, Y. A Coverage Hole Patching Algorithm for Heterogeneous Wireless Sensor Networks. *Electronics* **2022**, *11*, 3563. https://doi.org/10.3390/electronics11213563.
33. Liu, M.; Li, B.; Zhang, W. Research on Small Acceptance Domain Text Detection Algorithm Based on Attention Mechanism and Hybrid Feature Pyramid. *Electronics* **2022**, *11*, 3559. https://doi.org/10.3390/electronics11213559.
34. Li, G.; Qin, X.; Liu, H.; Jiang, K.; Wang, A. Modulation Recognition of Digital Signal Using Graph Feature and Improved K-Means. *Electronics* **2022**, *11*, 3298. https://doi.org/10.3390/electronics11203298.
35. Jiang, W.; Ren, T.; Fu, Q. Deep Learning in the Phase Extraction of Electronic Speckle Pattern Interferometry. *Electronics* **2024**, *13*, 418. https://doi.org/10.3390/electronics13020418.

References

1. Bhattacharyya, S. A brief survey of color image preprocessing and segmentation techniques. *J. Pattern Recognit. Res.* **2011**, *1*, 120–129. [CrossRef] [PubMed]
2. Krig, S.; Krig, S. Image Pre-processing. In *Computer Vision Metrics: Textbook Edition*; Springer: Cham, Switzerland, 2016; pp. 35–74.
3. Li, Y.; Tao, L.; Huan, L. Recent advances in feature selection and its applications. *Knowl. Inf. Syst.* **2017**, *53*, 551–577. [CrossRef]
4. Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Trevino, R.P.; Tang, J.; Liu, H. Feature selection: A data perspective. *ACM Comput. Surv. (CSUR)* **2017**, *50*, 1–45. [CrossRef]
5. Shamir, L.; Delaney, J.D.; Orlov, N.; Eckley, D.M.; Goldberg, I.G. Pattern recognition software and techniques for biological image analysis. *PLoS Comput. Biol.* **2010**, *6*, e1000974. [CrossRef] [PubMed]
6. Uchida, S. Image processing and recognition for biological images. *Dev. Growth Differ.* **2013**, *55*, 523–549. [CrossRef] [PubMed]
7. Ge, D.-Y.; Yao, X.-F.; Xiang, W.-J.; Chen, Y.-P. Vehicle detection and tracking based on video image processing in intelligent transportation system. *Neural Comput. Appl.* **2023**, *35*, 2197–2209. [CrossRef]
8. Hao, Q.; Qin, L. The design of intelligent transportation video processing system in big data environment. *IEEE Access* **2020**, *8*, 13769–13780. [CrossRef]
9. Ghamisi, P.; Yokoya, N.; Li, J.; Liao, W.; Liu, S.; Plaza, J.; Rasti, B.; Plaza, A. Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 37–78. [CrossRef]
10. Peng, J.; Sun, W.; Li, W.; Li, H.-C.; Meng, X.; Ge, C.; Du, Q. Low-rank and sparse representation for hyperspectral image processing: A review. *IEEE Geosci. Remote Sens. Mag.* **2021**, *10*, 10–43. [CrossRef]
11. Rajeswari, J.; Jagannath, M. Advances in biomedical signal and image processing—A systematic review. *Inform. Med. Unlocked* **2017**, *8*, 13–19. [CrossRef]
12. Schindelin, J.; Rueden, C.T.; Hiner, M.C.; Eliceiri, K.W. The ImageJ ecosystem: An open platform for biomedical image analysis. *Mol. Reprod. Dev.* **2015**, *82*, 518–529. [CrossRef] [PubMed]
13. Kim, I.S.; Choi, H.S.; Yi, K.M.; Choi, J.Y.; Kong, S.G. Intelligent visual surveillance—A survey. *Int. J. Control. Autom. Syst.* **2020**, *8*, 926–939. [CrossRef]
14. He, F. Intelligent video surveillance technology in intelligent transportation. *J. Adv. Transp.* **2020**, *1*, 8891449. [CrossRef]
15. Valente, J.; António, J.; Mora, C.; Jardim, S. Developments in image processing using deep learning and reinforcement learning. *J. Imaging* **2023**, *9*, 207. [CrossRef] [PubMed]
16. Khalifa, I.A.; Faris, K. The Role of Image Processing and Deep Learning in IoT-Based Systems: A Comprehensive Review. *Eur. J. Appl. Sci. Eng. Technol.* **2025**, *3*, 165–179. [CrossRef] [PubMed]
17. Bousnina, N.; Ascenso, J.; Correia, P.L.; Pereira, F. Impact of conventional and ai-based image coding on ai-based face recognition performance. In Proceedings of the 2022 10th European Workshop on Visual Information Processing (EUVIP), Lisbon, Portugal, 11–14 September 2022; pp. 1–6.
18. Bauskar, S. View of Unveiling the Hidden Patterns AI-Driven Innovations in Image Processing and Acoustic Signal Detection. *J. Recent Trends Comput. Sci. Eng.* **2020**, *8*, 10–70589.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Offline Mongolian Handwriting Recognition Based on Data Augmentation and Improved ECA-Net

Qing-Dao-Er-Ji Ren [1], Lele Wang [1,*], Zerui Ma [1] and Saheya Barintag [2]

[1] School of Information Engineering, Inner Mongolia University of Technology, Hohhot 010051, China; renqingln@imut.edu.cn (Q.-D.-E.-J.R.); 20221800122@imut.edu.cn (Z.M.)
[2] School of Mathematics Science College, Inner Mongolia Normal University, Hohhot 010028, China; saheya@imnu.edu.cn
* Correspondence: wangll@imut.edu.cn

Abstract: Writing is an important carrier of cultural inheritance, and the digitization of handwritten texts is an effective means to protect national culture. Compared to Chinese and English handwriting recognition, the research on Mongolian handwriting recognition started relatively late and achieved few results due to the characteristics of the script itself and the lack of corpus. First, according to the characteristics of Mongolian handwritten characters, the random erasing data augmentation algorithm was modified, and a dual data augmentation (DDA) algorithm was proposed by combining the improved algorithm with horizontal wave transformation (HWT) to augment the dataset for training the Mongolian handwriting recognition. Second, the classical CRNN handwriting recognition model was improved. The structure of the encoder and decoder was adjusted according to the characteristics of the Mongolian script, and the attention mechanism was introduced in the feature extraction and decoding stages of the model. An improved handwriting recognition model, named the EGA model, suitable for the features of Mongolian handwriting was suggested. Finally, the effectiveness of the EGA model was verified by a large number of data tests. Experimental results demonstrated that the proposed EGA model improves the recognition accuracy of Mongolian handwriting, and the structural modification of the encoder and coder effectively balances the recognition accuracy and complexity of the model.

Keywords: attention mechanism; character recognition; data augmentation; neural network

1. Introduction

Writing, which records human history and inherits human civilization, is a unique skill and cultural symbol of human beings. In the Inner Mongolia region of China, the traditional Mongolian script used in Inner Mongolia is the Uighur-Mongolian script and plays an important role in the local historical record and cultural inheritance. Digitizing Mongolian handwriting in bulk using character recognition technology is an important way for Mongolian culture to keep pace with the times.

Training character recognition models by using deep neural networks is a common practice in the current character recognition field, and it has a significant effect on Chinese and English recognition. The LeNet5 CNN model proposed by Lecun et al. [1] in 1998 is regarded as an early classical model in the field of character recognition. After data augmentation, the LeNet5 CNN recognized handwriting in the MNIST dataset with 99.2% accuracy. In 2011, scholars from IDSIA Labs used GPUs to train CNNs, opening the application of neural networks in large-class (1000-class) Chinese handwriting recognition. Cire et al. [2] integrated multiple CNNs with various input scales and trained the model with the NIST SD19 dataset (having 800,000 samples), achieving a recognition accuracy of 89.12%. The recognition accuracy on the MNIST dataset was even better, reaching 99.72%. In 2015, Shi et al. [3] used CNN to extract the entire input image features and then

converted the extracted feature map into a feature sequence input to RNN to predict the output sequence. They calculated the sequence output probability with the connectionist temporal classification (CTC). This end-to-end trainable neural network based on image sequence recognition is a classical convolutional recurrent neural network (CRNN). CRNN not only has outstanding achievements in popular character recognition such as Chinese and English script, but also shows excellent recognition performance in various ethnic minority character recognition tasks. In 2021, CRNN was applied to scanned Uyghur character recognition [4], with a recognition accuracy of more than 90%. In the same year, CRNN was also used in recognizing the script of The Book of Changes [5] and Tibetan-Chinese bilingual text [6], and both achieved reasonable recognition results. In 2020, the Baidu Flypaddle team proposed a lightweight character recognition system: PP-OCR [7]. The overall size of the system model is only 3.5 MB and it can recognize 6622 Chinese characters and 63 alphanumeric symbols. After training, the model also showed good performance in character recognition for several other languages (French, Korean, Japanese, and German). Since 2020, researchers have extended the "transformer" to computer vision tasks. In 2020, Dosovitskiy et al. [8] proposed a Vision Transformer (ViT), which applies a standard Transformer directly to images, with the fewest possible modifications, inspired by the Transformer scaling successes in NLP. In addition, ViT has reached or exceeded the latest level on many image classification datasets, showing superior performance. In 2021, Liu et al. [9] of Microsoft Research Asia proposed a new type of vision transformer, Swin Transformer, that can be used as a general-purpose backbone network for computer vision. Experimental results have shown that the self-attention mechanism based on the shifted window introduced in the algorithm is an effective method to solve the visual problem. In 2022, Riaz Nauman et al. [10] proposed the CNN-transformer architecture to solve the problem of offline Urdu handwriting recognition. The effectiveness of the proposed method was verified on the public NUST-UHWR dataset. In 2023, Dan Yongping et al. [11] introduced the particle swarm optimization method into the design of CNN handwritten Chinese character recognition, which reduced redundant calculations in the network and achieved good experimental results.

Traditional Mongolian is a kind of script that is written from top to bottom, and the handwritten Mongolian script has the characteristics of diverse styles and flexible glyphs, and it is difficult to divide the morphemes. In 2017, Fan et al. conducted research on Mongolian handwriting recognition based on grapheme segmentation and implemented a handwriting recognition system by using a small-scale font library in the HTK (hidden Markov model toolkit) environment [12]. The experimental results showed that short graphemes have better performance than long graphemes. In recent years, Mongolian handwriting recognition based on whole words has begun to emerge. In 2019, Liu [13] proposed a sequence-to-sequence offline handwriting Mongolian whole-word recognition model with an attention mechanism. On large-scale datasets, the recognition accuracy of the model reached 81.56%. In 2020, Wei [14] proposed an end-to-end handwritten Mongolian whole-word recognition method and the experimental results showed that the method is superior to the traditional ones based on morpheme segmentation. This method not only achieved the optimal recognition effect at that time, but also alleviated the out-of-vocabulary (OOV) problem. In 2021, Yang [15] built an online Mongolian handwriting recognition cloud service system, providing online Mongolian handwriting recognition services via a service interface.

Data augmentation (DA) is an important means to improve the training effect of neural network models by augmenting the training set. DA algorithms for character recognition research have appeared for Chinese, English, and other languages. With the application of machine learning and neural networks in many fields, DA technology has also been greatly developed. In 1998, Lecun et al. [1] developed data augmentation for the MNIST database by using various operations, such as scaling, rotation, and stretching, to effectively improve the recognition accuracy of the character recognition model. Experimental results have shown that the elastic deformation data augmentation method [16] proposed for sequence

character recognition can help improve the efficiency of character recognition models. In 2017, Zhong et al. [17] proposed a random erasing data augmentation (REDA) algorithm for the target detection task. By erasing random areas in the target image, random erasing data augmentation algorithms can improve the robustness against occlusion and realize data augmentation. In 2021, Han et al. [18] used generative adversarial networks (GANs) to enhance the data of ancient characters, reconstruct the region-specific information of the character, and improve the ability of the model to extract image features. It can be seen that in the field of character recognition, it is theoretically feasible to augment the data of character images for training.

The abovementioned research indicates that the use of deep neural networks to achieve Mongolian handwriting recognition has a sufficient theoretical basis, but directly pouring traditional Mongolian corpus into the character recognition models for popular languages such as English and Chinese may not be feasible, since the recognition performance may be very poor. Therefore, we need to fully explore the unique features of traditional Mongolian on the basis of sufficient handwritten character corpus and build a model suitable for Mongolian handwritten script. Moreover, the traditional Mongolian handwriting recognition research started late, and the public datasets are relatively scarce; therefore, it is imperative to use the DA algorithm to augment the Mongolian handwriting recognition dataset.

In summary, to overcome the obstacles to the current Mongolian handwriting recognition method, such as imperfect data foundation and low prediction accuracy, this paper proposes an improved handwriting recognition model suitable for Mongolian handwriting characteristics. The main contributions of this paper are as follows: horizontal wave transformation (HWTDA) and random erasing data augmentation (REDA) algorithms were studied and used to augment Mongolian handwritten data. In model training, ECA, GRU, and Attn modules were used to encode and predict Mongolian handwriting to fully explore the local and global features of character data to improve the prediction performance of our model.

2. An Improved Dual Augmentation Algorithm

The current mainstream handwriting recognition methods, especially the methods represented by deep neural network models, need to be based on a large amount of data, and small-scale data sets cannot optimize the parameters of the training model. Research shows that when the amount of data in the dataset is small, selecting the appropriate data augmentation method can improve the efficiency of model training to a certain extent [19–22].

Random erasure data augmentation algorithm is a commonly used data augmentation method that can generate more training data by randomly erasing a part of the input image. However, this algorithm may lose some key information, resulting in the performance degradation of the model. Therefore, we modified this algorithm so that it can better adapt to the challenges related to Mongolian handwriting recognition.

The DA method should be selected based on the characteristics of the Mongolian handwritten script. First, the traditional Mongolian script consists of letters, and each letter exhibits different beginning, middle, and ending writing formats in words. Therefore, random clipping data augmentation will destroy the Mongolian format and have a negative impact on model training. Second, traditional Mongolian script follows a fixed writing rule, with letters going from top to bottom and lines running from left to right. Therefore, the samples formed using simple rotation, flipping, etc. are not in line with real ones and are difficult to function. Third, traditional Mongolian script belongs to the phonetic scripts, the characters are seamlessly connected, the position of the characters is relatively free when writing, and the phenomenon of consecutive strokes and omissions is quite common, which provides room for selecting DA algorithms. Finally, in daily life, handwritten characters often have some writing stains due to poor writing habits or low writing instrument quality, and if the model cannot improve the stain inclusiveness, it will inevitably affect its practical performance.

Based on the above practical requirements and the analysis of Mongolian handwriting recognition datasets, the augmentation of Mongolian handwriting data is achieved from the following two aspects:

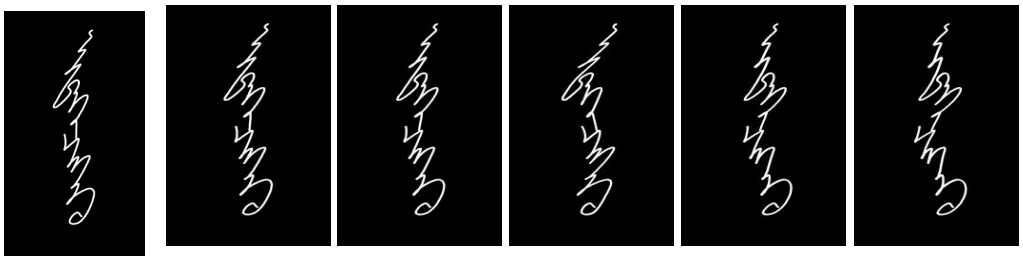
(1) From the perspective of handwriting image morphology, the HWT algorithm is used for augmenting the Mongolian handwriting data. The algorithm is softer and more flexible in processing the sample of the raw image. The obtained image samples with complete character structure follow the traditional Mongolian writing format, and the augmentation effect is diverse. Moreover, it can further enhance the details of the image and reduce the information lost due to the erasure operation, thereby enhancing the robustness of the model.

The normalized original image is used to perform horizontal wave transformation in the HWTDA algorithm, which can ensure that the parameters are universal for the image. The direction, range, and amplitude of the transformation are determined by parameter values. HWT can change pixels in an image into a horizontal sine wave shape with a given amplitude and frequency; therefore, the conversion parameters, i.e., the conversion frequency N and the magnitude R, need to be specified in advance. Each point in the image follows the transformation rules of Equations (1) and (2).

$$T(x) = x \tag{1}$$

$$T(y) = y + (\pm)R * \sin(N\pi * y) \tag{2}$$

From the above formula, HWT does not produce any distortion on the x-coordinate of a pixel. In Equation (2), the parameter R is used to control the amplitude of the sine wave, N is used to control the frequency of the transformation, $(T(x), T(y))$ is the transformation value of pixel (x, y), and (± 1) denotes the direction of the image transformation. The effect of HWTDA is shown in Figure 1.



(**a**) (**b**)

Figure 1. Experimental results of Horizontal Wave Transformation. (**a**) The original image; (**b**) the image processed by HWTDA with R = 5, 15, 25, 35, and 45 from left to right.

(2) From the perspective of image appearance, an improved random erasing data augmentation algorithm suitable for Mongolian handwritten characters is proposed in this work. The random erasure algorithm is changed to selectively erase the pixels in the image within the specified range in order to retain more important information. In addition, according to the actual application scenario, the random erasure area is set as an oval. These modifications can improve the quality of the model training data to a certain extent, thereby improving the model's performance.

First, in the existing random erasing data augmentation algorithms, the content to be erased is determined by two random values: the randomly selected point $P = (x_t, y_t)$ in the image range and the random area ratio of the region to be erased to the whole image. For the object detection task, as long as the detection subject in the image is not blocked by a large area, the detectability of the image can be maintained, but the character recognition task has its own particularities, especially for the traditional Mongolian script. Since the writing format of letters is closely related to the location in traditional Mongolian script, if

the occlusion area generated by the random erasing data augmentation algorithm covers the main part of the image to be recognized, it is difficult to accurately distinguish the text, even manually. Therefore, the hasty use of random erasing data augmentation algorithm to augment Mongolian handwritten images may be not suitable for model training and may fail. Second, the stains that appear in handwritten characters are usually caused by ink, which is generally round or elliptical, but the random erasing data augmentation algorithm is originally applied to handwritten characters with square noise, which is not in line with reality.

This article has made the following improvements to overcome the above problems. First, the selection range of random points is specified to ensure that the occlusion caused by erasing will not affect the handwriting recognition of the image. Second, in order to make the occlusion area similar to the real ones and ensure the diversity of the random erasing area, the rectangle in the existing random erasing data augmentation algorithm is replaced by an ellipse. Through the dual data augmentation algorithm, more high-quality training data can be generated to improve the performance of the Mongolian handwriting recognition model.

The flow of the improved REDA algorithm is as follows:

Step 1: Enter an input image and initialize the algorithm parameters. Let the input image be I, with a size of S, and $S = W * H$, where W and H represent the width and height of image I, respectively. Set the erasing area ratio range $[S_l, S_h]$ and the erasing major to short axis ratio range $[r_a, r_b]$ and initialize the erasing probability p.

Step 2: Select an elliptical region with a random area. Let the area of the elliptical region I_t be S_t and the ratio of the major axis a to the minor axis b be r_t. They are randomly initialized to S_t and r_t, with S_t/S within range $[S_l, S_h]$ and r_t within range $[r_a, r_b]$. The formula for calculating the major and minor axes is as follows.

$$a = 2\sqrt{\frac{S_t r_t}{\pi}} \tag{3}$$

$$b = 2\sqrt{\frac{S_t}{\pi r_t}} \tag{4}$$

Step 3: Determine the random erasing location. A location point $P = (x_t, y_t)$ in the image I is randomly generated and it satisfies the requirements $\frac{a}{2} \leq x_t \leq \alpha W$ or $(1-\alpha)W \leq x_t \leq W - \frac{a}{2}$, $\frac{b}{2} \leq y_t \leq H - \frac{b}{2}$, with $0 < \alpha < 0.5$. A parameter of α is used to ensure that the occlusion caused by erasing will not affect the recognition of Mongolian characters in the image.

Step 4: For the selected elliptical area I_t, erase its pixels according to a probabilistic random value p.

The performance of the improved REDA algorithm is shown in Figure 2.

(**a**) (**b**) (**c**)

Figure 2. Experimental test of Improved Random Erasing Data Augmentation. (**a**) The original image; (**b**) the original REDA algorithm processes images through rectangular areas; (**c**) the improved REDA algorithm processes images through elliptical areas.

This paper proposed an improved dual data augmentation (DDA) algorithm by combining the above two data augmentation methods, and its flow is presented in Figure 3.

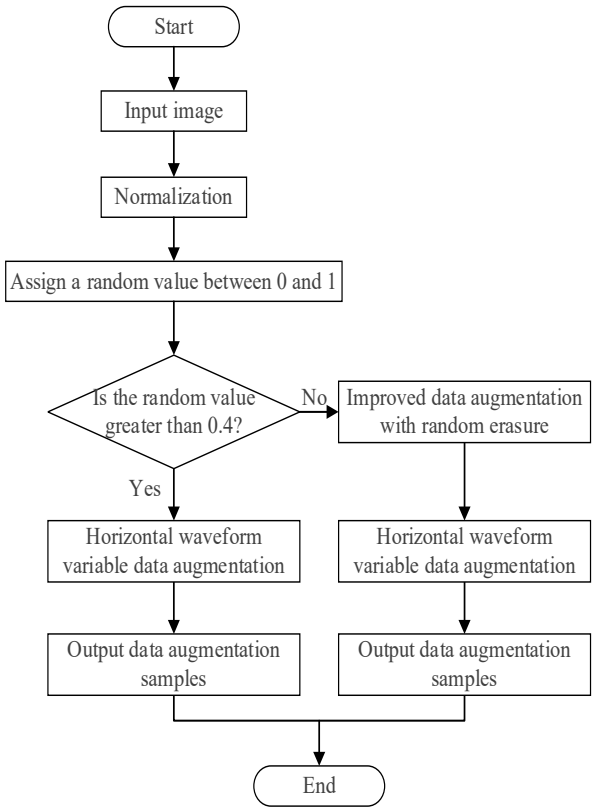


Figure 3. Flow of the Dual Data Augmentation Algorithm.

As shown in Figure 3, first, the Mongolian handwriting data are input into the model and then normalized. Second, a random value within $[0, 1]$ is assigned by the system, and if the random value is greater than 0.4, only use the HWTDA method and output data; otherwise, the DDA method is adopted, that is, the input image data are randomly erased and then, the HWTDA is performed to output the final data augmentation result. In this method, data augmentation of an image is determined by a random value, which alleviates the problem that the similarity between the randomly erased image and the original image is too high. The method of random erasing first and then using horizontal wave transform can also give richer image occlusion noise, which helps improve data richness.

3. Improved Mongolian Handwriting Recognition Model: EGA Model

CRNN can realize end-to-end character recognition and is a landmark model in the development of character recognition. However, a large number of experimental results have shown that CRNN has low recognition accuracy when performing Mongolian handwriting recognition tasks; therefore, it is difficult to be used practically. So, this paper presents an ECA-Net-GRU-Attn (EGA) Model based on the characteristics of Mongolian handwriting.

3.1. Training Process of the EGA Model

Since the proposed EGA model is constructed by improving the CRNN, it follows the CRNN network structure and consists of an ECA-Net for feature extraction, a BiGRU encoder and a BiLSTM decoder for processing feature sequences, and an attention module to enhance sequence prediction performance.

The processing flow of an input image in the EGA model is shown in Figure 4.

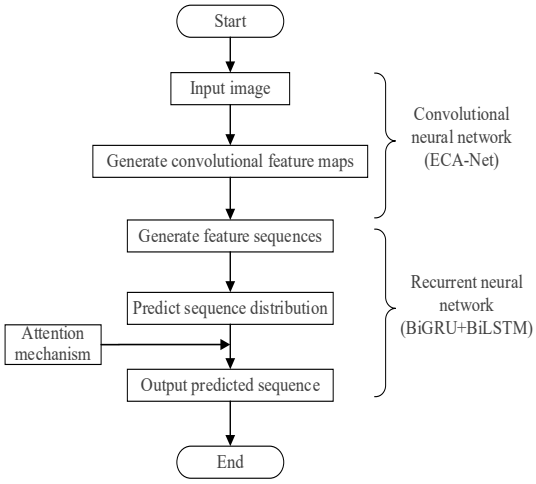
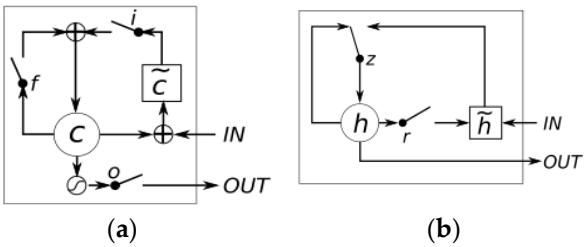


Figure 4. EGA Model Training Process.

First, compared to English handwritten script, the structure of traditional Mongolian handwritten script is more complex and flexible, which means that the feature extraction of images is more essential and difficult. The channel attention mechanism introduced in the feature extraction stage can grasp more detailed features of Mongolian handwriting images in a more directional way, improving the recognition accuracy of the model in recognizing Mongolian handwriting. In the EGA model, an ECA module, which is a channel attention module without dimensionality reduction, is utilized in the feature extraction stage of character recognition, which can overcome the shortcomings of insufficient feature extraction and insufficient processing of key areas. The ECA-Net has good experimental results in large-scale image classification, object detection, and instance ImageNet and MS segmentation [23], but we try to apply the technique in character recognition for the first time. Considering the complexity of the model, an ECA-Net model is designed by combining the ECA module with ResNet34 to extract the feature of Mongolian handwritten images.

Second, to balance the relationship between model recognition accuracy and complexity, the EGA model uses the BiGRU encoder instead of the BiLSTM encoder in the CRNN baseline model, which helps to reduce the overall complexity of the model. GRU and LSTM belong to recurrent neural networks based on the working mechanism of gating, and GRU is an evolved version of LSTM, reducing the original three gating to two. The diagrams of their structures are shown in Figures 5 and 6, respectively.



(**a**) (**b**)

Figure 5. Network LSTM and Network GRU Structure Comparison. (**a**) LSTM Network Structure; (**b**) GRU Network Structure.

From the comparison, it is clear that the structure of the GRU is significantly simpler. In addition, the EGA model performs the task of single-word recognition, and the training image used is homemade standard Mongolian handwriting recognition images by our laboratory; although data augmentation is added, the overall complexity of images is still low; therefore, GRU is fully qualified for encoding. The decoder follows the BiLSTM structure in CRNN, which has a more powerful ability to contact the context, helping to ensure the decoding quality.

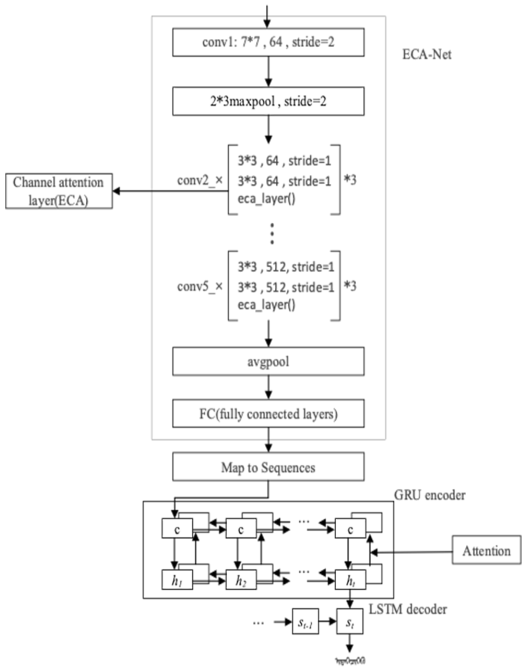


Figure 6. Network Structure of the EGA Model.

Finally, unlike English, Cyrillic Mongolian and other scripts, such as traditional Mongolian script, are in the form of a conjunction of letters from top to bottom and the beginning, middle, and ending writing formats of each letter in words are not exactly the same. Therefore, whether the context relationship of the training sample can be fully learned is crucial for the performance of traditional Mongolian handwriting recognition. The EGA model incorporates the attention mechanism into the encoder and decoder modules, which has good contextual analysis ability; therefore, introducing the attention mechanism in the output sequence prediction process can achieve further improvement of the CRNN model, which is helpful to increase the recognition accuracy of the model for Mongolian handwritten images.

3.2. Network Structure of the EGA Model

Process analysis of the EGA model can provide a more comprehensive understanding of the design concept of the model, and structural analysis of the network can deeply reveal the working principle of the model. The network structure of the EGA model is presented in Figure 6, and the way information is transmitted by each module of the EGA model can be seen more clearly through the network structure.

As shown in Figure 6, the ECA-Net is used to extract the input image features in the EGA model and the backbone network of the ECA-Net is the ResNet34, which consists of 33 convolutional layers as well as one maximum pool layer. Except for the first convolutional layer, every two convolutional layers form a residual unit and the Relu() function is executed between the residual elements as an activation function. The Mongolian handwritten input image is rotated 90 degrees to the left to form a rectangular structure with an aspect ratio of 75:40. The sliding window size of the pool layer in the ECA-Net is set to 2×3 to better collect image information.

The ECA module, which is a channel attention mechanism, can improve the model's attention to important channel features, thereby improving the generalization ability of the model on training and testing data. By considering the weights of different channels, the model can better capture the important features of Mongolian handwritten characters and improve the model's performance. Specifically, the channel attention mechanism can collect the interaction information between each channel and its k adjacent channels in a non-dimensionality-reducing manner and calculate the weight of the channel. In

the subsequent training process, the channel attention weight parameters learned by the module are represented by a matrix W_k and the expression of W_k is given in (5):

$$\begin{bmatrix} w^{1,1} & \cdots & w^{1,k} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & w^{2,2} & \cdots & w^{2,k+1} & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & w^{C,C-k+1} & \cdots & w^{C,C} \end{bmatrix} \tag{5}$$

The matrix W_k consists of $k * C$ parameters and C represents the size of the input image feature matrix, i.e., the number of output channels passing through the previous residual unit. The weight calculation formula for the image channel y_i is shown in (6):

$$\omega_i = \sigma\left(\sum_{j=1}^{k} w_i^j y_i^j\right),\ y_i^j \in \Omega_i^k \tag{6}$$

where y_i^j represents the j-th neighboring channel of y_i, w_i^j represents the weight of y_i^j, and Ω_i^k represents the set of k adjacent channels of y_i. The dynamic value of k is proportional to C, as shown in (7):

$$C = \phi(k) = 2^{(\gamma+k-b)} \tag{7}$$

The values for C and k are adaptively adjusted according to Equation (8):

$$k = \psi(C)\left|\frac{log_2(C)}{\gamma} + \frac{b}{\gamma}\right|_{odd} \tag{8}$$

where $|x|_{odd}$ represents the nearest odd number of x, and γ and b are set to be fixed constants. Due to the mapping function ψ, high-dimensional channels have a longer range of interactions, while low-dimensional channels experience a shorter range of interactions.

The ECA-Net adopts a global average pool layer at the end to connect a fully connected layer, passes the extracted Mongolian handwritten image feature map to the Map-to-Sequence layer through this fully connected layer, and converts it into a feature sequence. The feature vectors in the feature sequence are passed to the RNN as encoder input values.

The proposed EGA model has an encoder-decoder structure in the RNN. The configuration of the BiGRU encoder and the BiLSTM decoder can balance the complexity and the recognition accuracy of the Mongolian handwriting recognition model. Like the baseline CRNN model, the function of the cyclic layer in the EGA model is to predict the label distribution of each feature vector in the feature sequences, and the errors of the loop layer are backpropagated, converted into a feature sequence, and then fed back to the convolutional layer.

The input content of the GRU contains the input of the current timestep x^t and the hidden state h^{t-1} of the output of the previous timestep, and the hidden state contains the relevant information of the previous node. With x^t and h^{t-1}, the GRU can calculate the model output for the current timestep and the hidden state that will be passed to the next timestep. Specifically, this is achieved through the following steps:

(1) The gating status of the reset gate and the update gate are obtained by using h^{t-1} and x^t and are then normalized through the sigmoid function so that it acts as a gating signal. The operation of the reset gate in GRU is given in Equation (9):

$$r = \sigma\left(W_r x^t + U_r h^{t-1}\right) \tag{9}$$

where σ represents the sigmoid function and W_r and U_r are the weights learned in the reset gate. The calculation method for the update gate is shown in Equation (10):

$$z = \sigma\left(W_z x^t + U_z h^{t-1}\right) \tag{10}$$

W_z and U_z are the weights learned in the update gate.

(2) The hidden state of the current timestep t is calculated by Equation (11):

$$h^t = zh^{t-1} + (1-z)\tilde{h}^t \tag{11}$$

where $\tilde{h}^t$ contains the input of the current timestep and at the same time, the hidden state of the hidden unit of the previous timestep is added into the current hidden state; the calculation formula is shown in (12):

$$\tilde{h}^t = \phi\left(Wx^t + U\left(r \bigodot h^{t-1}\right)\right) \tag{12}$$

where W represents the weight of the current input and U represents the weight of the hidden state of the hidden unit in the previous timestep.

In the GRU, when the update gate approaches 0, the hidden state ignores the previous hidden state and updates only with the current input, which ensures that the hidden state selectively retains relevant important information and that the information can be expressed more compactly.

In summary, in the model training process, the hidden state h^{t-1} passed down from the previous timestep and the input x^t of the current timestep are used as the input of BiGRU. On one hand, it obtains the two gating states of the reset gate and the update gate and normalizes the obtained information through the sigmoid function to make it act as a gating signal. On the other hand, the image feature sequence is converted into a context vector C_t as the output.

This work introduces an attention mechanism to the encoder-decoder structure to give weights to the output of the encoder. The main implementation details of the attention mechanism are as follows: first, the input data are fed through a feedforward neural network to generate query vectors and key vectors. Then, the similarity between the query vectors and key vectors is calculated and the resulting scores are normalized to obtain the final attention weights. Finally, the attention weights are applied to the value vectors and summed up to obtain the weighted sum as the output feature vector. Through this process, the attention mechanism can learn the importance of different parts of the input data and enable the decoder to better utilize the information from the encoder. It also has strong interpretability and generalization ability.

Specifically, the attention mechanism learns a special attention weight by adding an additional feedforward neural network to the network structure α_{t_e,t_d}, where t_e and t_d represents the t-th timestep of the encoder and decoder, respectively. The introduction of this weight into the neural network helps to further reconcile the relationship between the hidden states in the encoder and decoder to highlight the focus of model training. The context vector C_t of the output of the t-th timestep encoder can be assigned by Equation (13):

$$C^t = \sum_{t_e=1}^{T} \alpha_{t_e,t_d} h_{t_e} \tag{13}$$

At the t-th timestep of the decoder, C_t is the weighted sum containing all encoder hidden states and their corresponding attention weights.

The structure of the decoder BiLSTM is similar to that of the encoder. When calculating the prediction sequence, the input values accepted by the decoder are the context vector C_t weighted by the attention mechanism, the system state S_{t-1} of the output of the decoder at the previous timestep, and the hidden state S_{t-1} of the decoder at the previous timestep. With the help of such information, the decoder can calculate the hidden state h_t of the current time node and the probability distribution of the output sequence of the node according to the probability calculation formula, as shown in (14).

$$p(S_t|S_{t-1}, S_{t-2}, \cdots, S_1, C_t) = g(h_t, S_{t-1}, C_t) \tag{14}$$

where g represents the given activation function. The softmax is used as the activation function for this stage in the EGA model. The largest output value of $p(S_t|S_{t-1}, S_{t-2}, \cdots, S_1, C_t)$ is treated as the output value of the t-th timestep. Ultimately, the output values of each timestep make up the output sequence of the model.

4. Data Experimental

4.1. Experimental Data

The dataset used in this article comes from the Inner Mongolia Normal University and includes 9479 handwritten Mongolian characters, totalling 47,395 images. The dataset covers various handwriting styles and samples from different writers, with a wide variety of fonts, and each character has a clear shape, which is highly representative and can truly reflect the features of Mongolian handwriting styles. Furthermore, the dataset is balanced, i.e., the number of images per character is similar among them. All these characters are carefully handwritten and drawn by writers, closely related to practical application scenarios and are highly authentic and practical, making them suitable for data training and testing for handwriting character recognition tasks. Some offline Mongolian handwriting sample images of the dataset are shown in Figure 7.

Figure 7. Offline Mongolian Handwriting Images.

4.2. Model Training Platform

According to the training requirements of the Mongolian handwriting recognition model, the model training environment built in this paper is shown in Table 1.

Table 1. Model Training Platform Parameters.

Name	Parameters
CPU	Intel Core i7-6700 CPU@3.40 GHz
GPU	Nvidia Tesla P100 + Huawei GPU Server
Operating system	Ubuntu 16.04.6
Programming language	Python 3.7
Deep learning framework	Pytorch 1.9.0

4.3. Indicators for Model Performance Evaluation

The quality of the Mongolian handwriting recognition model is primarily measured during the training process from two aspects: the recognition accuracy for the test set and the complexity of the trained model.

(1) Accuracy is the primary criterion to measure the effectiveness and practicality of Mongolian handwriting recognition models and it can be calculated by the confusion matrix, in which the prediction accuracy of the classification algorithm is evaluated by the relationship between the true category of the sample and the predicted value of the model, as shown in Table 2.

Table 2. Confusion Matrix.

	Forecast Category	
Real category	Positive category	Negative category
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

In the confusion matrix, TP and TN represent the situation where the predicted category matches the real category and the classification if the predicted value is correct; FP and FN represent the situation where the predicted category does not match the real category and the classification is wrong. The accuracy is defined by the proportion of correctly classified data to the total test set and calculated by Equation (15).

$$Accurary = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

In Mongolian handwriting recognition, predicted value classification is more diverse and not limited to positive and negative categories, and the correct classification represents the situation where the sequence output of the model is consistent with the sequence of image labels.

(2) Model complexity is another important criterion for measuring model quality. Under the condition of the same test platform and data, the model training time, the number of model parameters generated during the training process, and the size of the trained model are used as indicators to evaluate the model's complexity in this work.

4.4. Data Test of the Dual Data Augmentation Method and Result Analysis

The dual data augmentation algorithm combines random erasing with horizontal wave transformation, and its parameters also need to be uniformly initialized. The test parameters of dual data augmentation are listed in Table 3. The optimal hyperparameter values mentioned above were used to ensure the sample quality of the proposed data augmentation.

Table 3. Dual Data Augmentation Hyperparameter.

Hyperparameter Name	**Hyperparameter Meaning**	**Hyperparameter Value**
p	A random value that determines the image data augmentation method	[0, 1]
W	Image width	800
H	Image height	1500
S_l	For random erasing, the minimal erasing area ratio	0.01
S_h	For random erasing, the maximal erasing area ratio	0.06
r_a	The minimal ratio of the major to minor axes of the erasing area	0.5
r_b	The maximal ratio of the major to minor axes of the erasing area	2

The effect of using dual data augmentation to achieve data augmentation is shown in Figure 8. The images are, from left to right, the original image, the image processed only by HWTDA, and the image processed by DDA.

(**a**) (**b**) (c)

Figure 8. Effect of Dual Data Augmentation. (**a**) The original image; (**b**) the image processed only by HWTDA; (**c**) the image processed by DDA.

After DDA, the Mongolian handwriting recognition data have been effectively augmented, and the data before and after the augmentation are evenly mixed and randomly distributed to the training set and test set with a number ratio of 7:3 to form datasets for Mongolian handwriting recognition. The data augmentation and dataset segmentation results are summarized in Table 4.

Table 4. Dataset Segmentation after Dual Data Augmentation.

Data Type	Raw Data (Wordage × Frame Number)	Dual Data Augmentation (Wordage × Frame Number)		Total Data (Wordage × Frame Number)	Unit
		Horizontal Wave Transformation	Horizontal Wave Transformation + Random Erasing		
Training set	33,175 (6635 × 5)	33,175 (6635 × 5)	13,512 (6635 × 2)	79,862 (6635 × 12)	sheets
Test set	14,220 (2844 × 5)	14,220 (2844 × 5)	5791 (2844 × 2)	34,231 (2844 × 12)	sheets
Total	47,395 (9479 × 5)	47,395 (9479 × 5)	19,303 (9479 × 2)	114,093 (9479 × 12)	sheets

The direct purpose of designing and using dual data augmentation is to produce Mongolian handwritten images with more diverse forms and richer content, but the fundamental purpose is to deepen the understanding of Mongolian handwriting for the neural network model and improve the accuracy of Mongolian handwriting recognition through continuous learning of a large amount of data.

As shown in Figure 9, the CRNN model is trained separately by using the dataset before and after data augmentation, and the impact and significance of data augmentation can be better highlighted by comparing the training accuracy of the model. In Figure 9, the coordinate horizontal axis represents the model training epochs. The vertical axis represents the recognition accuracy (%) of the test set by the model during training.

From the accuracy curve, it can be seen that after 15 epochs of training, the recognition accuracy of the CRNN model trained using the raw dataset gradually converged to 56.2%. The growth rate of recognition accuracy of the CRNN models using data-augmentation datasets gradually slowed down but still showed an upward trend, and the recognition accuracy rate was 69.3%. The experimental results showed that DDA can help improve the recognition accuracy of the Mongolian handwriting recognition model.

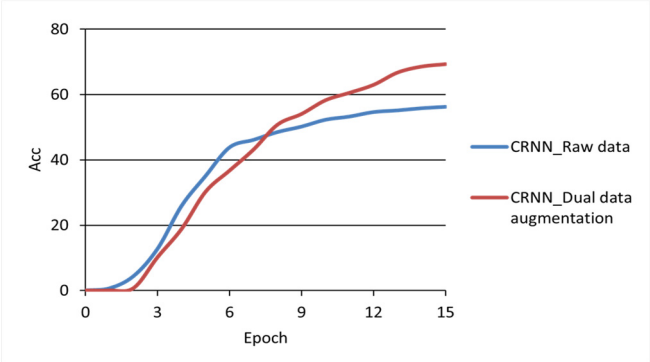


Figure 9. CRNN Model Recognition Accuracy Curve before and after Dual Data Augmentation.

This is also confirmed by the loss value curve of the model. Based on the loss values of each epoch for the test set, the model loss function curves are plotted in Figure 10. In the figure, the horizontal axis represents the model training epochs and the vertical axis represents the loss values of the model for the test set.

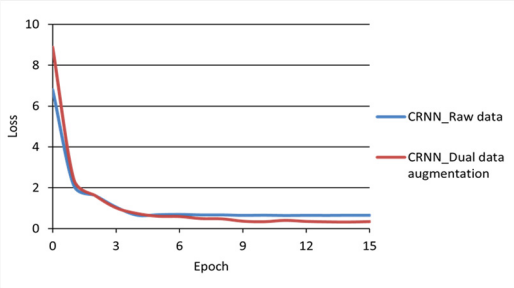


Figure 10. Model Loss Value Curve before and after Dual Data Augmentation.

It can be seen from the change trend of the loss value curve that after the model was trained for 15 epochs, the convergence speed of the CRNN model trained with the raw data and the dual data augmentation dataset gradually slowed down and the loss function converged to a lower value in the model with the dual data augmentation dataset, indicating that the dataset is more adequate for model training. The experimental results showed that the dual data augmentation effectively augmented the model training dataset and improved the recognition accuracy of the Mongolian handwriting recognition model.

4.5. Data Test for the EGA Model and Results

To ensure the training quality of the EGA model, we use the dataset augmented by DDA for model training.

The curves of the training loss value and the test loss value of the EGA model are presented in Figure 11, and the horizontal axis represents the model training epoch. The vertical axis represents the model loss values.

It can be seen that, with the increase in the number of iterations, the loss value of the training set and the test set of the model gradually decreased, and when the iteration reached about 25 epochs, the decrease in the loss value of the training set and the test set slowed down and gradually stabilized and the loss function of the test set converged to about 0.3. To prevent the model from being overfitted due to too many iterations, the training of the model was stopped.

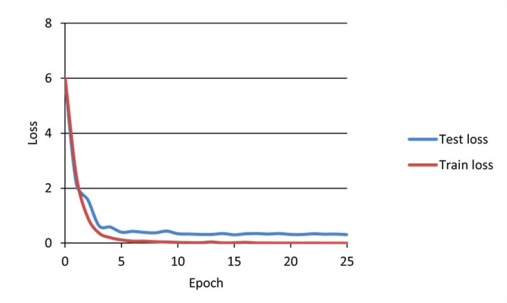


Figure 11. EGA Model Loss Value Curve.

In order to verify the improvement performance of the model, the EGA model of the experimental group and the CRNN baseline model of the control group were trained in the same experimental platform and parameter settings. Table 5 compares the performance of the EGA model and the CRNN model from the perspective of recognition accuracy and model complexity. The unit of the model scale is MB, the unit of the number of model parameters is M, the unit of accuracy is %, and the unit of the model training time is s.

Table 5. Performance Comparison Between EGA Model And CRNN.

Model	Model Size (MB)	Parameter Number (M)	Accuracy (%)	Time (s)
ECA-Net-GRU-Attn (EGA)	188.6	49.3154	89.322	215,112.2
CRNN	39.8	10.2280	74.307	196,519.8

With the recognition accuracy (%) of the model test set on the vertical axis and the model training epochs on the horizontal axis, the recognition accuracy curves of the EGA model and the CRNN model are shown in Figure 12. The recognition accuracy of the EGA model in the Mongolian handwriting recognition test set was better than that of the CRNN model, achieving the expected effect of the experiment.

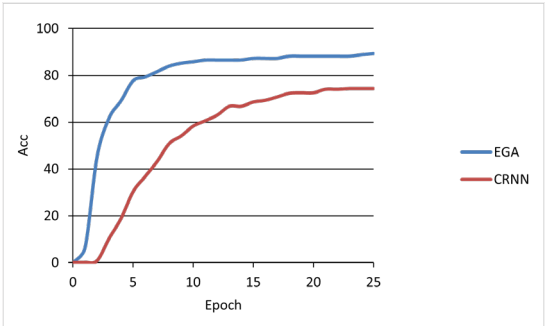


Figure 12. Comparison of Recognition Accuracy between EGA Model and CRNN Model.

From the perspective of model complexity, the size of the CRNN model and the number of parameters were small, the size of the EGA model was about 4.7 times that of the CRNN model, and the amount of parameters was about 4.8 times that of the CRNN model. In terms of training time, the training time of the CRNN model and the EGA model was 196,519 s and 215,112 s, respectively, indicating an increase of 18,592.4 s for the training time of the EGA model. It can be seen that with the increase in model depth, the model training efficiency decreases.

4.6. Comparison Test of Different Models and Result Analysis

In order to measure the impact of each module in the EGA model on performance and to demonstrate the advantages of the EGA model, comparative experiments for important modules of ECA, GRU, and Attn were conducted to test the recognition rate and training time of each model on the Mongolian handwriting dataset. The experimental results are summarized in Table 6.

Table 6. Comparison of Experimental Results of Each Model.

Model	Model Size (MB)	Parameter Number (M)	Accuracy (%)	Time (s)
ECA-Net-GRU-Attn (EGA)	188.6	49.3154	89.322	215,112.2
ResNet-GRU-Attn (RGA)	186.8	49.3153	85.496	207,709.5
ECA-Net-LSTM-Attn (ELA)	189.4	49.9728	90.846	223,211.8
ECA-Net-GRU-CTC (EGC)	184.3	48.2845	84.303	218,604.7
CRNN	39.8	10.2280	74.307	196,519.8
ResNet-GRU-Featfusion (RGF-CRNN)	115.4	30.1	85.905	215,112.7
Featfusion-ViT (F-ViT)	97.6	25.4	87.417	159,604.5

As shown in Table 6, from the perspective of the impact of the ECA module on the model recognition efficiency, the EGA model has a large scale, a large number of parameters, and a high recognition accuracy, with slightly longer model training time. Compared to the RGA model, the accuracy of Mongolian handwriting recognition was improved by 3.826%, and the model training time was increased by 7,402.7 s for the EGA model, indicating that, under the same experimental conditions, the introduction of the ECA module can improve the accuracy of handwriting recognition to a certain extent and prolong the model training time.

From the perspective of the impact of the GRU module on model recognition efficiency, compared to the ELA model, the model size and parameter amount are small, the Mongolian handwriting recognition accuracy was reduced by 1.524%, and the training time was increased by 8,099.6 s, indicating that the simplified GRU encoder module can reduce the complexity and training time of the model to a certain extent.

From the perspective of the impact of the Attn module on model recognition efficiency, compared to the EGA and EGC models, the EGA model has a larger scale and more parameters, the recognition accuracy was increased by 5.019%, and the training time was reduced by 3,492.5 s, showing that the introduction of the Attn module can improve the overall recognition efficiency of the model.

Furthermore, the experimental results in Table 6 demonstrate that the EGA model has certain advantages over three other models—CRNN, RGF-CRNN, and F-ViT—in the Mongolian handwritten characters recognition task. On the test set, the EGA model achieved the highest accuracy (89.322%) and showed significant improvement compared to the other three models. This indicates that the EGA model is more efficient at capturing the features of Mongolian characters.

In summary, introducing the ECA, GRU, and Attn modules can improve the recognition efficiency of the model to varying degrees, validating the effectiveness of the EGA model. From the perspectives of model size, parameters, accuracy, and training time, the EGA model has better recognition performance than the baseline model and performs well on data from different handwriting styles and writers, which is the main advantage of the model. However, the limitation of the model lies in its relatively larger parameter size and longer training time, which may occupy more storage space and consume more computing resources.

5. Conclusions

In this paper, Mongolian handwriting data were used as the research object, two methods were adopted to augment the Mongolian handwriting dataset, the CRNN model

and training set were used to establish a Mongolian handwriting recognition model, and the tests were conducted to evaluate the performance of the proposed model. The experimental results demonstrated that the overall accuracy of the CRNN handwriting recognition model was about 74.307% and the model training time was 196,519.8 s. In order to further improve the recognition performance of the CRNN model, we proposed an EGA model, which uses ECA-Net to realize feature extraction, a BiGRU encoder and BiLSTM decoder to process feature sequences, and an attention module to improve sequence prediction performance. The experimental results of each model were compared and analyzed on the augmented dataset. The EGA model can effectively improve the accuracy of Mongolian handwriting recognition. The size of the obtained EGA model was 188.6 MB, the number of parameters was 49.3154 M, the accuracy rate was 89.322%, and the model training time was 215,112.2 s in Mongolian handwriting recognition.

Although the performance of the Mongolian handwritten characters recognition model can be improved to some extent by using data augmentation methods and improving the network structure, the model has certain limitations due to restrictions in dataset size and structure. For example, the model has inadequate robustness for erroneous or exceptional characters and insufficient adaptability for multi-task or multi-scenario applications, and the prediction accuracy and the model performance still need to be further improved on larger datasets. In the future, we will optimize the EGA model based on the following two aspects: obtaining more data by collecting more datasets, such as introducing various data augmentation methods or considering the combined use of GANs and other generative models to generate more diverse and realistic Mongolian handwritten data, which can strengthen model training and improve recognition accuracy. In terms of model improvement, multi-task learning can be used, or more efficient feature fusion and model combination methods or training on large datasets can be adopted to enhance the model's performance and generalization ability. Using deep neural network structure to mine more valuable data information and shortening training time while maintaining the training effect is the focus of future research.

Author Contributions: Conceptualization, methodology, software, writing—original draft, Q.-D.-E.-J.R.; software, validation, writing—review, L.W.; software, writing—review, Z.M.; validation, writing—review and editing, S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (61966027), the Project of "Support Program for Young Scientific and Technological Talents" in Inner Mongolia Colleges and Universities (NJYT23059), Inner Mongolia Natural Science Foundation (2022MS06013), Inner Mongolia Science and Technology Program Project (2021GG0140), and Universities Directly Under the Autonomous Region funded by the Fundamental Research Fund Project (JY20220122).

Data Availability Statement: Data are contained within this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
2. Ciresan, D.C.; Meier, U.; Schmidhuber, J. Transfer learning for Latin and Chinese characters with deep neural networks. In Proceedings of the 2012 International Joint Conference on Neural Networks, Brisbane, Australia, 10–15 June 2012; pp. 1–6.
3. Shi, B.; Bai, X.; Yao, C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2298–2304. [CrossRef] [PubMed]
4. Tang, J.; Silamu, W.; Xu, M.; Xu, M.-M.; Xiong, L.-J.; Wang, M.-H. Uyghur scanning body recognition based on deep learning. *J. Northeast Norm. Univ. (Nat. Sci. Ed.)* **2021**, *53*, 71–76. [CrossRef]
5. Wang, D. Research and Application of Yi Online Handwriting Recognition. Master's Thesis, Southwest University, Chongqing, China, 2021.
6. Li, J. Tibetan-Chinese Bilingual Natural Scene Text Detection and Recognition System. Master's Thesis, Northwest University for Nationalities, Lanzhou, China, 2021.
7. Du, Y.; Li, C.; Guo, R.; Yin, X.; Liu, W.; Zhou, J.; Bai, Y.; Yu, Z.; Yang, Y.; Dang, Q.; et al. PP-OCR: A Practical Ultra Lightweight OCR System. *arXiv* **2020**, arXiv:2009.09941v3. [CrossRef]

8. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929. [CrossRef]
9. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 10012–10022.
10. Riaz, N.; Arbab, H.; Maqsood, A.; Nasir, K.; Ul-Hasan, A.; Shafait, F. Conv-transformer architecture for unconstrained off-line Urdu handwriting recognition. *Int. J. Doc. Anal. Recognit.* **2022**, *25*, 373–384. [CrossRef]
11. Yongping, D.; Zhuo, L. Particle Swarm Optimization-Based Convolutional Neural Network for Handwritten Chinese Character Recognition. *J. Adv. Comput. Intell. Intell. Inform.* **2023**, *27*, 165–172.
12. Fan, D.; Gao, G.; Wu, H. Research on Mongolian handwriting recognition based on morpheme segmentation. *J. Chin. Inf. Technol.* **2017**, *31*, 74–80.
13. Liu, C. Research on Recognition of Large Vocabulary Off-Line Handwritten Mongolian Whole Words. Master's Thesis, Inner Mongolia University, Hohhot, China, 2019.
14. Wei, H.; Zhang, H.; Zhang, J.; Liu, K. Multi-Task Learning Based Traditional Mongolian Words Recognition. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 1275–1281.
15. Yang, F. Research and Implementation of Mongolian Online Handwriting Recognition Based on Whole Word. Master's Thesis, Inner Mongolia University, Hohhot, China, 2021.
16. Luo, C.; Zhu, Y.; Jin, L.; Wang, Y. Learn to augment: Joint data augmentation and network optimization for text recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13743–13752.
17. Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random Erasing Data Augmentation. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 34.
18. Han, Y. Research on Data Enhanced Ancient Chinese Character Recognition Method. Master's Thesis, Xiamen University of Technology, Xiamen, China, 2021.
19. ul Sehr Zia, N.; Naeem, M.F.; Raza, S.M.K.; Khan, M.M.; Ul-Hasan, A.; Shafait, F. A convolutional recursive deep architecture for unconstrained Urdu handwriting recognition. *Neural Comput. Appl.* **2021**, *34*, 1635–1648. [CrossRef]
20. Eltay, M.; Zidouri, A.; Ahmad, I.; Elarian, Y. Generative adversarial network based adaptive data augmentation for handwritten Arabic text recognition. *PeerJ Comput. Sci.* **2022**, *8*, e861. [CrossRef] [PubMed]
21. Maalej, R.; Kherallah, M. New MDLSTM-based designs with data augmentation for offline Arabic handwriting recognition. *Multimed. Tools Appl.* **2022**, *81*, 10243–10260. [CrossRef]
22. Gao, H.; Ergu, D.; Cai, Y.; Liu, F.; Ma, B. A robust cross-ethnic digital handwriting recognition method based on deep learning. *Procedia Comput. Sci.* **2022**, *199*, 749–756. [CrossRef]
23. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. Eca-net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 11531–11539.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

YOLO-Rlepose: Improved YOLO Based on Swin Transformer and Rle-Oks Loss for Multi-Person Pose Estimation

Yi Jiang [1,†], Kexin Yang [1,*,†], Jinlin Zhu [1] and Li Qin [2]

[1] Department of Communications Engineering, Harbin University of Science and Technology, Harbin 150080, China; jasonj@hrbust.edu.cn (Y.J.); 2220600035@stu.hrbust.edu.cn (J.Z.)
[2] Department of Engineering Mechanics, Harbin University of Science and Technology, Harbin 150080, China
* Correspondence: 2220610148@stu.hrbust.edu.cn
† These authors contributed equally to this work.

Abstract: In recent years, there has been significant progress in human pose estimation, fueled by the widespread adoption of deep convolutional neural networks. However, despite these advancements, multi-person 2D pose estimation still remains highly challenging due to factors such as occlusion, noise, and non-rigid body movements. Currently, most multi-person pose estimation approaches handle joint localization and association separately. This study proposes a direct regression-based method to estimate the 2D human pose from a single image. The authors name this network YOLO-Rlepose. Compared to traditional methods, YOLO-Rlepose leverages Transformer models to better capture global dependencies between image feature blocks and preserves sufficient spatial information for keypoint detection through a multi-head self-attention mechanism. To further improve the accuracy of the YOLO-Rlepose model, this paper proposes the following enhancements. Firstly, this study introduces the C3 Module with Swin Transformer (C3STR). This module builds upon the C3 module in You Only Look Once (YOLO) by incorporating a Swin Transformer branch, enhancing the YOLO-Rlepose model's ability to capture global information and rich contextual information. Next, a novel loss function named Rle-Oks loss is proposed. The loss function facilitates the training process by learning the distributional changes through Residual Log-likelihood Estimation. To assign different weights based on the importance of different keypoints in the human body, this study introduces a weight coefficient into the loss function. The experiments proved the efficiency of the proposed YOLO-Rlepose model. On the COCO dataset, the model outperforms the previous SOTA method by 2.11% in AP.

Keywords: human pose estimation; deep learning; convolutional neural network; transformer

1. Introduction

The task of multi-person 2D keypoint detection aims to enable computers to understand all instances in an image and simultaneously identify the motion joints of each individual. Human pose estimation is one of the fundamental tasks in the field of computer vision, with extensive applications in numerous areas, including activity recognition [1], human–robot interaction [2], pedestrian tracking [3], and the re-identification of individuals [4].

Due to the various bending, stretching, or twisting movements of joints during human activities, as shown in Figure 1c,d,f, human joints may also be occluded by other objects, body parts, or themselves, as shown in Figure 1a,b,e. These factors make it challenging for pose estimation algorithms to accurately detect the positions and angles of joints. Therefore, pose estimation algorithms need to accurately capture these non-rigid variations to provide accurate pose estimation results. As a result, multi-person 2D human pose estimation is a challenging task.

Human pose estimation algorithms can be classified based on two criteria. The first criterion is the algorithm's workflow, which categorizes human pose estimation approaches

into three types: top-down, bottom-up, and single-stage pose estimation. The second criterion is the method used for predicting keypoints, which can be divided into regression-based pose estimation and heatmap-based pose estimation.

Figure 1. The challenges in multi-person 2D pose estimation are illustrated using images from the COCO dataset. (**c**,**d**,**f**) illustrate the joints bending, stretching, and twisting movements caused by human activities. (**a**,**b**,**e**) illustrate the joints may also be occluded by other objects, body parts, or themselves.

However, existing human pose estimation approaches still have some drawbacks. Firstly, top-down algorithms heavily rely on the performance of the human detector for accurate keypoint detection. If the human detector performs poorly, the accuracy of keypoint detection will also be affected. Additionally, this approach has a high computational cost [5], especially when there are a large number of human instances in the image, resulting in longer running times. Secondly, although heatmap-based methods show excellent performance, they require significant computational and storage resources, making it challenging to use them in single-stage approaches at the current stage.

In the task of object detection, the YOLO series [6] plays a significant role as a one-stage detector. In this study, the authors propose an improved model named YOLO-Rlepose based on YOLO to address the aforementioned issues. This study utilizes CSP-Darknet53 and PANet [7] as the backbone and neck of YOLO-Rlepose, respectively. In the head part, YOLO-Rlepose consists of four detection heads, each dedicated to detecting small, medium, and large objects. Each detection head includes an object box detection head, a keypoint detection head, and a convolutional layer for computing the loss. Additionally, the authors replace the original C3 module with an improved C3 module with Swin Transformer (C3STR) in the neck, enabling the network to better capture global information and rich contextual information.

This study proposes a regression-based single-stage human pose estimation method. To reduce the impact of noise on regression-based methods, a new loss function named Rle-Oks loss is introduced. The loss function simulates the distribution of the output by exploring maximum-likelihood estimation, thereby improving the accuracy of human pose regression. The contributions are described as follows:

- This study adds the Swin Transformer to the C3 module and proposes C3STR, enabling the network to better capture global information.
- This study introduces the Rle-Oks loss and applies it to human pose estimation, enabling the model to have keypoint weights when calculating the error between predicted and ground-truth keypoint values.

- On the COCO dataset, the proposed YOLO-Rlepose achieved 65.01 (AP), outperforming YOLO-Pose (previous SOTA method) by 2.11%.

2. Related Work

2.1. Multi-Person 2D Pose Estimation

The existing approaches to multi-person pose estimation can be broadly categorized into top-down methods, bottom-up methods, and single-stage human pose estimation based on their algorithmic workflows.

2.1.1. Top-Down Methods

Top-down methods first use a human detector to determine the bounding boxes of each human instance in the image and then crop these instances. Next, keypoints are detected among the obtained multiple-person instances. Representative algorithms of this kind include Hourglass [8] and CPN [9]. The Hourglass network is a sequential architecture that utilizes pooling and upsampling operations to capture different spatial relationships associated with the human body. By processing and integrating features, it aims to generate accurate predictions. This architecture is referred to as the 'stacked hourglass' network. CPN locates simple keypoints through the feature pyramid network GlobalNet and then integrates all features from GlobalNet using RefineNet to obtain the remaining keypoints. Generally, top-down methods tend to have slower inference speed during the inference process due to the need for image cropping before human keypoint detection. Additionally, the efficiency of the detection of the human body plays a crucial role in the performance of top-down methods.

2.1.2. Bottom-Up Methods

Bottom-up methods detect the keypoints in the image first and then connect the detected keypoints to form the human skeleton. Currently, most bottom-up methods are primarily based on the association of detected keypoints belonging to the same individual. Representative algorithms of this type of approach include Openpose [10] and HigherHRNet [11]. Openpose is the first real-time deep-learning-based algorithm for multi-person 2D pose estimation. It can track the facial expressions, torsos, limbs, and even fingers of individuals, making it suitable for both single-person and multi-person pose estimation with good robustness. HigherHRNet adopts a novel high-resolution feature pyramid module to generate high-resolution heatmaps, making the network more efficient. It also proposes a Multi-Resolution Supervision strategy, which assigns the training objectives of varying resolutions to their respective feature pyramid levels. Compared to top-down methods, bottom-up methods usually run faster but have lower performance due to the difficulty of grouping processing.

2.1.3. Single-Stage Human Pose Estimation

To address the drawbacks of both top-down and bottom-up methods, single-stage human pose estimation has been proposed. FCPose [5] is a fully convolutional framework for multi-person pose estimation that employs dynamic instance-aware convolution for keypoint estimation. This approach allows FCPose to eliminate the need for ROIs and post-processing after grouping. SPM [12] was proposed as the first single-stage model to improve the efficiency of multi-person pose estimation. The paper introduces a novel Structured Pose Representation, SPR, which unifies the position representation of both human instances and body keypoints, enabling the model to directly predict the poses of multiple individuals in a single stage. To adapt the network parameters to each instance, InsPose [13] introduces an instance-aware module. InsPose significantly improves the network's ability to recognize various poses. DirectPose [14] employs a novel Keypoint Alignment mechanism that overcomes the main challenge of the feature misalignment between convolutional features and predicted features in an end-to-end framework. PointSetNet [15] introduces a new object representation that can be viewed as an extension and generalization of tradi-

tional bounding boxes. YOLO-Pose [16] is a novel heatmap-free keypoint detection method based on the YOLO object detection framework. This method does not require the use of bottom-up post-processing methods to group the detected keypoints into a skeleton. This is because each bounding box is associated with a specific pose, which naturally groups the keypoints. Unlike top-down methods, instance cropping from bounding boxes is not needed since the poses of all individuals in the image are localized in a single inference.

2.2. Transformer in Vision

Transformers have been widely applied in natural language processing and have achieved significant progress in machine translation, text classification, and other tasks. Recently, many papers have attempted to introduce the Transformer architecture into computer vision tasks.

Early research focused on using Transformers as better decoders. For example, TransPose [17] introduces a Transformer model for human pose estimation, which directly processes the features extracted by a convolutional neural network to model global relationships, enabling the model to effectively capture dependencies between predicted keypoints. TokenPose [18] is a token-based approach to human pose estimation, where each keypoint is embedded with a token to estimate the position of occluded keypoints and model relationships between different keypoints. These methods are based on heatmaps and use complex Transformer encoders to enhance the model capacity. To improve the keypoint detection performance while maintaining high computational efficiency, Poseur [19] was proposed as a regression-based approach with a lightweight Transformer decoder.

Vision Transformer (ViT) demonstrated that the Transformer architecture can be directly applied to process images by treating them as a sequence of patches. ViT has achieved a performance comparable to that of convolutional networks in the field of computer vision. ViTPose [20] showcases the surprising capability of Vision Transformer in pose estimation, highlighting the simplicity of the model structure, the scalability of model size, the flexibility of training paradigms, and the portability of knowledge between models.

In computer vision tasks, the scale of input images varies significantly and is not fixed. To enable models to handle images of different scales more flexibly, the Swin Transformer [21] adopts a hierarchical structure similar to that of a convolutional neural network to process images, thereby improving computational efficiency. The hierarchical structure offers the flexibility to model at multiple scales, and its computational complexity scales linearly with the image size.

2.3. Heatmap-Based Pose Estimation and Regression-Based Pose Estimation

Currently, keypoint localization tasks can be roughly divided into two categories based on the method for predicting keypoints: heatmap-based and regression-based.

2.3.1. Heatmap-Based Pose Estimation

TompSon et al. [22] first proposed a 2D pose estimation method based on heatmaps, which represents the joint positions of the human body by estimating the per-pixel likelihood of each keypoint location. Since then, heatmap-based methods have become dominant in the field of 2D human pose estimation. To maintain high-resolution feature maps, some works have attempted to design powerful backbone networks for heatmap supervision. RSN [23] efficiently aggregates features with the same spatial size, resulting in fine-grained local representations that preserve rich spatial information at lower levels and achieve precise keypoint localization. HRNet [24] performs repeated multi-scale fusion, generating rich high-resolution representations that lead to more accurate keypoint predictions. Some works [25,26] have also focused on mitigating biases in heatmap data.

2.3.2. Regression-Based Pose Estimation

In the research of 2D human pose estimation, only a few studies have employed regression-based approaches. Although regression-based methods have faster inference

speeds, their accuracy is not as good as heatmap-based methods. Therefore, it receives less attention from researchers. Carreira et al. [27] proposed the Iterative Error Feedback (IEF) network to improve the performance of regression models. Wei et al. [15] employed Point-Set Anchors for object detection, instance segmentation, and human pose estimation. To achieve comparable performance between regression-based methods and heatmap-based methods in human pose estimation, RLE [28] was proposed as a method that utilizes maximum-likelihood estimation and flow models for keypoint prediction.

In the field of 3D human pose estimation, Rogez et al. [29] proposed an end-to-end architecture named LCR-Net, which can detect 2D and 3D poses in natural images. This network locates individuals by extracting candidate regions. Then, a classification branch is used to score each proposed pose and independently regress each anchored pose. Pavllo et al. [30] demonstrated that 3D poses in videos can be effectively estimated using a fully convolutional model with extended temporal convolutions based on 2D keypoints. In the field of human pose estimation, Wang et al. [31] proposed an innovative technique that accurately generates high-quality 3D pose information in outdoor images without being limited by internal conditions. Zeng et al. [32] introduced a segmentation and recombination method named SRNet, which can easily adapt to single-image and temporal models, and it shows significant improvement in predicting rare and unseen poses. Choi et al. [33] proposed Pose2Mesh, a novel system based on Graph Convolutional Neural Networks (GraphCNNs), which directly estimates the 3D coordinates of human mesh vertices from 2D human pose estimation. Fang et al. [34] introduced a pose grammar to address the problem of 3D human pose estimation, further improving the model's generalization ability.

3. Proposed Method

The proposed network architecture in this paper is a hybrid model named YOLO-Rlepose, which combines both convolutional and Swin Transformer models, primarily based on CSP-Darknet53. YOLO-Rlepose consists of three main parts. In the first part, the model utilizes CSP-Darknet53 as the backbone network and incorporates the C3STR module, which is a fusion with the Swin Transformer, into the backbone network to extract features more efficiently. The integration of the Swin Transformer helps enhance the feature extraction process. In the second part, the model employs the PANet structure to further refine the features. In the third part, keypoint detection heads and object box detection heads are introduced to predict keypoints and bounding boxes on four different scales of feature maps.

In addition, this study proposes a novel loss function named Rle-Oks loss. This loss function can estimate the potential distribution of errors between predicted and ground-truth keypoint values and facilitate human pose regression.

3.1. YOLO-Pose

Currently, the majority of human pose estimation approaches utilize top-down or two-stage methods. They first detect the locations of each person in the image and then perform single-person pose estimation for the detected individuals. The complexity of this top-down approach increases linearly with the number of individuals in the image. Due to its complexity and unstable runtime, it is challenging to apply this method to real-time applications.

YOLO-Pose is a human pose estimation algorithm that employs a similar approach to bottom-up methods. Unlike most bottom-up human pose estimation algorithms, it does not use heatmaps. Instead, it associates all keypoints of a person with an anchor. For a given image, the anchor associated with a person stores not only the bounding box but also the complete 2D pose keypoints. The size of the detection box is normalized based on the height and width of the anchor, but the keypoints are not constrained by the anchor's dimensions. Therefore, YOLO-Pose can still predict keypoints that are outside the bounding box, as shown in Figure 2.

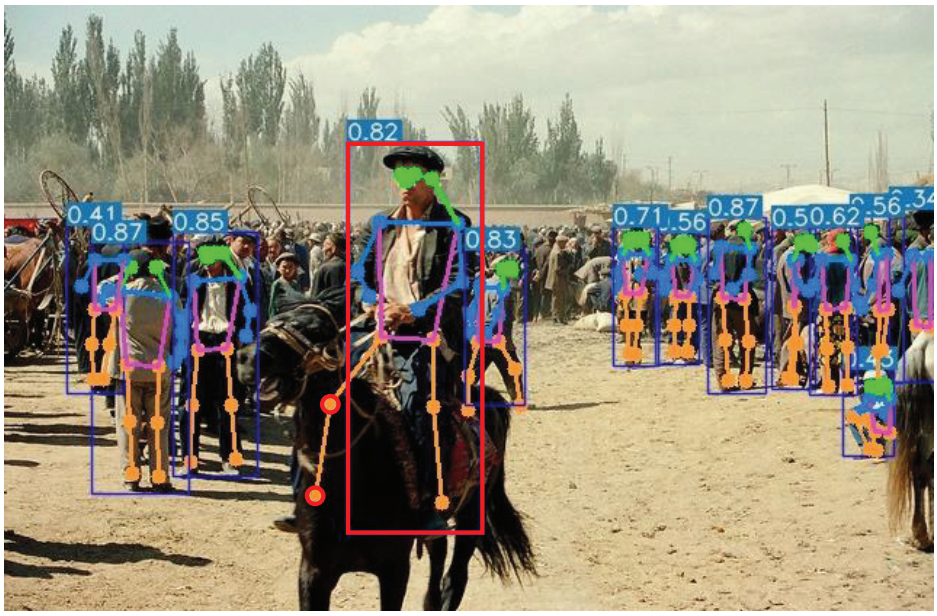


Figure 2. YOLO-Pose accurately predicts occluded keypoints that are located outside the bounding box in images from the COCO dataset.

The results in Figure 2 demonstrate the output of the human pose estimation model, including the predicted bounding boxes and the estimated human skeletons. The bounding boxes not only indicate the position of the human body but also reflect the accuracy of the algorithm's predictions. The predicted human skeletons showcase the algorithm's ability to recognize keypoints on the human body. By overlaying these two pieces of information, we can visually assess the performance of the human pose estimation model.

3.2. C3 Module with Swin Transformer

For multi-person human pose estimation in complex scenes, collecting relevant keypoint information from neighboring regions can help alleviate keypoint confusion. However, convolutional neural networks have limitations in capturing global contextual information due to the nature of convolutional operations.

Compared to convolutional operations, the Transformer model leverages multi-head self-attention mechanism to effectively capture spatial information among keypoints in an image, resulting in more accurate keypoint detection. Compared to convolutional neural networks, Transformers exhibit higher robustness to perturbations and displacements.

In existing Transformer-based models, the scale of tokens is fixed, which is not suitable for the computer vision domain. Moreover, the pixel resolution of images is much larger than the number of words in text paragraphs. In many visual tasks, such as semantic segmentation, dense predictions need to be made at the pixel level. Therefore, Transformers struggle to handle high-resolution images. To address this issue, Swin Transformer introduces hierarchical feature maps.

The Swin Transformer is an approach that replaces the standard multi-head self-attention module (MSA) with a shifted-window-based module. In the Swin Transformer, all layers remain unchanged except for the replacement of the MSA module. The structure of the Swin Transformer is illustrated in Figure 3a.

This study modified the C3 module in the original YOLOv5 architecture using the Swin Transformer module named C3STR. Compared to the original C3 module, the modified C3STR module can better capture global information and enrich contextual information. Figure 3b illustrates the structure of the C3 module.

The C3 module is an important component of the YOLOv5 network, that aims to increase the depth and receptive field of the network, thereby enhancing the feature extraction capability. The C3 module consists of three basic convolutional blocks and N Bottleneck

modules. The first convolutional block has a stride of 2, which reduces the size of the feature map by half, while the second and third convolutional blocks have a stride of 1. The convolutional blocks in the C3 module utilize 3 × 3 convolutional kernels. Additionally, Batch Normal layers and LeakyReLU activation functions are incorporated between convolutional blocks to improve the stability and generalization ability of the model.

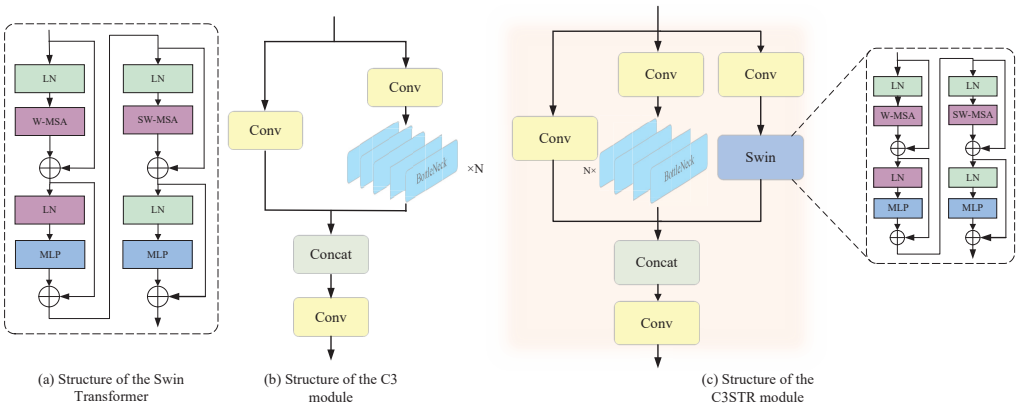


Figure 3. (**a**) W-MSA and SW-MSA are multi-head self-attention modules with regular and shifted-window configurations, respectively. (**b**) The left branch of the module consists of a single basic convolutional module, while the right branch is composed of convolutional layers and multiple Bottleneck units. The main purpose of this module is to learn residual features. Finally, the outputs from the two branches of the C3 module are concatenated together. (**c**) In addition to the C3 module, the C3STR module incorporates a Swin Transformer branch, which consists of a Swin Transformer module and three standard convolutional layers.

To enable the C3 module to gather relevant keypoint information from neighboring regions, this study integrated the Swin Transformer into the C3 module, resulting in the C3STR module. The structure of the C3STR module is illustrated in Figure 3c.

This module consists of three parallel branches, each playing a different role in feature extraction.

Firstly, the leftmost branch contains a convolutional module with the SiLU activation function. This branch focuses on directly processing the input feature map through convolutional operations to preserve the spatial information integrity.

Secondly, the middle branch has a more complex structure composed of a convolutional module and N Bottleneck modules. Each Bottleneck module employs a residual connection structure to facilitate the training of deep networks. In this module, the channel number of the input feature map is halved by a 1×1 convolutional kernel to reduce the parameter count and computational complexity. Subsequently, a 3×3 convolutional kernel is used to expand the channel number, restoring it to the original dimension. This process ensures that the channel number of the input and output feature maps remains unchanged.

The rightmost branch combines a standard convolutional module with a Swin Transformer module. The Swin Transformer utilizes a self-attention mechanism, enabling it to capture long-range dependencies, which is particularly important for understanding complex human poses.

Finally, the outputs of these three branches are merged by concatenating their feature maps along the channel dimension. Then, the concatenated feature map is processed again by a standard convolutional module to integrate the information from different branches and output the final feature representation. This design allows the C3STR module to fully leverage the advantages of different network architectures, thus achieving more accurate predictions in the task of human pose estimation.

Using the Transformer too early in the network may result in the loss of meaningful contextual information. Therefore, this study only modified a subset of the C3 modules,

specifically the ones directly connected to the head. The network architecture is illustrated in Figure 4.

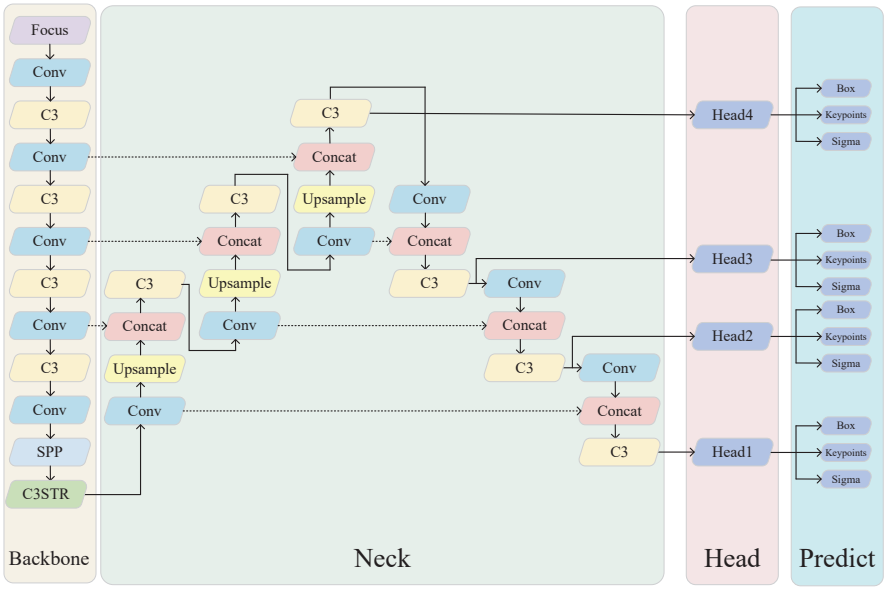


Figure 4. YOLO-Rlepose network architecture.

3.3. Rle-Oks Loss

In the field of human pose estimation, heatmap-based methods have dominated the landscape. These methods generate likelihood heatmaps for each keypoint and locate the keypoints through argmax and soft-argmax operations. While heatmap-based methods exhibit excellent performance, they come with high computational and storage requirements.

In contrast, regression-based methods for human pose estimation are more efficient but often exhibit lower performance. During human pose estimation, real-time edge devices often directly map the input to the output joint coordinates. Regression-based methods are highly effective in such cases. However, in practical applications, the presence of occlusion, motion blur, and truncation issues leads to ambiguous ground-truth labels, resulting in the poor performance of regression-based methods in these scenarios. Heatmap-based methods leverage likelihood heatmaps to make the network robust to such ambiguities, but regression-based methods are still susceptible to the influence of these noisy labels.

Therefore, to mitigate the impact of noise on regression-based methods, maximum-likelihood estimation can be employed to model the distribution of the output, thereby facilitating human pose regression [28]. The regression paradigm employed to capture the potential output distribution is referred to as Residual Log-likelihood Estimation (RLE).

A generative model, in simple terms, is a model that generates new samples that follow the same distribution as the given training data. Assuming the training data follow the distribution $p_{data}(x)$, the generated samples follow the distribution $p_{model}(x)$, and our goal is to make these two distributions, $p_{data}(x)$ and $p_{model}(x)$, similar. Therefore, the essence of a generative model is to fit the given training samples using a known probability model, which can be represented by a parameterized distribution $q_\theta(x)$.

Since different regression losses essentially make different assumptions about the output probability distribution, utilizing a flow-based model [35] to learn the distribution of errors between predicted and ground-truth keypoint values can help construct a better regression loss function.

Normalizing flows helps the model learn to construct complex distributions by transforming simple distributions through invertible mappings. Firstly, a flow model ϕ is used to map a zero-mean initial distribution $\bar{Z} \sim N(0, I)$ to a zero-mean deformed distribution $\bar{X} \sim P_\phi(\bar{X})$. Then, a regression model Θ is used to predict $\hat{\mu}$ and $\hat{\sigma}$ to control the location and scale of the distribution. The final distribution $P_{\Theta,\phi}(X \mid \tau)$ is obtained by shifting

and rescaling $\bar{X}$ to X, where $X = \bar{X} \cdot \hat{\sigma} + \hat{\mu}$. Therefore, the loss function can be written as Equation (1) [28]:

$$\begin{aligned} L &= -\log P_{\Theta,\phi} \mid_{X=\mu_g} \\ &= -\log P_\phi(\bar{\mu}_g) - \log\left|det \frac{\partial \bar{\mu}_g}{\mu_g}\right| \\ &= -\log P_\phi(\bar{\mu}_g) + \log \hat{\sigma}. \end{aligned} \tag{1}$$

Given an input image τ, the regression model predicts a distribution $P_\Theta(X \mid \tau)$, which represents the probability of the ground truth occurring at position X. In the equation, $\bar{\mu}_g = \frac{\mu_g - \hat{\mu}}{\hat{\sigma}}$ and $\partial \bar{\mu}_g / \partial \mu_g = 1/\hat{\sigma}$, where $\mu_g - \hat{\mu}$ represents the error between the predicted value and the ground-truth value, and $\hat{\sigma}$ is a constant. With this design, the flow model can now reveal the deviation between the output and the ground truth.

However, in the problem of human pose estimation, certain keypoints are more crucial than others. For instance, keypoints on the human body, such as shoulders, knees, and hips, are more important compared to keypoints on the head. Obviously, the less important keypoints are penalized more heavily for the same pixel-level error.

OKS (Object Keypoint Similarity) is the most popular evaluation metric for assessing keypoint detection. This metric is inspired by the IoU (Intersection over Union) metric used in object detection, with the purpose of measuring the similarity between the ground-truth and predicted human body keypoints. The OKS can be calculated as Equation (2):

$$OKS = \frac{\sum_{i=1}^{N_{kpts}} e^{-\frac{d_i^2}{2s^2 k_i^2}} \delta(v_i > 0)}{\sum_{i=1}^{N_{kpts}} \delta(v_i > 0)}. \tag{2}$$

In this equation, d_i represents the Euclidean distance between the ground-truth and detected keypoints, and δ is the visibility flag for the ground-truth keypoints. $s*k_i$ is the standard deviation of the Gaussian distribution; here, s is the scale parameter of the target, which represents the size or dimension of the target in the image. k_i represents the weights of the individual keypoints of the human body, and N_{kpts} represents the total number of keypoints in the human body.

The RLE loss only considers the direct difference between predicted and ground-truth values, failing to fully account for variations in human target size and the relative importance of different keypoints. This approach neglects the possibility that the distribution of keypoint errors may vary with changes in target size and keypoint importance, potentially resulting in the model's inability to accurately fit the true data distribution. To address this issue, this paper proposes a new loss function, named Rle-Oks loss, by referencing the OKS evaluation metric. This loss function not only considers the direct differences between keypoints but also normalizes these differences, similar to the OKS evaluation metric. Specifically, the Rle-Oks loss refers to the OKS evaluation metric and divides each keypoint error, denoted by $\mu_{g_i} - \hat{\mu}_i$, by a normalization factor ϵ_i determined by the target size s and keypoint-specific weight k_i. This provides a more detailed and fair measurement of the error. Through this approach, the Rle-Oks loss can more accurately reflect the error between predicted keypoints and ground-truth keypoints and helps the model better adapt to targets of different sizes and keypoints of different importance.

In the task of human pose estimation, the prediction error of each target is usually accumulated from the keypoint errors it consists of. Considering that a target may contain 17 keypoints (e.g., head, neck, shoulders, elbows, wrists, hips, knees, and ankles), this paper defines the total loss function as the sum of the losses of these 17 keypoints. Therefore, the Rle-Oks loss function is represented by Equation (3):

$$L = \sum_{i=1}^{N_{kpts}} \log(\hat{\sigma}_i \cdot \epsilon_i) - \log P_\phi(\bar{\mu}_{g_i} \cdot \epsilon_i). \tag{3}$$

where $\epsilon_i = \frac{1}{2sk_i}$, $\bar{\mu}_{g_i} = \frac{\mu_{g_i} - \hat{\mu}_i}{\hat{\sigma}_i}$, and $\hat{\sigma}_i$ is a constant.

At the beginning stage of training, the flow-based model may struggle to learn the correct distribution of errors, leading to increased difficulty in training the regression model and potentially decreasing the model's performance. To address this issue, the target probability distribution formula can be written as Equation (4):

$$\log P_\phi(\bar{X}) = \log Q(\bar{X}) + \log G_\phi(\bar{X}) + \log s. \tag{4}$$

where the first term $Q(\bar{X})$ represents a simple distribution (such as a Gaussian or Laplace distribution), $G_\phi(\bar{X})$ is the distribution that the flow model needs to learn, and the third term is a constant. Through this transformation, it can be observed that the dependence of the regression model on the flow model is greatly reduced, as the learned results of the flow model are only used to complement the simple distribution.

The presence of occlusions poses a challenge for keypoint detection tasks, as occlusions can prevent algorithms from accurately identifying the position and pose of the target. Therefore, the Rle-Oks loss takes into account the occlusions of keypoints, which can improve the accuracy and robustness of keypoint detection.

4. Experiments

The proposed model was evaluated on the COCO dataset. The COCO dataset consists of over 200,000 images with a total of 250,000 human instances, each annotated with 17 keypoints. The training set contains 57,000 images, while the validation and test sets contain 5000 and 20,000 images, respectively. The proposed model was trained on the training set, and the performance was evaluated on the test set. To evaluate the performance, this study uses the Average Precision (AP) based on the Object Keypoint Similarity (OKS) as the main evaluation metric on the COCO dataset. The experimental results are reported as the Average Precision (AP), Average Recall score (AR), and other metrics, such as AP50, AP75, and APL at different thresholds and target sizes.

In the field of human pose estimation, Average Precision (AP) is a crucial performance evaluation metric. The calculation of AP involves a threshold value, denoted by t, which is used to determine whether a keypoint detection is considered correct. Specifically, OKS is employed to measure the similarity between the predicted keypoints and the ground-truth keypoints. For a given threshold value t, if the OKS of a predicted keypoint is greater than t, it is regarded as a successful detection. Conversely, if the OKS is less than t, the detection is considered unsuccessful, indicating potential false positives, false negatives, or other errors. For each threshold value t, we count the number of detections with an OKS greater than the threshold and compare it to the total number of detections, resulting in a ratio. This ratio represents the proportion of correctly detected keypoints by the model at the current threshold, with a higher value indicating better detection performance. AP provides a comprehensive measure by aggregating the performance evaluations at different thresholds.

During the training process, to ensure the consistency of the input images, we first resized them to the desired size while maintaining a fixed aspect ratio. To maintain the consistency of the aspect ratio, we performed padding operations at the bottom of the image. This ensures that each sample has the same size and aspect ratio when performing pose estimation on the input images. The SGD optimizer with a cosine scheduler was used in training.

4.1. Implementation Details

YOLO-Rlepose was implemented with the help of PyTorch 1.11. All models were trained and tested on NVIDIA GeForce RTX 2080 Ti. The device is a graphics card introduced by NVIDIA, located in Santa Clara, California, United States. During the training phase, partially pre-trained models from YOLO-Pose were utilized to accelerate the training progress. Since YOLO-Rlepose shares a significant portion of the backbone and some parts

of the head with YOLO-Pose, many weights can be transferred, which significantly saves training time.

During the training phase, both the regression model and the flow model are optimized simultaneously. The initial density distribution of the flow model is set to a Laplace distribution. During inference, there is no need to run the flow model, allowing the loss function to be flexibly applied to various regression algorithms without increasing the testing time. Since the flow model is lightweight, it has a minimal impact on the training speed. Due to the generality of this loss function, it can also be applied to other versions of YOLO.

4.2. Results on COCO Test Set

Table 1 compares the proposed model with other methods based on Higher, HRNet, and EfficientHRNet. The proposed model demonstrates competitive performance in terms of AP values compared to existing methods. The achievements in AP values are particularly significant.

Table 1. Comparison with bottom-up methods on the COCO dataset.

Method	Backbone	Size	AP	AP50	AP75	APL	AR
Openpose [10]	–	–	61.8	84.9	67.5	68.2	66.5
Hourglass [8]	Hourglass	512	56.6	81.8	61.8	67.0	–
PersonLab [36]	ResNet-152	1401	66.5	88.0	72.6	72.3	71.0
PiPaf [37]	–	–	66.7	–	–	72.9	–
HRHet [24]	HRNet-W32	512	64.1	86.3	70.4	73.9	–
EfficientHRNet-H_0 [38]	EfficientNetB0 [39]	512	64.0	–	–	–	–
EfficientHRNet-H_0 † [38]	EfficientNetB0 [39]	512	67.1	–	–	–	–
HigherHRNet [11]	HRNet-W32	512	66.4	87.5	72.8	74.2	–
HigherHRNet [11]	HRNet-W48	640	68.4	88.2	75.1	74.2	–
HigherHRNet † [11]	HRNet-W48	640	70.5	89.3	77.2	75.8	–
DEKR [40]	HRNet-W32	512	67.3	87.9	74.1	76.1	72.4
DEKR [40]	HRNet-W48	640	70.0	89.4	77.3	76.9	75.4
YOLOv5s6-Pose [16]	Darknet-csp-d53-s	960	62.9	87.7	69.4	71.8	69.8
YOLOv5m6-Pose [16]	Darknet-csp-d53-m	960	66.6	89.8	73.8	75.2	73.4
YOLOv5l6-Pose [16]	Darknet-csp-d53-l	960	68.5	90.3	74.8	76.5	75.0
YOLOv5s6-Rlepose	Darknet-csp-d53-s	960	65.0	87.8	71.2	71.7	71.5
YOLOv5m6-Rlepose	Darknet-csp-d53-m	960	67.6	89.2	74.5	73.8	74.0

† Indicates multi-scale testing.

One of the reasons why the proposed model performs well in terms of AP values is the adoption of a novel loss function and network module, which enables the network to better understand the spatial relationships between keypoints. Through this approach, the model gains a better understanding of the structure and relationships of human poses.

Furthermore, the proposed model exhibits superiority in predicting keypoints that are partially occluded. As shown in Figure 5, the model accurately predicts partially occluded keypoints, which is crucial for practical applications where occlusions may occur.

Experiments indicate that the proposed model is competitive in keypoint detection tasks and achieves notable results in terms of AP values and predicting keypoints in occluded body parts. The authors believe that the proposed model can play an important role in various practical applications, such as human action analysis and human–computer interaction.

Figure 5. YOLO-Rlepose accurately predicts partially occluded keypoints in images from the COCO dataset.

4.3. Comparison of Rle-Oks Loss with Rle Loss and OKS Loss

One of the main contributions of this model is the Rle-Oks loss. The advantage of the Rle-Oks loss lies in its consideration of both the spatial relationships between keypoints and the weights between keypoints. In contrast, the traditional OKS loss only considers the distance between keypoints, while the Rle-Oks loss fails to take into account the visibility of keypoints and their weights. Therefore, the Rle-Oks loss can more accurately reflect the error in keypoint localization and better capture the distribution of real errors.

In the experiments, the authors compared the performance of OKS loss, Rle loss, and Rle-Oks loss in the yolov5s6-rlepose model. The results in Table 2 clearly show that the Rle-Oks loss, with the inclusion of keypoint weighting coefficients, exhibited better performance. This indicates that the Rle-Oks loss can more accurately estimate the localization error of keypoints and provide more precise loss feedback, thereby facilitating model training and optimization.

The experiments demonstrate that the Rle-Oks loss is an effective loss function that can model the error distribution better and provide more accurate loss estimation. Using the Rle-Oks loss in keypoint localization tasks can lead to improved performance, thereby enhancing the accuracy and stability of the model.

Table 2. Comparison of different loss functions. It can be observed that the Rle-Oks loss performs the best in terms of AP.

Method	Loss	Size	AP	AP50	AP75
YOLOv5s6-Rlepose	OKS	960	62.9	87.7	69.4
YOLOv5s6-Rlepose	Rle Loss	960	55.24	83.26	60.93
YOLOv5s6-Rlepose	Rle-Oks Loss	960	64.02	87.47	70.96

4.4. Ablation Experiments

In this section, the impact of different components on the network performance is evaluated, as shown in Table 3. This study is based on the baseline model YOLOv5s6-Pose, where YOLOv5s6-Rlepose was incorporated as the backbone network, and the C3STR module was integrated into the original network. Additionally, the original loss function was replaced with the Rle-Oks loss.

To assess the influence of different components on the network performance, we conducted a series of experiments. Firstly, we compared the performance of the baseline model YOLOv5s6-Pose with the modified YOLOv5s6-Rlepose, which includes the Rle loss, on the COCO dataset. The results showed that the modified model did not perform well in

the human pose estimation task, indicating that the inclusion of the Rle-Oks loss did not yield desirable prediction results.

Next, this study further investigated the impact of the Rle-Oks loss on the network performance. The experimental results demonstrated that the network using the Rle-Oks loss achieved improvements in AP and AP75, confirming the effectiveness of the Rle-Oks loss in human pose estimation tasks.

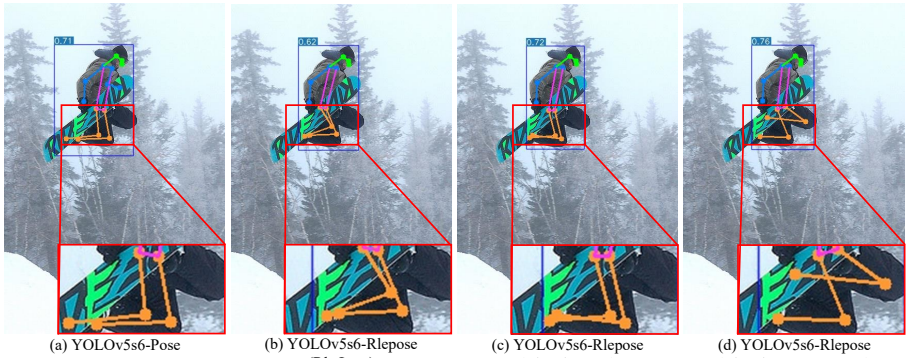
Furthermore, the effects of the combination of the Rle-Oks loss and the C3STR module were studied on the experimental results. The results showed that the simultaneous use of the Rle-Oks loss and the C3STR module exhibited better robustness and accuracy in the human pose estimation task. Compared to the baseline model YOLOv5s6-Pose on the COCO dataset, the AP value achieved the greatest improvement, with a gain of 2.11%.

The experimental results demonstrate significant performance improvements in the human pose estimation task by incorporating the C3STR module and the Rle-Oks loss to enhance the baseline model. These findings indicate the crucial roles played by the C3STR module and the Rle-Oks loss in human pose estimation tasks.

Table 3. Ablation experiments on the COCO dataset.

Method	AP	AP50	AP75	AR
YOLOv5s6-Pose	62.9	87.7	69.4	71.8
YOLOv5s6-Rlepose (Rle Loss)	60.66	85.61	66.68	67.27
YOLOv5s6-Rlepose (Rle-Oks Loss)	64.02	87.47	70.96	70.40
YOLOv5s6-Rlepose (Rle-Oks Loss+C3STR)	65.01	87.84	71.20	71.54

Figure 6 presents the results of the ablation experiments, as recorded in Table 3. In this experiment, the proposed YOLO-Rlepose model achieved a performance improvement of 2.11% compared to the YOLO-Pose model. Although this improvement may not be very apparent in the visual results, a careful zoom-in analysis reveals an enhancement in the accuracy of keypoint predictions.



(a) YOLOv5s6-Pose (b) YOLOv5s6-Rlepose (Rle Loss) (c) YOLOv5s6-Rlepose (Rle-Oks Loss) (d) YOLOv5s6-Rlepose (Rle-Oks Loss+C3STR)

Figure 6. Comparison of results from ablation experiments. The red border indicates the enlarged section.

Although the model in this study has achieved improvements in accuracy, it must be noted that this performance enhancement comes at the cost of increased computational complexity. The detailed numbers are given in Table 4.

It should be clarified that the loss function only affects the model during the training process. During this stage, the loss function evaluates the difference between the ground-truth values and the predicted values of the model and adjusts the model's parameters through backpropagation to minimize this difference. Once the model training is completed, the loss function no longer plays a role. Therefore, the Rle-Oks loss does not impact the computational complexity during the inference stage.

Table 4. Comparing the parameter and computation complexities of various experiments.

Method	#Param.(M)	GFLOPs
YOLOv5s6-Pose	15.09	20.2
YOLOv5s6-Rlepose (Rle Loss)	15.09	20.2
YOLOv5s6-Rlepose (Rle-Oks Loss)	15.09	20.2
YOLOv5s6-Rlepose (Rle-Oks Loss+C3STR)	16.0	23.7

5. Conclusions

This study incorporated several state-of-the-art techniques, such as the Swin Transformer and an improved flow-based loss function, into YOLOv5 to create an effective human keypoint detector named YOLO-Rlepose. The results of our approach on the COCO dataset are shown in Figure 7. The detector excels at detecting keypoints for small objects. The introduction of the C3STR module allows for the better capture of global information and richer contextual information, thereby effectively improving the model's keypoint detection capability. Additionally, the adoption of Rle-Oks loss reduces the impact of noise on regression-based methods, resulting in the more accurate detection of keypoints by the network. The experiments demonstrate that YOLO-Rlepose achieves favorable performance on the COCO dataset. The authors hope that this report can assist developers and researchers in gaining better insights into and experiences in the analysis and processing of human pose estimation.

Figure 7. Results of human pose estimation by YOLO-Rlepose on the COCO dataset. The method demonstrates excellent keypoint detection performance for small objects.

Author Contributions: Methodology, Y.J.; software, K.Y.; writing—original draft, K.Y. and J.Z.; visualization, L.Q.; supervision, Y.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: https://cocodataset.org/.

Acknowledgments: The authors give their thanks the editor and anonymous referees for their comments, which helped improve the quality of this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Du, Y.; Wang, W.; Wang, L. Hierarchical recurrent neural network for skeleton based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 1110–1118.
2. Jain, H.P.; Subramanian, A.; Das, S.; Mittal, A. Real-time upper-body human pose estimation using a depth camera. In Proceedings of the Computer Vision/Computer Graphics Collaboration Techniques: 5th International Conference, MIRAGE 2011, Rocquencourt, France, 10–11 October 2011; Proceedings 5; Springer: Berlin/Heidelberg, Germany, 2011; pp. 227–238.
3. Andriluka, M.; Iqbal, U.; Insafutdinov, E.; Pishchulin, L.; Milan, A.; Gall, J.; Schiele, B. Posetrack: A benchmark for human pose estimation and tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 19–21 June 2018; pp. 5167–5176.
4. Liu, J.; Ni, B.; Yan, Y.; Zhou, P.; Cheng, S.; Hu, J. Pose transferrable person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 19–21 June 2018; pp. 4099–4108.
5. Mao, W.; Tian, Z.; Wang, X.; Shen, C. Fcpose: Fully convolutional multi-person pose estimation with dynamic instance-aware convolutions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9034–9043.
6. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
7. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 19–21 June 2018; pp. 8759–8768.
8. Newell, A.; Yang, K.; Deng, J. Stacked hourglass networks for human pose estimation. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part VIII 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 483–499.
9. Chen, Y.; Wang, Z.; Peng, Y.; Zhang, Z.; Yu, G.; Sun, J. Cascaded pyramid network for multi-person pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 19–21 June 2018; pp. 7103–7112.
10. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
11. Cheng, B.; Xiao, B.; Wang, J.; Shi, H.; Huang, T.S.; Zhang, L. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In Proceedings of the IEEE/CVF Conference on computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5386–5395.
12. Nie, X.; Feng, J.; Zhang, J.; Yan, S. Single-stage multi-person pose machines. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6951–6960.
13. Shi, D.; Wei, X.; Yu, X.; Tan, W.; Ren, Y.; Pu, S. Inspose: Instance-aware networks for single-stage multi-person pose estimation. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual, 20–24 October 2021; pp. 3079–3087.
14. Tian, Z.; Chen, H.; Shen, C. Directpose: Direct end-to-end multi-person pose estimation. *arXiv* **2019**, arXiv:1911.07451.
15. Wei, F.; Sun, X.; Li, H.; Wang, J.; Lin, S. Point-set anchors for object detection, instance segmentation and pose estimation. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part X 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 527–544.
16. Maji, D.; Nagori, S.; Mathew, M.; Poddar, D. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 2637–2646.
17. Yang, S.; Quan, Z.; Nie, M.; Yang, W. Transpose: Keypoint localization via transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 11802–11812.
18. Li, Y.; Zhang, S.; Wang, Z.; Yang, S.; Yang, W.; Xia, S.T.; Zhou, E. Tokenpose: Learning keypoint tokens for human pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 11313–11322.
19. Mao, W.; Ge, Y.; Shen, C.; Tian, Z.; Wang, X.; Wang, Z.; den Hengel, A.v. Poseur: Direct human pose regression with transformers. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 72–88.
20. Xu, Y.; Zhang, J.; Zhang, Q.; Tao, D. Vitpose: Simple vision transformer baselines for human pose estimation. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 38571–38584.
21. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
22. Tompson, J.J.; Jain, A.; LeCun, Y.; Bregler, C. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in Neural Information Processing Systems, Proceedings of the NIPS 2014, Montreal, QC, Canada, 8–13 December 2014*; Neural Information Processing Systems Foundation, Inc. (NeurIPS): San Diego, CA, USA, 2014; Volume 27.
23. Cai, Y.; Wang, Z.; Luo, Z.; Yin, B.; Du, A.; Wang, H.; Zhang, X.; Zhou, X.; Zhou, E.; Sun, J. Learning delicate local representations for multi-person pose estimation. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part III 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 455–472.

24. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5693–5703.
25. Sun, X.; Shang, J.; Liang, S.; Wei, Y. Compositional human pose regression. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2602–2611.
26. Zhang, F.; Zhu, X.; Dai, H.; Ye, M.; Zhu, C. Distribution-aware coordinate representation for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 7093–7102.
27. Carreira, J.; Agrawal, P.; Fragkiadaki, K.; Malik, J. Human pose estimation with iterative error feedback. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4733–4742.
28. Li, J.; Bian, S.; Zeng, A.; Wang, C.; Pang, B.; Liu, W.; Lu, C. Human pose regression with residual log-likelihood estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 11025–11034.
29. Rogez, G.; Weinzaepfel, P.; Schmid, C. Lcr-net: Localization-classification-regression for human pose. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3433–3441.
30. Pavllo, D.; Feichtenhofer, C.; Grangier, D.; Auli, M. 3D human pose estimation in video with temporal convolutions and semi-supervised training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7753–7762.
31. Wang, L.; Chen, Y.; Guo, Z.; Qian, K.; Lin, M.; Li, H.; Ren, J.S. Generalizing monocular 3D human pose estimation in the wild. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019; pp. 4024–4033.
32. Zeng, A.; Sun, X.; Huang, F.; Liu, M.; Xu, Q.; Lin, S. Srnet: Improving generalization in 3D human pose estimation with a split-and-recombine approach. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XIV 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 507–523.
33. Choi, H.; Moon, G.; Lee, K.M. Pose2mesh: Graph convolutional network for 3D human pose and mesh recovery from a 2D human pose. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part VII 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 769–787.
34. Fang, H.S.; Xu, Y.; Wang, W.; Liu, X.; Zhu, S.C. Learning pose grammar to encode human body configuration for 3D pose estimation. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
35. Rezende, D.; Mohamed, S. Variational inference with normalizing flows. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; PMLR; pp. 1530–1538.
36. Papandreou, G.; Zhu, T.; Chen, L.C.; Gidaris, S.; Tompson, J.; Murphy, K. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 269–286.
37. Kreiss, S.; Bertoni, L.; Alahi, A. Pifpaf: Composite fields for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11977–11986.
38. Neff, C.; Sheth, A.; Furgurson, S.; Tabkhi, H. Efficienthrnet: Efficient scaling for lightweight high-resolution multi-person pose estimation. *arXiv* **2020**, arXiv:2007.08090.
39. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; PMLR; pp. 6105–6114.
40. Geng, Z.; Sun, K.; Xiao, B.; Zhang, Z.; Wang, J. Bottom-up human pose estimation via disentangled keypoint regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14676–14686.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Detection of Small Lesions on Grape Leaves Based on Improved YOLOv7

Mingji Yang [1], Xinbo Tong [1] and Haisong Chen [2,*]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; yangmingji@hrbust.edu.cn (M.Y.); 2120600050@stu.hrbust.edu.cn (X.T.)
[2] School of Undergraduate Education, Shenzhen Polytechnic University, Shenzhen 518055, China
* Correspondence: hschen@szpt.edu.cn

Abstract: The precise detection of small lesions on grape leaves is beneficial for early detection of diseases. In response to the high missed detection rate of small target diseases on grape leaves, this paper adds a new prediction branch and combines an improved channel attention mechanism and an improved E-ELAN (Extended-Efficient Long-range Attention Network) to propose an improved algorithm for the YOLOv7 (You Only Look Once version 7) model. Firstly, to address the issue of low resolution for small targets, a new detection head is added to detect smaller targets. Secondly, in order to increase the feature extraction ability of E-ELAN components in YOLOv7 for small targets, the asymmetric convolution is introduced into E-ELAN to replace the original 3×3 convolution in E-ELAN network to achieve multi-scale feature extraction. Then, to address the issue of insufficient extraction of information from small targets in YOLOv7, a channel attention mechanism was introduced and improved to enhance the network's sensitivity to small-scale targets. Finally, the CIoU (Complete Intersection over Union) in the original YOLOv7 network model was replaced with SIoU (Structured Intersection over Union) to optimize the loss function and enhance the network's localization ability. In order to verify the effectiveness of the improved YOLOv7 algorithm, three common grape leaf diseases were selected as detection objects to create a dataset for experiments. The results show that the average accuracy of the algorithm proposed in this paper is 2.7% higher than the original YOLOv7 algorithm, reaching 93.5%.

Keywords: YOLOv7; small target detection; E-ELAN network; attention mechanism

1. Introduction

The United States Department of Agriculture (USDA) released the "World Fresh apple, grape and pear market and Trade Report", which shows that the 2022/2023 season global table grape production is expected to increase by 1.1 million tons to 27.3 million tons, achieving the fourth consecutive year of growth; China's grape production will reach 12.6 million tons, the world's largest grape producer [1]. At the same time, the output of the European Union and Turkey has increased significantly. However, Australia suffered heavy rains in the spring, and the attack of bacteria on grapes was extremely serious, affecting the yield of grapes. The wet conditions of South African grapes during the growing season have increased the pressure of fungal diseases, which have had a significant negative impact on the harvest, while the peak heat in December and January has brought sunburn to the grapes, and the lack of irrigation power during the high summer temperatures has resulted in crop losses [2].

When grapes grow, they are often eroded by diseases, which not only reduce the fruit and yield of grapes but even make the vineyard suffer a devastating blow in serious cases. Because grapes have important nutritional, medicinal and economic value, improving the efficiency of prevention and control of grape leaf diseases is the key to improving fruit

quality and yield [3]. In the process of grape growth, leaf diseases are mostly small targets. Affected by the growth time, grape leaf disease spots have problems such as random locations and different sizes of disease spots. Some disease spots have small pixels, which makes it easy to miss inspection and leads to difficulties in manual identification. Therefore, accurately and efficiently identifying the types of grape leaf diseases and providing corresponding prevention and control strategies is of great significance for the development of the grape industry [4].

At present, grape leaf diseases are still mainly observed manually. Relevant personnel need to observe the disease spots on the leaves with the naked eye and determine whether it is a specific disease based on their own experience. The accuracy of disease types depends on the experience of relevant personnel. In addition, some of the diseased areas on grape leaves are very small, which not only increases the difficulty of detection but also greatly affects the discrimination ability of personnel, leading to an increase in the rate of false detection rate [5,6].

In recent years, the research and application of artificial intelligence technology have been in full swing, especially in the field of modern agricultural precision planting, which has aroused strong interest from many researchers. The use of artificial intelligence technology for the intelligent identification of crop diseases has also become an important research hotspot. The crops involved are diverse, such as grapes, tomatoes, potatoes and corn. The types of methods involved are also diverse, mainly divided into two categories: traditional machine learning methods and deep learning methods [7].

The application of machine learning theory and technology in crop pest identification began in the 1980s and 1990s. In 2005, Tian Youwen [8] proposed a method for identifying grape diseases by applying SVM and chromaticity moment analysis, which had good classification and generalization ability for small sample datasets, and its recognition rate of black spot, powdery mildew and downy mildew is more than 90%. In 2007, Youwen [8] used SVM to identify grape leaf diseases with an average recognition rate of 87.5%. In 2016, Mohan [9] used the Scale Invariant Feature Transform (SIFT) algorithm to extract color features of rice leaf blasts, brown spots and bacterial wilt. Finally, SVM and K-nearest neighbor algorithm (KNN) were used for classification, and a rice disease recognition method with an accuracy of 91.10% was proposed. Padol [10] used SVM classification technology to detect and classify grape leaf diseases. The method first found the lesion area through K-means clustering segmentation, then extracted the color and texture features, and finally used SVM to classify the leaf disease. Waghmare [11] proposed a grape leaf disease recognition system based on multi-class support vector machines, and the classification accuracy of this method for grape leaf disease is 96.66%. In 2019, Jaisakthi [12] used the global threshold method to extract the diseased parts of grape leaves and then classified the grape leaf images through the support vector machine, obtaining an accuracy of 93.04% on the test images.

The above methods are all crop disease recognition methods based on traditional machine learning technology. Compared to early manual recognition methods, their recognition accuracy and efficiency have been significantly improved. However, these methods have problems such as low efficiency, weak anti-interference ability and poor universality, which seriously limits their promotion in actual agricultural production. With the development of deep learning technology, convolutional neural network (CNN) replaces the traditional artificial feature extraction method, which has a stronger feature expression ability and significantly improves the accuracy and efficiency of plant disease recognition [5].

CNN is the most widely used method in deep learning image classification technology and is favored by many plant disease recognition researchers; it can automatically extract image features to classify and recognize leaf disease types in images. In 2016, Mohanty [13] proposed a model based on the integration of AlexNet and GoogleNet to detect 26 kinds of diseases in the Plant Village dataset, with a detection accuracy of 99.35%. In 2017, Ramcharan [14] proposed an automated cassava disease identification method

based on the Inceptionv3 network, which used transfer learning to train CNN, and the classification accuracy reached 93%. Bin Liu [15] designed a new deep convolutional neural network model based on AlexNet, which used apple leaf disease images for pre-processing to generate new datasets, and the accuracy rate reached 97.62% in the datasets of four common apple leaf diseases. In 2018, Liu [16] proposed a CNN-based recognition model for strawberry leaf powdery mildew disease. The CNN structure with nine different network depths was designed by this method, and the nine models were trained and tested with different proportions of training sets and test sets. The results showed that the average accuracy reached 98.61%. In 2019, Too, E. C. [17] used VGGNet, Inception V4, ResNet and DenseNet to detect the apple disease dataset and found that DenseNet performed best, with a test accuracy of 99.75%. In 2020, Gong An [18] proposed using multiple CNNS to identify diseases. The experimental results showed that the recognition accuracy of the fused model is 87.19%. Long Yin [19] used ResNet50 and other networks as the basic framework to train and identify crop disease data. Experimental results showed that the improved model has the highest recognition accuracy of 90.2%.

With the rise of CNN-based target detection methods, these methods can not only identify the type of leaf disease but also locate the location of the disease. In 2017, Fuentes [20] proposed a method based on deep learning target detection to identify and locate tomato diseases. In this method, Faster R-CNN, R-FCN and SSD were used to detect tomato diseases and insect pests. The results show that R-FCN has the highest mAP (mean average precision) of 85.98%. In 2018, Adhikari [21] proposed a tomato disease detection method based on YOLO, which used Raspberry Pi for disease image acquisition and recognition; the deployment of the model was realized, and its mAP reached 76%. Liu Tanyu [22] first removed the background in the grape disease image to reduce the interference of the disease region and adopted fast-RCNN technology to extract leaf disease spots, which made grape leaves adaptable for detection. For common grape diseases, the average mAP of this method was 75.52%. Xin [23] proposed a real-time grape detection model using the YOLOv3 model as the basic framework. The model replaced the Darknet-53 backbone network in YOLOv3 with the EfficientNet network to effectively balance the image resolution and the depth and width of the training network, and the final recognition rate reached 97.29%. In 2019, Jiang [24] expanded the apple leaf disease dataset through data enhancement and combined Inception structure in GoogleNet and Rainbow concatenation with SSD network, and the experiment showed that the mAP reached 78.80%. Its detection speed is 23.13 fps. Zhao [25] adopted the YOLO network to identify tomato diseases, and the experimental results show that the mAP of this method is 97.24%. In 2020, He Xin [26] proposed a method for identifying grape leaf diseases. The method first improved the pooling and mask branch in mask R-CNN. Then, the SENet attention module was integrated into the ResNet50 classification network to form a new multi-feature fusion classification network, and the recognition accuracy was 90.83%. Zhu Xu [27] proposed a blueberry canopy fruit detection and recognition method based on Faster R-CNN. By constructing their own dataset and using improved Faster R-CNN for training, the average recognition accuracy of the obtained model could reach 94%. In 2021, He Zifen [28] used the SE attention mechanism and asymmetric mixing convolutional module in Mask R-CNN to detect three diseases of apple and obtained an average segmentation accuracy of 94.7%, reducing the training memory. In 2022, Wang Zhuo [29] proposed a real-time apple detection model, YOLOV4-CA. In this model, MobileNet v3 was used as the backbone feature extraction network, and deep separable convolution was introduced into the feature fusion network with an average accuracy of 92.23%. Huang Tongbin [30] artificially realized the recognition of citrus fruits in the natural environment by introducing the attentional mechanism module (CBAM) to improve the feature extraction capability of the network and proposed a citrus recognition method based on the improved YOLOv5 model. The test results show that the average accuracy of the model reaches 91.3%.

In 2023, YOLOv7 rose strongly, and many researchers used YOLOv7 to detect items and achieved good results. Zhao Zhenbing [31] proposed an anti-vibration hammer

corrosion detection algorithm based on improved YOLOv7. The method first carried out color space transformation through the HSV color model to highlight corrosion features, then introduced multi-scale feature fusion, and made full use of feature maps under different scales to obtain weighted feature fusion. It can accelerate the inference speed and improve the accuracy of the model. The test results show that the detection accuracy of the improved algorithm is increased by 1.6%, and the inference speed is increased by 17%. Jialiang [32] proposed the detection of dragon fruit based on YOLOv7, and the accuracy, recall rate and average accuracy were 84.4%, 92.4% and 93.2%, respectively. LiangLiang Z [33] proposed a multi-scale UAV aerial image target detection model MS-YOLOv7 based on YOLOv7, aiming at the common problems of a large number of targets and a high proportion of small targets in UAV aerial images. The new network uses multiple detection heads and CBAM convolutional attention module to extract features of different scales, and the average detection accuracy of MS-YOLOv7 is increased by 4.9% compared with the YOLOv7 object detection algorithm.

With the progress of computing technology, as well as the remarkable innovation and breakthrough of image detection of crop leaf diseases and related technologies, many scholars at home and abroad have used Faster R-CNN, SSD and other algorithms to detect common disease spots in crop leaf diseases images, and the average accuracy rate has reached over 90%, which provides new possibilities for effective management of crop leaf diseases. However, the existing detection methods still have some shortcomings, such as low accuracy and slow detection speed for small target diseases of plant leaves. Therefore, it is worth exploring and studying to detect small target lesions on grape leaves quickly and accurately by using the advantages of deep convolution neural networks. This paper improved on the original YOLOv7 model and added an improved channel attention mechanism. After training, a grape disease detection model is obtained. The main contributions of the proposed method are summarized as follows:

1. In response to the problem of low accuracy in identifying small targets in the original YOLOv7 model, a specialized output prediction branch was added to detect small defect targets and perform feature fusion.
2. In order to increase the feature extraction ability of E-ELAN, asymmetric convolutional ACNet was introduced into E-ELAN to enhance its ability to extract small target features.
3. In response to the insufficient information extraction for small target lesions on grape leaves using the existing YOLOv7 algorithm, an improved CA attention mechanism, which can effectively identify small lesions, was added.
4. The SIoU loss function was introduced to enhance the localization ability of the network, thereby improving the detection accuracy of the network and reducing false or missed detections of small targets during the detection process.

2. Analysis of the Characteristics of Grape Leaves with Diseases

The main diseases of grape leaves are blight, black rot and black measles, as shown in Figure 1. Their lesions mainly have the following characteristics:

1. The grape leaf blight affects the normal metabolism of the leaves and causes the grape leaves to fall off, leading to a reduction in grape yield. The grape leaf blight disease mainly harms the grape leaves. According to the area of the disease spot, it is generally divided into two kinds: big blight and small blight. The edge of the big blight is dark brown, and the shape is generally polygonal or quasi-circular. The small blight first produces small yellow-green spots on the grape leaves, and then the spots gradually increase. The shape is circular, and the color changes from dark brown to light brown as the lesion increases, as shown in Figure 1a.
2. The black rot disease spot of grape leaves shows a black circular outline in the leaves. With the expansion of the disease spot, the center of the disease spot becomes gray-white, and then the edge becomes black, as shown in Figure 1b.

3. At the initial stage of grape leaf black measles, red-brown irregular spots appear first and then expand into round spots with a diameter of about 2 cm. The front of the lesion has obvious concentric rings of the same depth and color, while the back of the lesion is covered with a light brown layer of mold during humid weather, as shown in Figure 1c.
4. Even under a single condition, the size of the lesions is diverse, and the same lesion may even show different sizes due to factors such as angle and image size.

(**a**)

(**b**)

(**c**)

Figure 1. The examples of the grape leaves with diseases:(**a**) Grape leaves with blight disease; (**b**) Grape leaves with black rot diseasel; (**c**) Grape leaves with black measles disease.

According to the characteristics of the above-mentioned grape leaf disease images, the target shape and scale of grape leaf disease are changeable, which makes it very difficult for YOLOv7 to detect small disease spots because YOLOv7 mainly relies on convolution operation in feature extraction. For small-scale disease spots, due to their limited pixel area, many details may be lost after convolution processing, which makes it difficult to extract effective features. And because the positioning accuracy of YOLOv7 depends on the size of its receptive field, for small-scale targets, the receptive field may not be enough to cover the whole target, resulting in inaccurate positioning.

3. Improvement of YOLOv7 Model

YOLO series algorithms are known for their fast speed, small model size and simple structure, so they are widely used in industrial production and life. From the performance comparison of the YOLOv1 to YOLOv4 algorithm, the comprehensive performance of the algorithm is improved with each new version. Although the detection accuracy of YOLOv4 is lower than that of YOLOv3, the speed has been greatly improved, which is very suitable for real-time target detection. YOLOv5 integrates some characteristics of YOLOv3 and YOLOv4, and the speed is faster. By improving the accuracy of the detection algorithm, YOLOv7 makes the detected target more accurate and faster than the previous version and can process a large number of images in a short time. This paper chooses to optimize the YOLOv7 model to improve the performance of the algorithm further.

As shown in Figure 2, YOLOv7 adopts the backbone network composed of BConv module, E-ELAN module and MP module for downsampling and feature extraction, and adopts SPPCPSC module to expand the receptive field. Then, three feature maps of different scales are input into the feature fusion module, which is composed of the CatELAN module, UpSample module and MP module. After the feature fusion module, Cat is used to fuse the context information, the output is re-parameterized through the REPConv layer, and the output tensor is finally obtained. YOLOv7 introduces model reparameterization into the network architecture to reduce the calculation amount of the model and introduces label allocation strategy and efficient E-ELAN network architecture to form a model feature extraction module. Finally, YOLOv7 puts forward the training method of the auxiliary head, which increases the training cost and improves the accuracy without affecting the reasoning time.

The YOLOv7 algorithm uses a top-down and bottom-up bidirectional fusion network to aggregate features at different levels. It uses shallow features to distinguish simple targets and deep features to distinguish complex targets, and it enhances the semantic information of the entire pyramid by transferring high-level strong semantic features. However, the detection effect of YOLOv7 on small blade targets is poor, mainly because the resolution of small targets is low, and they are susceptible to interference from noise, complex background and other factors, resulting in more difficult detection of small targets. In view of the above shortcomings, the detection performance of YOLOv7 is mainly improved from the following aspects:

(1) To solve the problem of YOLOv7′s insufficient accuracy in identifying minor lesions, this paper adds a detection layer dedicated to detecting small targets and involves the new feature layer in the feature fusion process. Through this scheme, the network can detect more small targets with lower resolution.
(2) Aiming at the problems of low resolution and insufficient information for small targets, asymmetric convolution ACNet is introduced into E-ELAN to further enhance the ability to extract features.
(3) To solve the problem of image noise, background and other factors interfering with the accuracy of grape leaf lesions, this paper adopts an improved CA attention mechanism to strengthen key features and suppress the interference of noise, which can improve the model's attention to small targets.
(4) Aiming at the problem of poor positioning ability of CIoU loss function, SIoU is introduced in the original YOLOv7 network model to optimize the loss function, reduce the degree of freedom of the loss function and enhance the positioning ability of the network, thus improving the network detection accuracy and reducing the missed detection of small targets in the detection process.

The improved YOLOv7 model is shown in Figure 2. In the figure, the red part represents the Input section of YOLOv7; The yellow section represents the Backbone section of YOLOv7; The purple part is the Neck part of YOLOv7; The light blue part is the Head part of YOLOv7; The dark blue section represents the specific components of each module of YOLOv7; The green section is an added module based on YOLOv7.

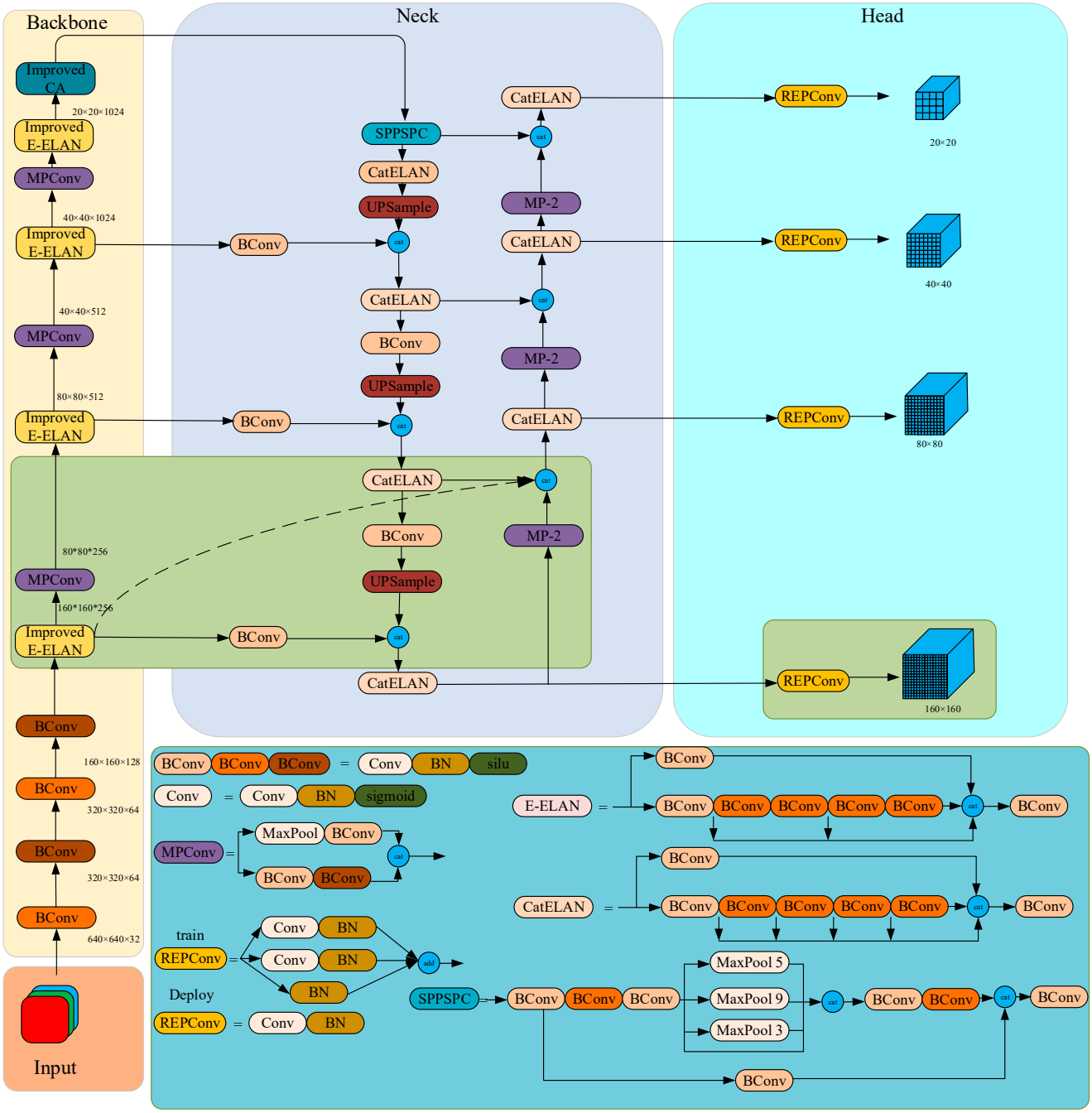


Figure 2. Schematic diagram of improved algorithm of YOLOv7.

3.1. Improvement of Feature Fusion Network

When using a convolutional neural network to extract image features, with the deepening of convolution layers, the resolution of the feature map decreases gradually, and the range of the local receptive field to perceive the original image expands continuously. Moreover, the feature map closer to the top tends to pay attention to the global information of the image, so the deep features extracted by the deep neural network are very unfavorable for small target detection. Although the shallow feature map has a small local receptive field and contains more position and detailed information, which can make up for the deficiency of deep features to some extent, if only shallow features are used to detect targets, it will cause serious false detection and missed detection due to the lack of guidance of high-level semantic information. Thus, shallow features and deep features are fused, and then the object position is predicted on the fused feature maps of different scales.

In YOLOv7, if the size of the input image is 640 × 640 × 3, the feature maps of 80 × 80 × 255, 40 × 40 × 255 and 20 × 20 × 255 are obtained after 8×, 16× and 32× downsampling, respectively, and the network finally detects the target on these three feature maps with different scales. Among the feature maps of these three scales, the smallest local receptive field is the eight times downsampled feature map; if the feature map is mapped to the original input image, each pixel corresponds to the 8 × 8 area of the original image. For the target with low resolution, the receptive field of the feature map obtained by eight times downsampling is still too large, and it is easy to lose the position and detail information of some small targets, which leads to its poor performance on the small target dataset. In order to improve the current situation of missed detection, we optimized the structure of YOLOv7 and added a new prediction branch. After three times of up-sampling, the size of the detection layer corresponding to the new feature layer is 160 × 160, which can be used to detect defect targets with a size of more than 4 × 4 and can effectively improve the detection effect of small lesions.

The feature fusion structure used in the YOLOv7 algorithm is PANet, which aggregates different levels of features through a top-down and bottom-up bidirectional fusion network. In order to integrate more shallow information into the feature enhancement part, some information extracted from the Backbone is also integrated into the FPN process to improve the recognition accuracy of the model. In the process of feature enhancement, in order to minimize the information loss caused by the deepening of the network and the increase in convolution modules, the original model continuously upsamples and downsamples the loaded information, which undoubtedly loses much detailed information and reduces the accuracy of the model in detecting small targets. On the basis of the previous improvement, the modules on the Backbone are connected by leaps to retain more detailed information and reduce the loss of information. The improved feature fusion results are shown in Figure 3. The red arrow represents upsampling, the blue arrow represents downsampling, the middle two boxes are YOLOv7's FPN module and PAN module, the last box is the detection layer part, and the red small box is the newly added P2 detection layer on the original basis.

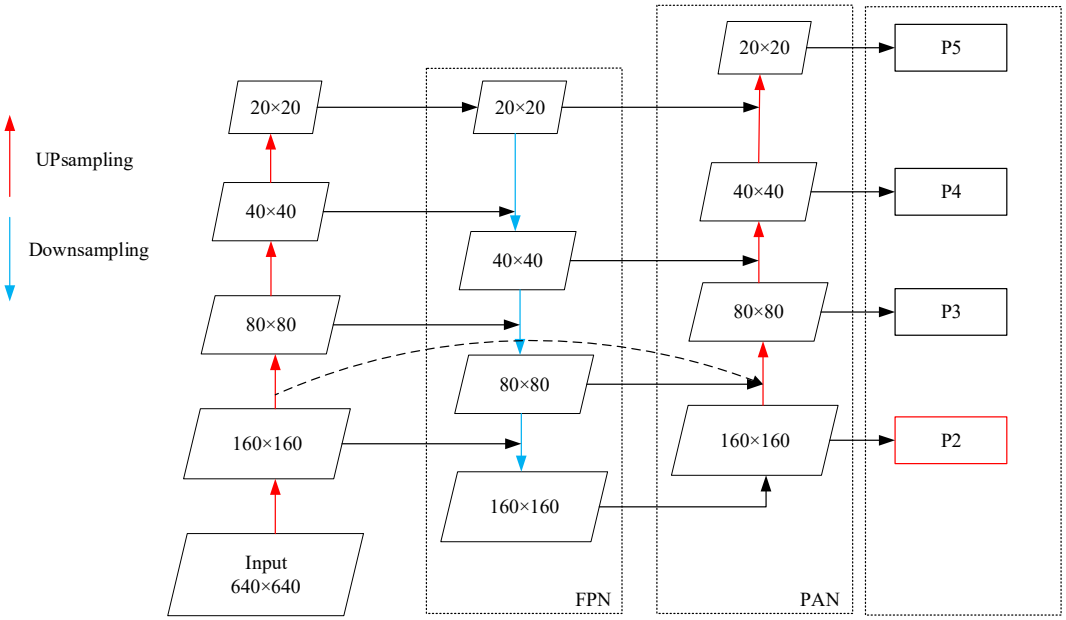


Figure 3. FPN + PAN structure after adding detection layer P2.

In summary, on the one hand, this structure increases the depth of the network so that the network can better learn the multi-level feature information of the target and enhance the network's ability to detect multi-scale targets in complex environments. On the other hand, it can transfer more shallow features to deep features so that the network can obtain more small target information and improve the detection ability of the model for small targets.

3.2. Improvement Based on E-ELAN

E-ELAN in YOLOv7 is improved based on ELAN. Compared with the ELAN network, E-ELAN adds the use of group convolution to increase the cardinality of features. This structure can enhance the feature extraction ability, improve the use of parameters and calculation efficiency and improve the learning ability of the network without destroying the original gradient path. E-ELAN has a good feature extraction ability when extracting the information of major lesions but does not carry out much calculation when extracting the information of small targets. As a result, the detection cannot guarantee the complete accuracy of the lesion category.

The attention mechanism inside the E-ELAN module can help to understand the decision-making process of the model, but it may not provide a completely accurate explanation or deal with complex scenarios. Thus, this paper introduces asymmetric convolution ACNet [34]. The core idea of ACNet is to decompose the original convolution, which sums up the outputs of two convolutional branches with horizontal and vertical kernels, respectively, thereby reducing the number of parameters and computations while ensuring comparable accuracy. During the training phase, the feature extraction layer adopts a parallel structure of 3×3 conventional convolution kernels and asymmetric convolution kernels for feature extraction and then combines the extracted feature maps with channels to improve the richness of semantic information within the feature maps. In the detection stage, in order to operate the model using convolutional kernels with consistent parameters while minimizing computational complexity, the parallel convolutional kernels in the AC module are fused to obtain their equivalent convolutional kernels, and the convolutional kernels are used for feature extraction. Moreover, when the distribution parameters of the box convolution are uneven, the weight of the center cross position is greater, further enhancing the expression ability.

ACNet uses three parallel $d \times d$, $1 \times d$ and $d \times 1$ cores to replace the original $d \times d$ core, as shown in Figure 4 (let d = 3):

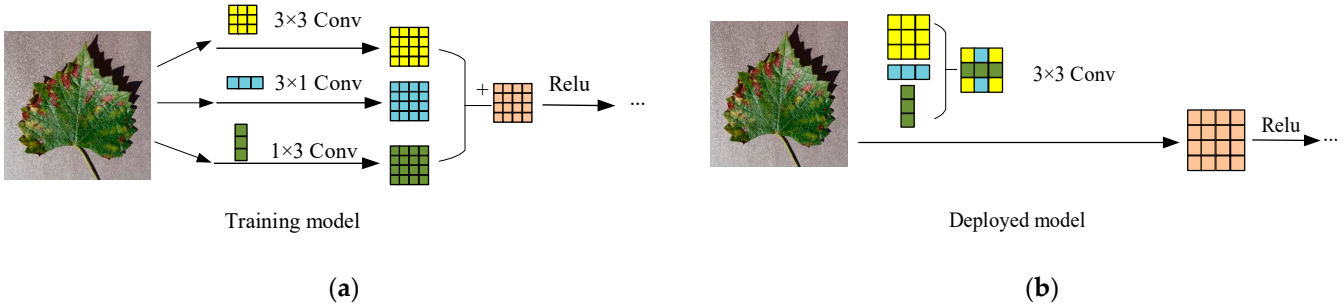


Figure 4. Schematic diagram of AC structure training and deployed process: (**a**) ACNet training model; (**b**) ACNet deployed model.

ACNet is divided into the training stage and the reasoning stage. In the training phase, ACNet replaces every 3×3 convolution in the existing network with 1×3, 3×1 and 3×3 convolution and finally fuses the calculation results of these three convolution layers to obtain the output of the convolution layer. In the reasoning stage, ACNet mainly fuses three convolution kernels. In this part, the fused convolution kernel parameters are used to initialize the existing network; in the reasoning stage, the network structure is exactly the

same as the original network, except that the network parameters adopt the parameters with stronger feature extraction ability, that is, the fused convolution kernel parameters.

The E-ELAN network in YOLOv7 is improved, and shortcut connection, 1×3 and 3×1 convolution structures are added between ELAB modules to enrich the features extracted from the network, as shown in Figure 5. The module with a plus sign in the circle is used to add up the weights of the branches

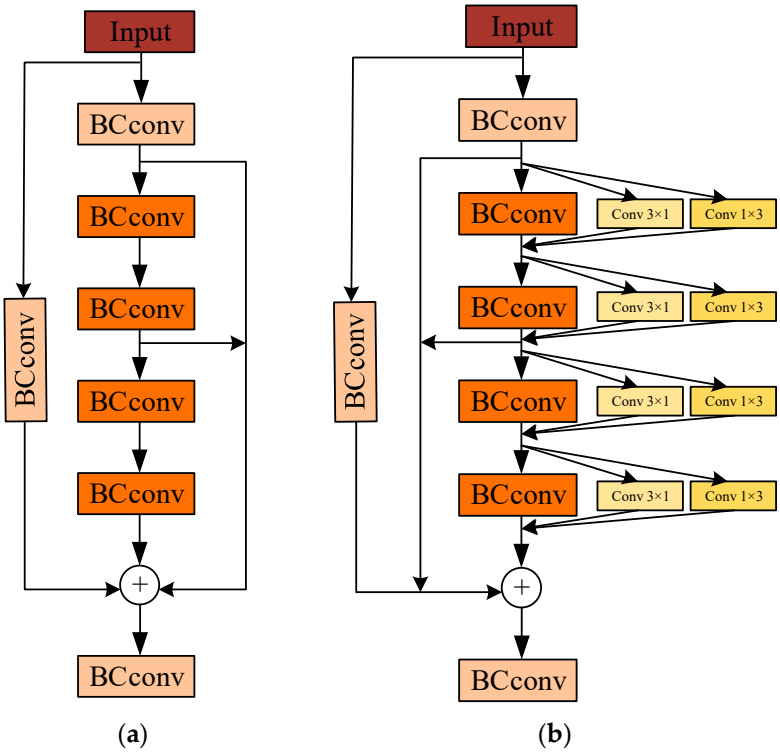


Figure 5. E-ELAN model and improved E-ELAN model: (**a**)E-ELAN model; (**b**) improved E-ELAN model.

E-ELAN in YOLOv7 is improved, and 3×1 and 1×3 convolution structures are added between modules to make the network extract more abundant features to improve the detection accuracy of the model.

3.3. Improvement Combined with Channel Attention

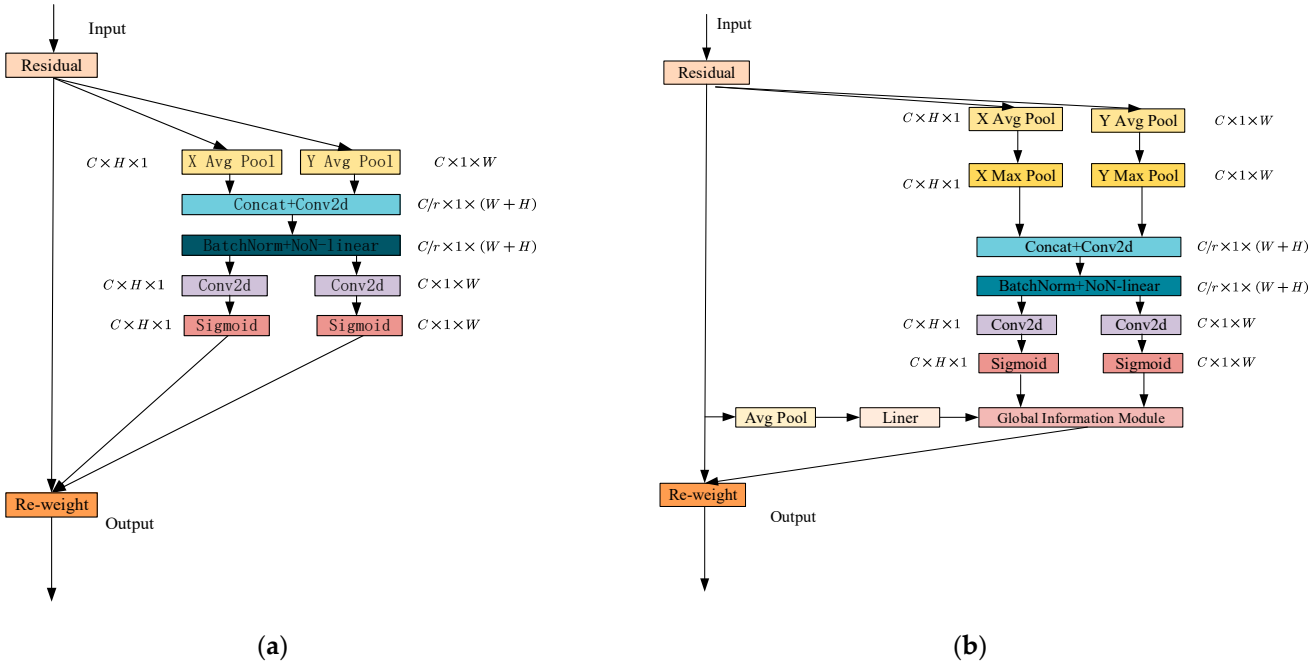
The attention mechanism establishes dynamic weight parameters by making relevant and irrelevant choices on information features to strengthen important information and weaken useless information. Among them, the Squeeze and Exclusion (SE) [35] channel attention mechanism is widely used for its simplicity and high efficiency. However, the attention mechanism of the SE channel ignores the importance of location information. In the Convective Block Attention Module (CBAM) [36], GAP (global average pooling) and GMP (global maximum pooling) are combined to improve performance, thereby increasing the efficiency of deep learning algorithms and improving some defects of traditional deep learning, but it needs more computing resources and has higher computational complexity. Coordinate attention (CA) [37] is a flexible and efficient attention mechanism that can not only obtain cross-channel information but also obtain location information and help the model to locate and identify the target of interest more accurately without introducing much computational overhead. The CA module completes the weight distribution of feature information through two stages: coordinate information embedding and coordinate attention generation. CA attention mechanism performs GAP operation on the input feature map in the height and width directions to obtain the feature map in two directions. However, the simplicity of GAP makes it difficult to capture the complex information of

various inputs. The existing CA channel attention mechanism divides the input feature map into horizontal and vertical parts and calculates the corresponding weights, respectively. However, this method cannot balance the global information and leads to the neglect of important spatial information.

In order to overcome the limitations of the existing CA attention module, this paper makes the following improvements:

(1) Add the GMP pathway to the GAP pathway of the original CA attention mechanism. The input feature map contains many targets with different sizes, so it is a better choice to average the pool first and then maximize the pool. This is because average pooling can capture all the information in the feature map, smooth the image and reduce noise better. Then, the maximum pooling is carried out, which can emphasize the most prominent features of the target while retaining the size transformation and further extracting effective information about the target.

(2) Calculate the weight by processing the input feature map. Adaptive average pooling and full connection layer operation are performed on the input feature map to generate a two-dimensional Softmax vector, which is used to represent the mixed proportion of horizontal and vertical attention weights. This method can effectively balance the global and local information, as shown in Figure 6.



(**a**) (**b**)

Figure 6. CA attention and improved CA attention: (**a**) CA attention; (**b**) improved CA attention.

3.4. Loss Function

The loss function in the YOLOv7 network model is shown in Formula (1):

$$L_{object,Loss} = L_{los,Loss} + L_{conf,Loss} + L_{class,Loss} \quad (1)$$

where $L_{loc,Loss}$ represents positioning loss, $L_{conf,Loss}$ represents confidence loss and $L_{class,Loss}$ represents classification loss. The coordinate loss is calculated using CIoU [38]. In order to describe the aspect ratio of the real frame and the predicted frame, when the aspect ratio is the same, the penalty item value is 0, which means that the difference in the aspect ratio is calculated as the penalty item, but it cannot represent the real difference between the length and width. Different from the CIoU loss function, the SIoU [39] loss function calculates the relationship of length and width of the real box and the prediction box separately, rather than calculating the difference of the aspect ratio, avoiding the loss of penalty terms when

the aspect ratio of the prediction box and the real box is equal, and ensuring the stability of convergence, as shown in Figure 7.

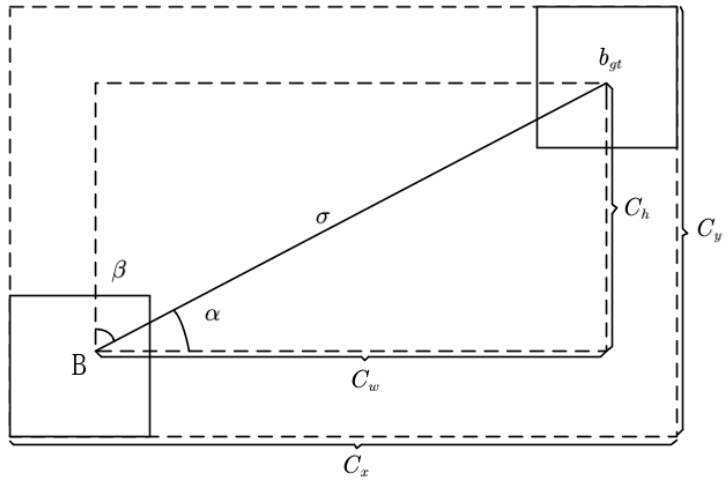


Figure 7. SIoU loss function model.

c_x and c_y is the length and width of the real frame, c_w and c_h represent the difference in width and height between the center point of the target box and the prediction box, B is the prediction frame, b_{gt} is the real frame, and σ represents the distance between the center point of the target box and the prediction box.

In order to solve the direction angle regression problem of the target box, the SIoU loss function introduces angle perception, distance loss and shape loss on the basis of the original IoU to improve the speed and accuracy of reasoning during the training of the detection model.

The angle perception added by SIoU first predicts whether the angle is greater than 45° and determines the use of minimization β or α. The calculation of angle cost is shown in Equation (2):

$$\Lambda = 1 - 2 \times \sin^2\left(\arcsin(x) - \frac{\pi}{4}\right) \tag{2}$$

where

$$x = \frac{c_h}{\sigma} = \sin(\alpha) \tag{3}$$

$$\sigma = \sqrt{\left(b_{c_x}^{gt} - b_{c_x}\right)^2 - \left(b_{c_y}^{gt} - b_{c_y}\right)^2} \tag{4}$$

$$c_h = \max\left(b_{c_y}^{gt} - b_{c_y}\right) - \min\left(b_{c_y}^{gt} - b_{c_y}\right) \tag{5}$$

$$c_w = \max\left(b_{c_x}^{gt} - b_{c_x}\right) - \min\left(b_{c_x}^{gt} - b_{c_x}\right) \tag{6}$$

In the formula, $b_{c_x}^{gt}$ and $b_{c_y}^{gt}$ represent the coordinates of the center point of the target box; b_{c_x} and b_{c_y} represent the coordinates of the center point of the prediction box. The distance cost represents the distance between the center points of the predicted box and the actual box. Based on the angle cost defined above, SIoU redefines distance cost as follows.

$$\Delta = \sum_{t=x,y} \left(1 - e^{-\gamma \rho_t}\right) \tag{7}$$

$$\rho_x = \left(\frac{b_{c_x}^{gt} - b_{c_x}}{c_w}\right)^2 \tag{8}$$

$$\rho_y = \left(\frac{b_{c_y}^{gt} - b_{c_y}}{c_h}\right)^2 \tag{9}$$

$$\gamma = 2 - \Lambda \tag{10}$$

The shape cost Ω is defined as follows:

$$\Omega = \sum_{t=w,h} \left(1 - e^{-w_t}\right)^{\theta} \tag{11}$$

$$w_w = \frac{|w - w_{gt}|}{\max(w - w_{gt})} \tag{12}$$

$$w_h = \frac{|h - h_{gt}|}{\max(h - h_{gt})} \tag{13}$$

θ controls the degree of attention to shape error, and its selection is related to the specific dataset. If θ = 1, shape error will be optimized, thereby reducing the free movement of the predicted box and the actual box shape. The regression loss of the target box is written as follows:

$$L_{SIoU} = 1 - IoU + (\Delta + \Omega)/2 \tag{14}$$

4. Results

4.1. Experimental Environment

The experimental environment for the CPU is Intel(R) Core(TM) i7-9700k, the GPU is NVIDIA GeForce RTX 2080 Ti with 24 GB memory, RAM is DDR4 with 64 GB, the CUDA version is 1.12.1, and the deep learning framework is Pytorch 1.7.

4.2. Dataset Description

The dataset is grape leaf disease on the public dataset Plant Village [40]; labelimg was used as the tool for labeling datasets. This experiment extracts blight, black rot and black measles disease and uses the labelimg toolbox to manually label the grape disease spots. In order to improve data diversity and solve the model overfitting caused by data imbalance, this paper expanded the dataset by mirroring, flipping and other operations, as shown in Table 1. The first column is the serial number of grape leaf diseases, the second column is the name of grape leaf diseases, and the third column is the number of original pictures of each type of grape leaf disease in the Plant Village dataset. The original dataset is prone to overfitting in the training process of the deep learning model, so this paper expands the dataset to increase the generalization ability of the model, thereby alleviating the overfitting problem. In addition, the expanded dataset has better robustness and enhanced the stability of the model, and the fourth column is the number of expanded pictures in the Plant Village dataset; using python scripts, the original image is expanded by translation, flipping, rotation, mirroring, scaling and noise. Column 5, column 6 and column 7 are the numbers of images of the training set, the verification set and the test set, respectively.

Table 1. Dataset information.

Series	Class	Number of Original Images	Number of Expanded Images	Number of Training Set	Number of Validation Set	Number of Test Set
1	blight	1076	2908	2037	290	581
2	black rot	1180	2320	1624	232	464
3	black measles	1383	2528	1770	253	505

By reading the related articles in the field of crop leaf disease detection, it was found that most of the training sets account for 70% of the total dataset, the verification sets account for 10%, and the test sets account for 20%, so this paper divides them according to

this ratio. After inputting the dataset into the model, it was found that the indexes such as accuracy, recall and average accuracy tend to be stable within 150 to 180 rounds; therefore, this paper iterates 200 times. When training a deep learning model, the batch size directly affects the required computing resources and memory. A large batch size may overflow the GPU memory, while a small batch size may fail to make full use of the GPU's computing power. In this case, we chose a batch size of four to achieve a balance between compute resources and memory. The learning rate determines the amplitude of weight updating in each iteration of the model. The improved YOLOv7 model is a relatively complex model containing a large number of parameters and layers. For a model of this scale, a higher initial learning rate can accelerate the updating of parameters and the convergence of the model. This can provide better training stability and convergence speed, and the learning rate of 0.01 can help the model find a good solution space faster. Cycle learning rate is a learning rate adjustment strategy that allows the learning rate to change dynamically during the training process. By setting a periodic learning rate, the step size can be adjusted at different stages of training to better optimize the model. In this paper, the cycle learning rate is set at 0.01. This is in order to better adapt to the needs of model training and help the model find the best step size in different training stages to improve the training efficiency and performance. Learning rate momentum is a hyperparameter used to accelerate the gradient descent process, usually between 0 and 1, in which a larger momentum value can accelerate the convergence of the model. Through experiments and tuning, we found that in the model, setting the learning rate momentum to 0.937 can help the model converge faster and improve the stability of training, so we chose to set the learning rate momentum to 0.937. The weight attenuation coefficient is 0.0005. The weight attenuation coefficient is used to regularize the loss function to reduce the possibility of overfitting the model. The weight parameter in the model can be penalized to be closer to zero, and in many deep learning practices, 0.0005 is considered a relatively good experience value for different models and tasks.

4.3. Evaluating Indicator

In this paper, four performance indicators, missed detection rate (M), precision rate (P), recall rate (R) and mean average precision (mAP), are used to verify the effectiveness of the network. mAP refers to the average of average accuracy AP of all categories. Generally, the threshold is set to 0.5; that is, the prediction box with IoU greater than 0.5 is valid, which is represented by mAP@0.5.

4.4. Experimental Results and Analysis

Firstly, the influence of SIoU loss function on the network is verified. YOLOv7 is trained with the original CIoU loss function and SIoU loss function. The visualization result of the loss function is shown in Figure 8. The loss value of using SIoU is smaller than that of CIoU, and the stability is also improved to some extent. Therefore, it is more important to use SIoU as the bounding box loss function of the dataset in this paper to improve the performance of the network model.

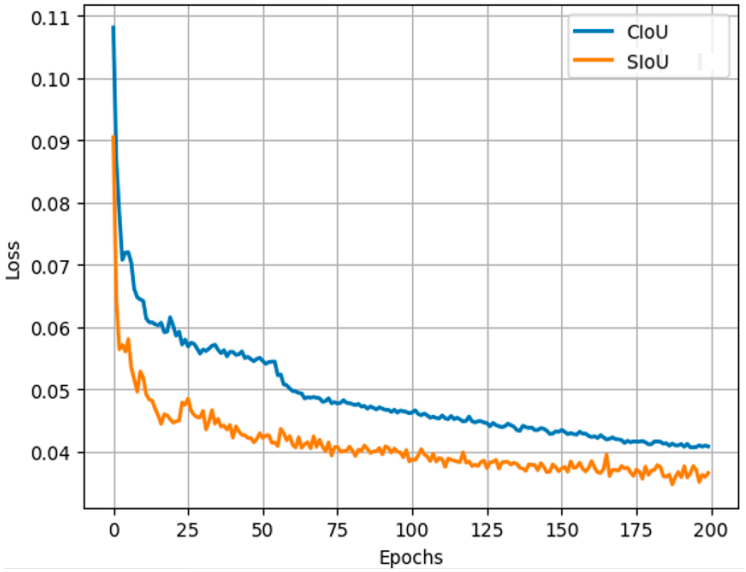


Figure 8. Loss visualization results diagram.

In order to verify the effectiveness of the method proposed in this paper in improving the accuracy of the YOLOv7 network, we compared it with other object detection models, including Faster-RCNN, YOLOv3, SSD and YOLOv7, as shown in Table 2.

Table 2. Detection results of different models.

Method	Blight (%)	Black Rot (%)	Black Measles (%)	mAP (%)
Faster-RCNN	89.6	89.6	89.9	89.7
SSD	89.3	89.8	90.6	89.9
YOLOv5	89.6	89.8	91.5	90.3
YOLOv7	89.9	91.4	91.1	90.8
YOLOv8	89.5	89.7	91.1	90.1
Improved YOLOv7	93.5	93.7	93.3	93.5

The mAP of the improved model is the highest, reaching 93.5%, which is 3.8%, 3.6%, 3.2%, 2.7% and 3.4% higher than that of Faster-RCNN, SSD, YOLOv5, YOLOv7 and YOLOv8, respectively. Compared with YOLOv7, the mAP of blight, black rot and black measles increased by 3.6%, 1.7% and 2.2%, respectively. The mAP of the improved YOLOv7 is the largest, which shows that adding a small target detection layer improve CA attention can enhance the feature extraction ability of YOLOv7 and can solve the problem of low accuracy in identifying small disease spots in the early stage of grape leaf diseases.

In order to further analyze the performance of different classes, the precision, recall, mAP@0.5 and mAP@0.5:0.95 of each class of the improved network are analyzed as shown in Table 3. The detection accuracy of black rot and blight is higher than that of black measles, mainly because the size and shape of black rot and downy mildew spots are regular, which is beneficial to model detection.

In order to verify the effectiveness of this algorithm for small targets, 1551 images in the test set are selected to verify this algorithm. Among them, there are 571 images of blight disease with 1730 small targets, 478 images of black rot with 1356 small targets and 502 images of ring spot disease with 2098 small targets. As can be seen from Table 4, the recognition rate of the three diseases was significantly improved by an improved YOLOv7 algorithm. Compared with the original YOLOv7 model, the recognition rate of the three diseases was increased by 6%, 8% and 7%, respectively.

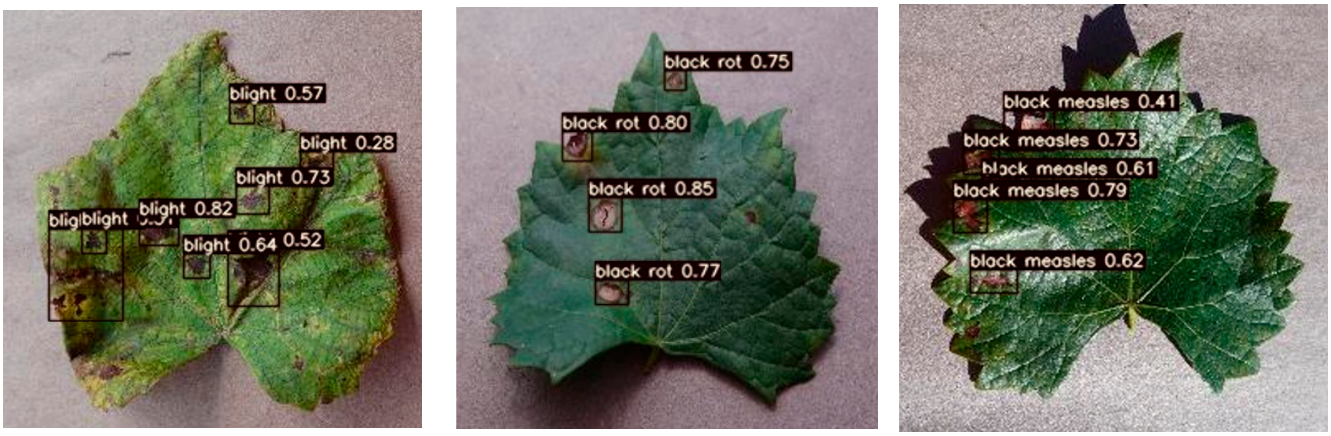
Table 3. Performance analysis of YOLOv7 and improved YOLOv7.

Class	Method	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
blight	YOLOv7	89.9	92.6	89.9	55.3
	Improved YOLOv7	93.5	95.6	93.5	58.5
black rot	YOLOv7	91.4	92.9	91.4	52.5
	Improved YOLOv7	93.6	94.7	93.7	55.4
black measles	YOLOv7	81.6	79.3	91.1	43.4
	Improved YOLOv7	84.3	81.9	93.3	45.4

Table 4. Comparison of small target detection quantity.

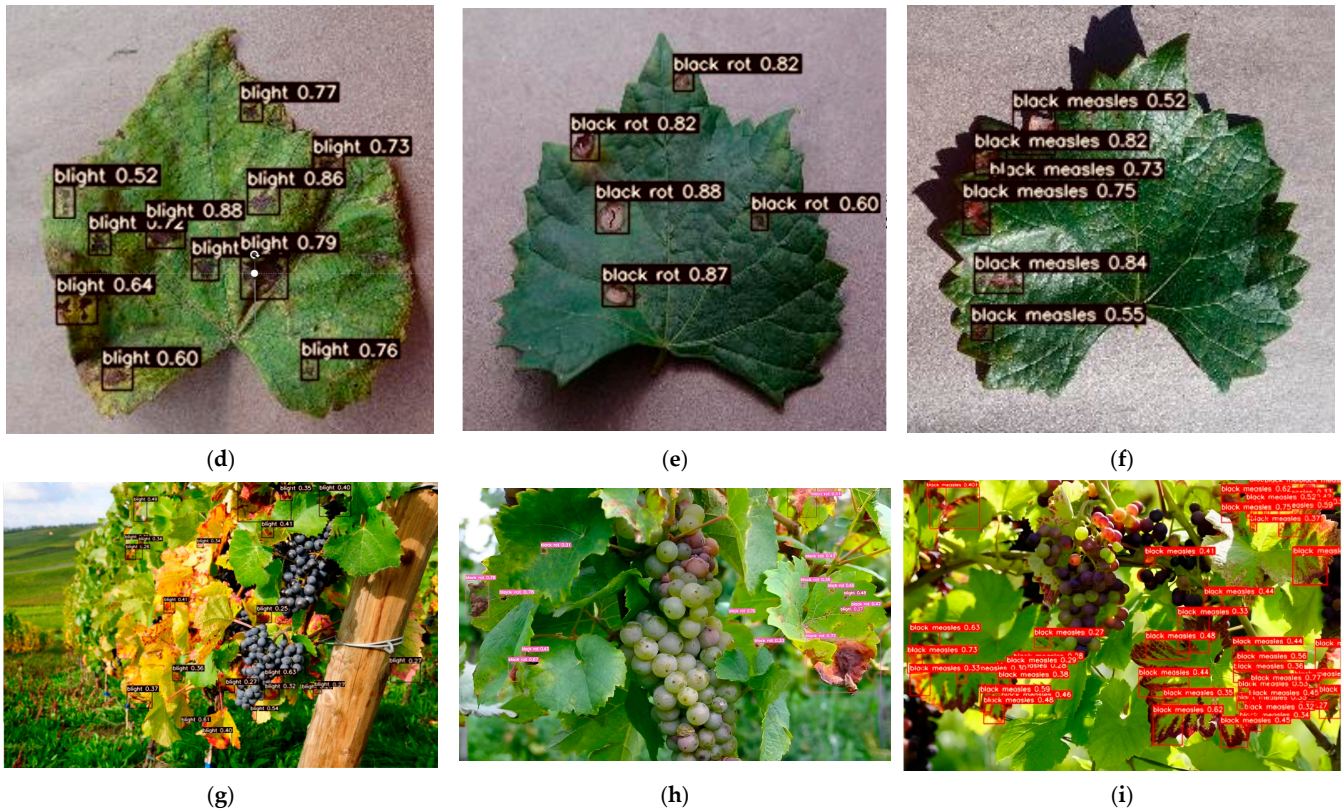
	Small Target Quantity	YOLOv7	Improved YOLOv7
blight	1730	1570	1675
black rot	1356	1223	1339
black measles	2098	1803	1954

In order to verify the advantages of the improved model in the identification of grape leaf diseases, the detection results for each of the three diseases are shown in Figure 9 by YOLOv7 and improved YOLOv7, respectively. According to the comparison, it can be found that the original YOLOv7 model in the figure has many missed detections in the detection of small disease spots, and the discrimination of a single disease spot is low. The missed detection rate of the improved YOLOv7 algorithm for grape leaf lesion detection is obviously reduced, and the regional location of the lesion is more accurate. At the same time, the disease spots of different leaves can be detected simultaneously in the natural environment, and the disease spots in the edge area of the image also have a good detection effect, as shown in Figure 9.



(**a**) (**b**) (**c**)

Figure 9. *Cont*.



(**d**) (**e**) (**f**)

(**g**) (**h**) (**i**)

Figure 9. The disease detection results of YOLOv7 and improved YOLOv7: (**a**) Blight detected by YOLOv7; (**b**) Black rot detected by YOLOv7; (**c**) Black measles detected by YOLOv7; (**d**) Blight detected by improved YOLOv7; (**e**) Black rot detected by improved YOLOv7; (**f**) Black measles detected by improved YOLOv7; (**g**) The improved YOLOv7 detects blight under natural conditions; (**h**) The improved YOLOv7 detects black rot under natural conditions; (**i**) The improved YOLOv7 detects black measles under natural conditions.

Through field experiments, we can better understand the performance of the algorithm in actual use and provide us with more datasets in real scenarios, which is also very beneficial to the training and verification of the algorithm. Therefore, we took three images using a Canon EOS 90D mid-range DSLR camera with a pixel size of 1280 × 720, it is produced in Shenzhen, Guangdong, China. Grape disease scenes were detected under natural conditions, enhancing the credibility of grape disease image identification under natural imaging conditions. As can be seen from Figure 9, this algorithm can not only detect on the test set but also has a good detection effect in complex environments such as background occlusion under natural conditions and similar colors between disease spots and background.

5. Discussion

In order to verify the effectiveness of the improved CA attention in this paper, we compared it with CA attention, as shown in Figure 10 and Table 5.

Table 5. Performance analysis of CA and improved CA.

Method	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
CA	87.8	88.4	91.5	51.3
Improved CA	89.3	88.8	92.4	51.9

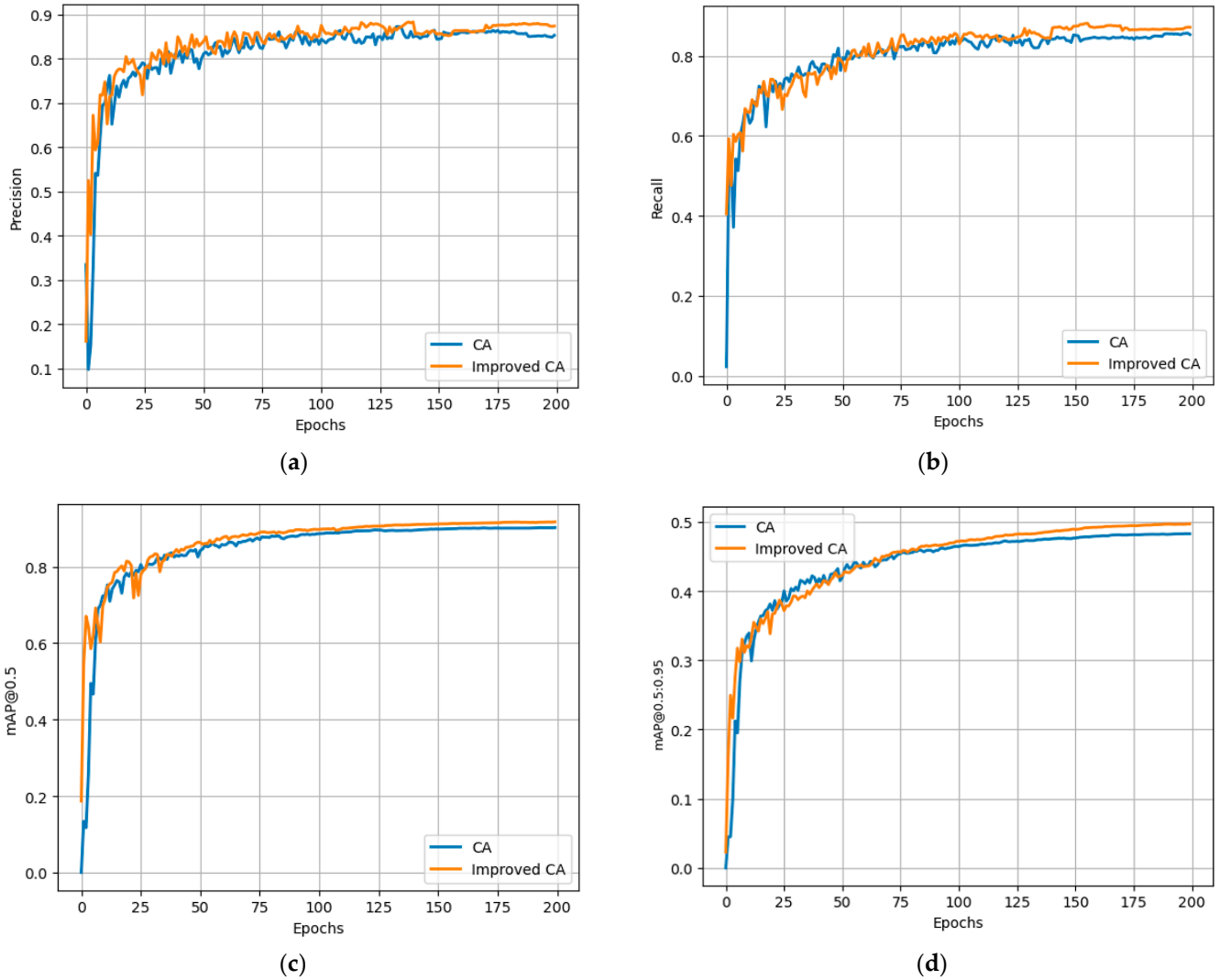


Figure 10. Performance comparison chart of CA and improved CA algorithms: (**a**) precision; (**b**) recall; (**c**) mAP@0.5; (**d**) mAP@0.5:0.95.

As can be seen from the table, compared with the original CA model, the precision, recall, mAP@0.5 and mAP@0.5:0.95 of the improved algorithm are increased by 1.5%, 0.4%, 0.9 and 0.6%, respectively.

As can be seen from Figure 10, precision, recall, mAP and mAP@0.5:0.95 all increase to varying degrees, indicating that CA has a stronger feature extraction ability after improved attention. Moreover, in rounds 10 to 25, the curves of the model with improved CA were smoother than those with CA, indicating that the stability of the model was further enhanced.

The mean of various objects was taken as the overall performance evaluation index and compared with the original YOLOv7, as shown in Figure 11. It shows that the P2 detection layer constructed by this algorithm improves the accuracy of small targets, the improved E-ELAN enhances the ability to extract small target features, the improved CA attention enhances the ability to extract key information of disease spots, and the SIoU enhances the positioning ability of the network, which means that the improved model can improve the image detection accuracy of grape leaf diseases and greatly improve the detection effect of small disease spots. The improved YOLOV7 was compared with the original YOLOv7, as shown in Figure 11. As can be seen from Figure 11, precision, recall, mAP and mAP:0.95 all increase to varying degrees, and most of the four indexes of the improved YOLOV7 are higher than YOLOv7 in 200 rounds, indicating that the improved

YOLOV7 has stronger feature extraction capability. It can solve the problem of low accuracy of small spot recognition in the early stage of grape leaf disease.

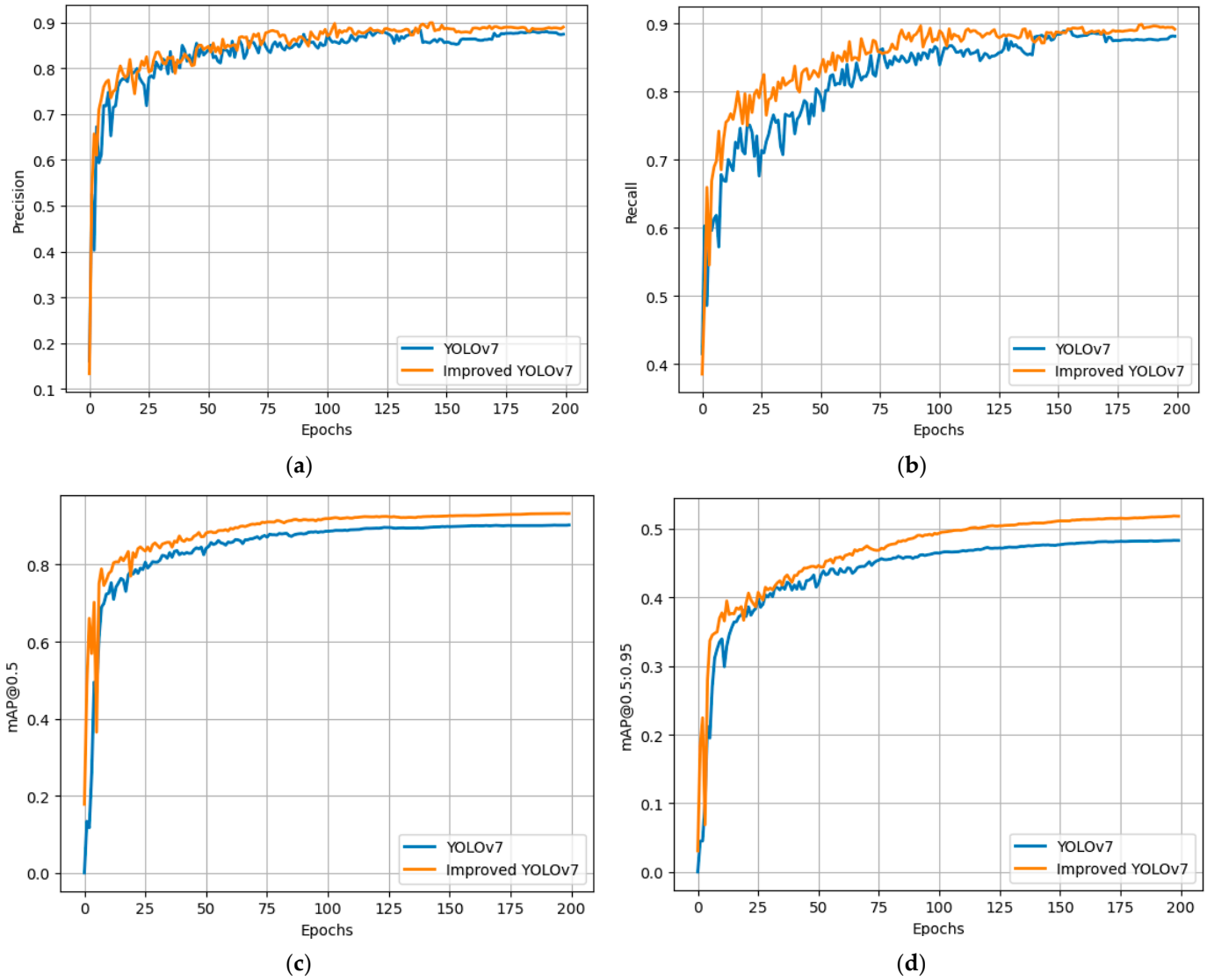


Figure 11. Performance comparison between YOLOv7 and the improved YOLOv7: (**a**) precision; (**b**) recall; (**c**) mAP@0.5; (**d**) mAP@0.5:0.95.

In order to verify the model detection effect after adding different improvements, the improved feature fusion network, improved E-ELAN module, improved CA attention module and SIoU loss function are ablated and analyzed. The comparison results of ablation experiments are shown in Table 6. As can be seen from the table, the mAP was significantly improved after the improvement of the feature fusion network, which shows that the low-level and high-level feature fusion and the addition of detection heads have played a very good role and significantly improved the detection performance. Adding improved E-ELAN and CA can obviously improve mAP, which shows that adding these two modules is effective. After using the SIoU loss function, the model diagram is slightly improved. Finally, it can be seen that the SIoU loss function not only accelerates the convergence of the model but also improves the detection performance of the model. In the table, "×" indicates that it was not added in YOLOv7, and "√" indicates that it was added in YOLOv7.

Table 6. Ablation experimental results.

Experiment	SIoU	Improved Feature Fusion Network	Improved E-ELAN	Improved CA	mAP@0.5
1	×	×	×	×	90.8
2	√	×	×	×	90.9
3	√	√	×	×	91.7
4	√	×	√	×	91.3
5	√	×	×	√	92.5
6	√	√	√	×	92.3
7	√	×	√	√	92.9
8	√	√	×	√	92.6
9	√	√	√	√	93.5

6. Conclusions

In this paper, three common diseases of grape leaves, namely black rot, black rot and black measles, were taken as the research objects. An improved YOLOv7-based grape leaf disease detection algorithm was proposed to counteract the problem of poor detection effect of small spots. The algorithm first constructs a new feature layer and introduces PANet to improve the E-ELAN structure. Secondly, Maxpool is introduced into the original CA attention to obtain more abundant feature information, and the global information module is introduced into the CA attention to assign different weights to deep image features so as to obtain channel information and position information and improve the recognition effect of the model. Finally, the loss function CIoU is replaced by SIoU to enhance the positioning ability of the network. Compared with the original YOLOv7, the algorithm proposed in this paper increases the mAP by 2.7% and has a good detection effect in both natural and simple environments.

This paper aimed to detect three kinds of diseases in grape leaves. In order to verify the robustness of the model proposed in this paper, it is necessary to increase the categories in the grape disease dataset to obtain a more comprehensive effect. In addition, the algorithm can be applied to the detection of other agricultural diseases. Building an agricultural disease detection platform through mobile phones can also provide farmers with convenient and efficient disease detection services.

Author Contributions: Conceptualization, M.Y., X.T and H.C.; methodology, M.Y. and X.T.; software, X.T.; validation X.T.; writing—review and editing M.Y., X.T. and H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the high-end foreign experts' introduction program (G2022012010L).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: https://github.com/spMohanty/PlantVillage-Dataset (1 February 2018).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Global Grape Production Will Reach 27.3 Million Tons, with China's Production Ranking First. Available online: http://news.foodmate.net/2023/07/665306.html (accessed on 8 January 2024).
2. Grape Harvest in the Southern Hemisphere, a Few Happy and a Few Sad. Available online: https://m.163.com/dy/article_v2/I610I88M0553OUSU.html (accessed on 8 January 2024).
3. Chen, J.; Zhang, D.; Nanehkaran, Y.A. Identifying plant diseases using deep transfer learning andenhanced lightweight network. *Multimedia Tools Appl.* **2020**, *79*, 31497–31515. [CrossRef]
4. Wang, D.; Chai, X. Application of machine learning inplant diseases recognition. *J. Chin. Agric. Mech.* **2019**, *40*, 171–180. [CrossRef]
5. Adeel, A.; Khan, M.A.; Sharif, M.; Azam, F.; Shah, J.H.; Umer, T.; Wan, S. Diagnosis and recognition of grape leaf diseases: An automated system based on a novel saliency approach and canonical correlation analysis based multiple features fusion. *Sustain. Comput. Inform. Syst.* **2019**, *24*, 100349. [CrossRef]

6. Agrawal, N.; Singhai, J.; Agarwal, D.K. Grape Leaf Disease Detection and Classification Using Multi-Class Support Vector Machine. In Proceedings of the 2017 International Conference on Recent Innovations in Signal Processing and Embedded Systems(RISE), Bhopal, India, 27–29 October 2017; pp. 238–244.
7. Tian, Y. Grape disease recognition based on texture feature and support vector machine. *Chin. J. Sci. Instrum.* **2005**, *26*, 606–608.
8. Tian, Y.; Li, T.; Li, C.; Piao, Z.; Sun, G.; Wang, B. Grape disease image recognition method based on support vector machine. *Trans. Chin. Soc. Agric. Eng.* **2007**, *23*, 175–180.
9. Mohan, K.J.; Balasubramanian, M.; Palanivel, S. Detection and recognition of diseases from paddy plant leaf images. *Int. J. Comput. Appl.* **2016**, *144*, 143–156.
10. Padol, P.B.; Sawant, S.D. Fusion Classification Technique Used to Detect Downy and Powdery Mildew Grape Leaf Diseases. In Proceedings of the International Conference on Global Trends in Signal Processing, Jalgaon, India, 22–24 December 2016.
11. Waghmare, H.; Kokare, R.; Dandawate, Y. Detection and Classification of Diseases of Grape Plantusing Opposite Colour Local Binary Pattern Feature and Machine Learning for Automated Decision Support System. In Proceedings of the 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 11–12 February 2016; pp. 513–518.
12. Jaisakthi, S.M.; Mirunalini, P.; Thenmozhi, D. Grape Leaf Disease Identification Using Machine Learning Techniques. In Proceedings of the 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, India, 21–23 February 2019; pp. 1–6.
13. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 1419. [CrossRef] [PubMed]
14. Ramcharan, A.; Baranowski, K.; McCloskey, P. Deep learning for image-based cassava disease detection. *Front. Plant Sci.* **2017**, *8*, 1852. [CrossRef]
15. Liu, B.; Zhang, Y.; He, D.J. Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry* **2017**, *10*, 11. [CrossRef]
16. Liu, Z.; Yuan, X.; Weng, J.; Liao, Y.; Xie, L. Recognition of strawberry leaf powdery mildew disease based on convolutional neural networks. *Jiangsu J. Agric. Sci.* **2018**, *34*, 527–532.
17. Too, E.C.; Yujian, L.; Njuki, S. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* **2019**, *161*, 272–279. [CrossRef]
18. Gong, A.; Jing, X. Crop disease image recognition based on multi-convolutional neural network model fusion. *J. Comput. Technol. Dev.* **2020**, *30*, 134–139.
19. Long, Y.; Liu, C.; Sun, K. Crop disease recognition based on deep convolutional neural networks. *J. Wuhan Univ. Light Ind.* **2020**, *39*, 17–22.
20. Fuentes, A.; Yoon, S.; Kim, S.C. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* **2017**, *17*, 2022. [CrossRef] [PubMed]
21. Adhikari, S.; Saban Kumar, K.C.; Balkumari, L. Tomato Plant Diseases Detection System Using Image Processing. In Proceedings of the 1st KEC Conference on Engineering and Technology, Lalitpur, Nepal, 27–29 September 2018; Volume 1, pp. 81–86.
22. Liu, T.; Feng, Q.; Yang, S. Grape leaf disease detection method based on convolutional neural network. *Northeast. Agric. Univ.* **2018**, *49*, 73–83.
23. Qi, X. Construction and application research of grape pest and disease recognition model based on deep learning. *Xian Build. Univ. Sci. Technol.* **2022**. [CrossRef]
24. Jiang, P.; Chen, Y.; Liu, B. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access* **2019**, *7*, 59069–59080. [CrossRef]
25. Zhao, J.; Qu, J. A Detection Method for Tomato Fruit Common Physiological Diseases Based on YOLOv2. In Proceedings of the 2019 10th International Conference on Information Technology in Medicine and Education (ITME), Qingdao, China, 23–25 August 2019; pp. 559–563.
26. He, X. *Research on Grape Leaf Disease Recognition Method Based on Deep Learning*; Northwest Agriculture and Forestry University: Xianyang, China, 2020.
27. Zhu, X.; Ma, H.; Ji, J. Detection and recognition analysis of blueberry canopy fruits based on Faster R-CNN. *J. South. Agric. Sci.* **2019**, *51*, 1493–1501.
28. He, Z.; Huang, J.; Liu, Q.; Zhang, Y. Segmentation of apple leaf disease based on asymmetric mixing convolutional neural network. *Trans. Chin. Soc. Agric. Mach.* **2021**, *52*, 221–230.
29. Zhuo, W.; Jian, W.; Wang, X.; Jia, S.; Bai, X.; Zhao, Y. Lightweight detection method of apple in natural environment based on improved YOLO v4. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53*, 294–302.
30. Huang, T.; Huang, H.; Li, Z. Citrus fruit recognition method based on improved YOLOv5 model. *Huazhong Agric. Univ.* **2022**, *41*, 170–177.
31. Zhao, Z.; Guo, G.; Zhang, L.; Li, Y. A new anti-vibration hammer rust detection algorithm based on improved YOLOv7. *Energy Rep.* **2023**, *9*, 345–351. [CrossRef]
32. Zhuang, J.; Zhang, Y.; Wang, J. A Dragon Fruit Picking Detection Method Based on YOLOv7 and PSP-Ellipse. *Sensors* **2023**, *23*, 3803.
33. Zhang, L.; Zhang, L. MS-YOLOv7:YOLOv7 Based on Multi-Scale for Object Detection on UAV Aerial Photography. *Drones* **2023**, *7*, 188.

34. Ding, X.; Guo, Y.; Ding, G. ACNet: Strengthening the Kernel Skeletons for Powerful CNN via Asymmetric Convolution Blocks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1911–1920.
35. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
36. Woo, S.; Park, J.; Lee, J.Y. Cbam: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
37. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 13713–13722.
38. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
39. Gevorgyan, Z. SIoU loss: More powerful learning for bounding box regression. *arXiv* **2022**, arXiv:2205.12740.
40. Hughes, D.; Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:1511.08060.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

A Residual Network with Efficient Transformer for Lightweight Image Super-Resolution

Fengqi Yan [1], Shaokun Li [1], Zhiguo Zhou [1,2,*] and Yonggang Shi [1,*]

[1] School of Integrated Circuits and Electronics, Beijing Institute of Technology, Beijing 100081, China; 3220210682@bit.edu.cn (F.Y.); 3120221325@bit.edu.cn (S.L.)
[2] Tangshan Research Institute of Beijing Institute of Technology, Tangshan 063000, China
* Correspondence: zhiguozhou@bit.edu.cn (Z.Z.); ygshi@bit.edu.cn (Y.S.)

Abstract: In recent years , deep learning approaches have achieved remarkable results in the field of Single-Image Super-Resolution (SISR). To attain improved performance, most existing methods focus on constructing more-complex networks that demand extensive computational resources, thereby significantly impeding the advancement and real-world application of super-resolution techniques. Furthermore, many lightweight super-resolution networks employ knowledge distillation strategies to reduce network parameters, which can considerably slow down inference speeds. In response to these challenges, we propose a Residual Network with an Efficient Transformer (RNET). RNET incorporates three effective design elements. First, we utilize Blueprint-Separable Convolution (BSConv) instead of traditional convolution, effectively reducing the computational workload. Second, we propose a residual connection structure for local feature extraction, streamlining feature aggregation and accelerating inference. Third, we introduce an efficient transformer module to enhance the network's ability to aggregate contextual features, resulting in recovered images with richer texture details. Additionally, spatial attention and channel attention mechanisms are integrated into our model, further augmenting its capabilities. We evaluate the proposed method on five general benchmark test sets. With these innovations, our network outperforms existing efficient SR methods on all test sets, achieving the best performance with the fewest parameters, particularly in the area of texture detail enhancement in images.

Keywords: single-image super-resolution; blueprint-separable convolution; efficient transformer; spatial attention; channel attention

1. Introduction

Single-Image Super-Resolution (SISR) [1] is one of the classical problems in the field of computer vision and image processing; its main objective is to reconstruct a high-resolution image from an input low-resolution image. To solve this highly ill-posed problem, many different approaches have been proposed, among which, Convolutional Neural Networks (CNNs), with their powerful feature extraction capabilities, have become the mainstream approach in this field in recent years [2–4]. Recently, due to its success in Natural Language Processing (NLP), transformers [5] have attracted the attention of the computer vision field, and after its success in advanced computer vision [6,7], transformers have also provided new approaches to underlying vision tasks [8,9] and super-resolution [10,11].

With the development of deep learning and convolutional neural networks, it has been realized that deeper networks [2,12] can substantially improve the quality of recovered images. Their remarkable achievements can be attributed in part to the utilization of large-scale models characterized by dense parameters. This trend has, however, resulted in an escalating demand for hardware resources in Super-Resolution (SR) networks. Such resource intensiveness imposes constraints on the progress and real-world applicability of SR techniques. To address this problem, many lightweight SR networks have been

proposed. The main approaches currently include Neural Architecture Search (NAS) [13,14], recursive operations [15], information or feature distillation mechanisms [16,17], and single-pass shallow networks [15,18]. However, these methods often do not aggregate global information well due to the limitations of the convolution operation, which leads to the problem of artifacts in the recovered images.

With the emergence of transformers, many image restoration methods based on it have achieved significant progress [11]. Particularly, a recently designed SwinIR network has made groundbreaking advancements in super-resolution tasks. However, existing transformer-based super-resolution networks still face challenges such as excessive parameter counts and slow inference speeds. Additionally, these networks are relatively weak for spatial and channel modeling, leaving room for improvement. Moreover, there is a limited amount of research on lightweight transformer networks specifically designed for super-resolution tasks. Therefore, the objective of this study is to develop a lightweight transformer network suitable for single-image super-resolution tasks.

Currently, most efficient SR models [16,17] employ feature distillation techniques to reduce the parameter count. However, this approach is not hardware-friendly and significantly reduces the inference speed [19]. To address these limitations, we propose a Residual Network with Efficient Transformer (RNET) for lightweight image SR. This introduces a novel approach called 'local residual connections'. This method effectively balances the parameter count and restoration performance while maintaining hardware-friendliness. To achieve superior visual quality, we design a Hybrid Feature Extraction Block (HFEB); it consists of three main components: (1) a local feature extraction module built using Blueprint-Separable Convolution (BSConv) [20], which is an improved version of Deep Separable Convolution (DSConv) [21], to achieve efficient separation and reduce redundancy by better exploiting intra-kernel correlations; (2) an attention module, which consists of Enhanced Spatial Attention (ESA) [16] and Contrast-aware Channel Attention (CCA) [22] to enhance the model's ability; and (3) an efficient transformer module, consisting of Multi-BSConv head Transposed Attention (MBTA) that performs feature interactions across channels, and a Gated-BSConv Feed-forward Network (GBFN). This transformer module has linear complexity that greatly reduces the computational pressure and makes the network better able to aggregate long-distance features. By incorporating these components, our RNET significantly improves the performance of super-resolution tasks while producing visually appealing and detail-rich restored images.

Our main contributions are as follows:

1. We propose a local feature extraction module that utilizes BSConv and two efficient attention modules and demonstrate its effectiveness at SR tasks.
2. We propose an efficient transformer module that enhances global feature extraction without significantly increasing computational complexity. This module contributes to producing more-detailed restored images.
3. We propose a strategy of local residual connections, which differs from the commonly used feature distillation approach. This strategy maintains a low parameter count and achieves excellent performance.

2. Related Work

2.1. CNN-Based Image Super-Resolution

In recent years, the successful application of deep Convolutional Neural Networks (CNNs) in the underlying vision tasks has been widely recognized [23,24]. In particular, SRCNN [2] was the first to introduce deep convolutional neural networks to the image SR task; the authors used a pioneering three-layer convolutional neural network to map the correlation between LR and HR images; they achieved encouraging results and provided a new approach to the single-image super-resolution task. A number of approaches have been subsequently proposed to further enhance the capabilities of convolutional neural networks in SR tasks, among which, increasing the depth of the network and using residual connection are widely recognized as effective approaches. Kim et al. [3] proposed VDSR,

which uses a 19-layer convolutional neural network to achieve SISR. Lim et al. [12] modified SRResNet [25] to construct a deeper and wider residual network, called EDSR, which has a large number of learnable parameters that greatly improve SR performance and has served as a baseline to inspire many subsequent works [16,26]. For example, Zhang et al. [4] added a dense block to the original residual block, and soon after that, Zhang et al. [26] used residual channel attention to further improve the ability of the network. In the last five years, several works have explored different network frameworks, such as recurrent neural networks [15,16], graphical neural networks [27], and generative adversarial networks (GANs) [25,28]. Specially, Xintao et al. [28] used a network with a residual-in-residual structure and successfully recovered more-realistic texture detail information with the training strategy of a GAN.

2.2. Efficient Super-Resolution

Existing SR models often need to introduce large computational cost in order to improve performance, which limits the practical application of these methods. To solve this problem, many lightweight networks have been designed. For example, Deep Recurrent Convolutional Network (DRCN) [29] and Deep Recurrent Residual Network (DRRN) [30] reduce computation by introducing recursive layers to share parameters. Lai et al. [31] combine a traditional image algorithm using a Laplacian pyramid with deep learning to achieve real-time reconstruction. Tai et al. [32] propose MemNet, which uses a gating mechanism to link deep features with shallow information. Zhang et al. [33] use a reparameterization module to build a fast inference model that can be used for mobile devices. The IDN proposed by Hui et al. [17] compresses the model size by using group convolution and combining short-term and long-term features. Hui et al. [22] then propose an information distillation strategy based on IDN that extracts hierarchical features step-by-step through a splitting operation to further improve the efficiency of the model. Based on this work, Liu et al. [16] further improve the information multiple distillation block in IMDN and achieve better results. In addition, Han et al. [34] propose self-calibrating convolution and pixel-focused blocks and achieve a performance improvement without increasing the complexity. Wang et al. [35] propose a lightweight Contextal Transformation Layer (CTL) and build a Contextal Transformation Network (CTN) based on it. CTN achieves good results in both lightweight remote-sensing-image super-resolution tasks and natural-image super-resolution tasks.

2.3. Vision Transformer

The success of transformers [5] in the field of natural language processing has also attracted the attention of the computer vision community. The core idea of transformers is self-attention, which captures long-term information between sequence elements. Transformers have been successfully applied to many advanced vision tasks, including image classification [6], target detection [36], and segmentation [37]. Although transformers have shown strong capability for studying long-term dependencies between image pixels, there is still a lot of work to show that convolution can help transformers achieve better visual representation [38,39]. ViT [6] was the first work to replace convolution with a transformer. To generate sequence elements, ViT flattens 2D image patches into vectors and feeds them into a transformer. Also, transformers have been introduced to underlying vision tasks [8,9], especially super-resolution [10,11]. Liang et al. [11] proposed SwinIR, which introduced swin transformers [7] to SISR for the first time and demonstrated the great potential of transformers in SR tasks. Wang et al. [40] propose Detail-Preserving Transformer (DPT), which introduces a new self-attention mechanism that considers global space-angle relationships. DPT has become the SOTA method in the field of light-field-image SR. Currently, most ViTs segment images into a series of patches and then learn their dependencies by self-attention. This helps networks build excellent long-range or global dependencies. However, self-attention mechanisms bring huge computation and

GPU resource consumption. Therefore, designing efficient vision transformers has become a hot and open research topic in recent years.

3. Method

In this section, we first present the overall architecture of the Residual Network with Efficient Transformer (RNET). Then, we introduce the Hybrid Feature Extraction Block (HFEB), which can efficiently extract the texture information of the image. Finally, we introduce the Efficient Transformer (ET) block.

3.1. Network Structure

The overall structure of our method Residual Network with Efficient Transformer (RNET) is shown in Figure 1. RNET is composed of three main parts: a shallow feature extraction module, a deep feature extraction module, and an image reconstruction module.

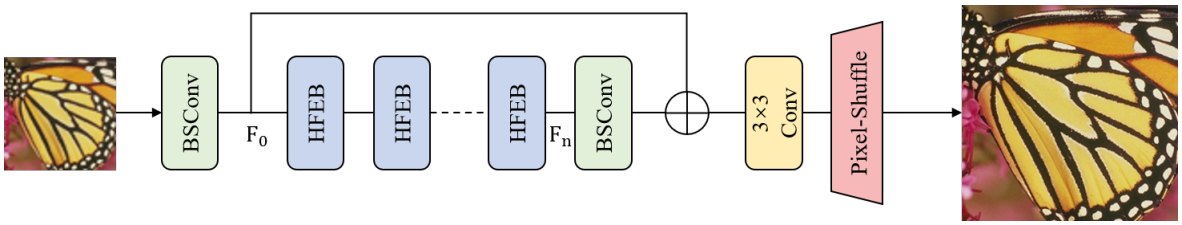


Figure 1. The structure of the proposed RNET for lightweight image super-resolution.

Regarding the shallow feature extraction module: first, given the input low-resolution image $I_{LR} \in R^{H \times W \times C_{in}}$, we use BSConv to obtain the shallow features $F_0 \in R^{H \times W \times C}$. The structure of BSConv is shown in Figure 3. The process can be expressed as follows:

$$F_0 = H_{BSconv}(I_{LR}) \tag{1}$$

where $H_{BSconv}(\cdot)$ denotes BSConv. Next, the deep feature extraction module consists of multiple directly connected Hybrid Feature Extraction Blocks (HFEBs), and this process can be expressed as:

$$F_n = H^n_{HFEB}(F_{n-1}) \tag{2}$$

where $H^n_{HFEB}(\cdot)$ denotes the nth HFEB. F_{n-1} and F_n, respectively, denote the input features and output features of the nth HFEB. In addition, we use a BSConv layer $H_{BSconv}(\cdot)$ to smooth the depth features. Lastly, the image reconstruction module $H_{Rec}(\cdot)$ consists of a 3×3 standard convolution with pixel shuffling: the objective is to upsample the fused features and recover them to HR size. Finally, we fuse shallow and deep features by adding global residual connections. The process can be expressed as:

$$I_{SR} = H_{Rec}(H_{BSconv}(F_n) + F_0) \tag{3}$$

where I_{SR} indicates the final recovered image through the above network.

3.2. Hybrid Feature Extraction Block

In this subsection, we introduce the Hybrid Feature Extraction Block (HFEB): the specific structure of this module is shown in Figure 2c. We use BSConv instead of traditional convolution, which greatly reduces the computational burden. Also, we use local residual connections instead of separable distillation structures. Kong et al. [19] demonstrate that the local residual structure can significantly reduce inference time while maintaining the model's capability.

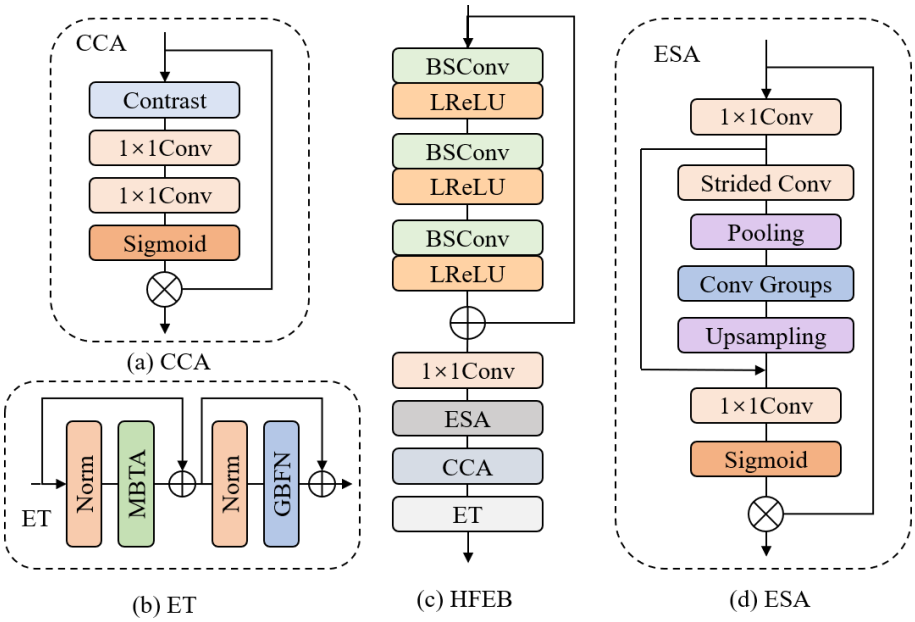


Figure 2. (**a**) The architecture of CCA block. (**b**) The architecture of ET block (**c**) The architecture of our Hybrid Feature Extraction Block (HFEB). (**d**) The architecture of ESA block.

HFEB is divided into three main parts to extract image features: namely, a local feature extraction module, a hybrid attention mechanism, and an efficient transformer. First, we use several stacked BSConv and LeakyRelu layers to extract local features. Specifically, each BSConv layer is equipped with a LeakyRelu activation function. Given the input feature F_{in}, this part can be described as:

$$\begin{aligned} F_{LF1} &= \sigma(H_{BSconv}(F_{in})) \\ F_{LF2} &= \sigma(H_{BSconv}(F_{LF1})) \\ F_{LF3} &= \sigma(H_{BSconv}(F_{LF2})) \\ F_{out1} &= F_{in} + F_{LF3} \end{aligned} \tag{4}$$

where F_{LFn} denotes the output of the nth BSConv and LeakyRelu combination module, $\sigma(\cdot)$ denotes the LeakyRelu activation function, $H_{BSconv}(\cdot)$ denotes BSConv opration, and F_{out1} represents the output of the entire local feature extraction module.

Next, to get more-detailed information, we first pass the output of the local feature part through a 1×1 convolution before feeding it to the hybrid attention module. This module consists of a lightweight Enhanced Spatial Attention (ESA) block [16] and a Contrast-aware Channel Attention (CCA) block [22], and this process can be formulated as:

$$F_{out2} = H_{CCA}(H_{ESA}(H_{conv1\times1}(F_{out1}))) \tag{5}$$

$H_{CCA}(\cdot)$ and $H_{ESA}(\cdot)$ represent the CCA block and ESA block, respectively. $H_{conv1\times1}(\cdot)$ represents the 1×1 convolution. F_{out2} represents the output of the hybrid attention module.

Finally, to better aggregate global information, we feed the output of the hybrid attention module into the Efficient Transformer (ET) module. This module consists of Multi-BSConv head Transposed Attention (MBTA) and Gated-BSConv Feed-forward Network (GBFN); it retains the global feature extraction capability of a transformer while greatly reducing the computational burden. The whole process can be described as:

$$F_{out} = H_{ET}(F_{out2}) = H_{GBFN}((H_{MBTA}(F_{out2})) \tag{6}$$

where $H_{ET}(\cdot)$ represents the ET block. $H_{GBFN}(\cdot)$ and $H_{MBTA}(\cdot)$ represent the GBFN and MBTA, respectively. F_{out} represents the output of the whole HFEB.

3.2.1. BSConv

BSConv is a method that decomposes the standard convolution into a point-wise 1×1 convolution and a depth-wise convolution. It can be regarded as a variant of depth-separable convolution. The structure of BSConv is shown in Figure 3. Similar to depth-separable convolution, it has less computational burden than standard convolution, and it has been shown that BSConv has better results than standard convolution in most cases [20].

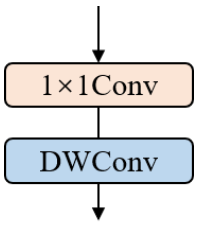


Figure 3. The structure of BSConv, where DWConv represents the depth-wise convolution.

3.2.2. ESA and CCA

The effectiveness of ESA and CCA modules for SR tasks has been demonstrated [16,22], so we introduce these two modules into our HFEB. The structure of ESA is shown in Figure 2d. It uses a 1×1 convolution to reduce the input channel, and it uses strided convolution and strided max-pooling to reduce the spatial size. Then, to obtain better results and higher efficiency, a group of BSConv layers are used to extract features instead of standard convolution. Features are processed by a 1×1 convolution to restore the origin channel. Finally, a Sigmoid function generates the attention matrix and the multiplied input feature. In order to achieve both spatial and channel information, we add the CCA block [22] after the ESA block. The structure of the CCA module is shown in Figure 2a; the structure represents a channel attention module specifically designed for low-level image processing. In contrast to conventional channel attention modules, CCA adopts the sum of the standard deviation and mean instead of global pooling. This modification proves advantageous for enhancing image details and texture structure information. Figure 4 presents the visualization results of feature maps before and after the incorporation of these two modules. Upon integrating the ESA and CCA modules, the proposed HFEB in this paper demonstrates improved capability to extract clearer edge and texture information. The inclusion of these two attention modules allows our network to further enhance the accuracy of SISR.

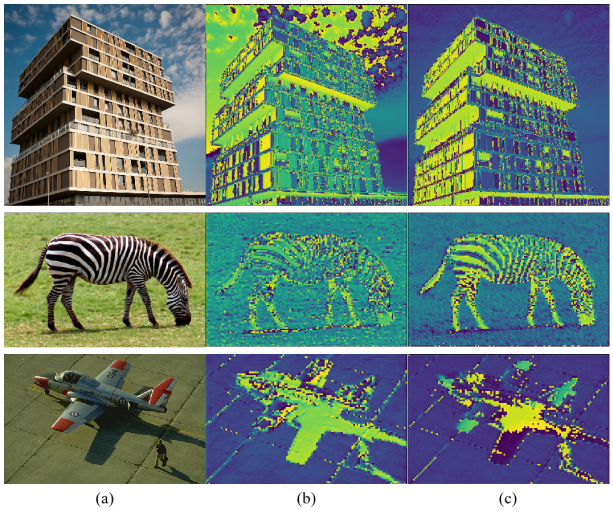


Figure 4. Feature map visualization results. Column (**a**) is the input image. Column (**b**) is the feature map visualization result without adding ESA and CCA modules. Column (**c**) is the feature map visualization result after adding ESA and CCA modules.

3.3. Efficient Transformer

Inspired by Restormer [41], in order to achieve better global modeling capabilities, we adopt an efficient transformer module, as shown in Figure 2b, consisting of MBTA and GBFN to further improve the performance of the network. The specific structure is shown in Figure 5.

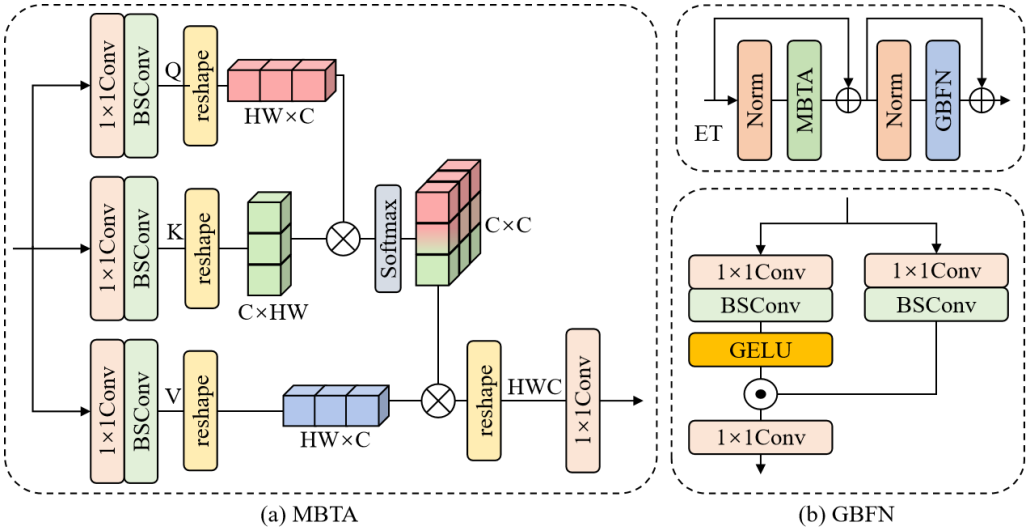


Figure 5. The architecture of our Efficient Transformer (ET) block. (**a**) The architecture of MBTA in ET. (**b**) The architecture of GBFN in ET.

3.3.1. Multi-BSConv Head Transposed Attention (MBTA)

The computational cost in a traditional transformer mainly comes from the self-attention layer. The time and memory complexity of the key-query dot-product interaction grows quadratically with the spatial resolution of the input. MBTA, on the other hand, greatly reduces the computational overhead by using cross-channel rather than cross-space self-attention; the specific structure is shown in Figure 5a. The MBTA process is described below: for the input nominal $X \in R^{H\times W\times C}$, we first achieve cross-channel pixel-level feature aggregation by 1×1 point-wise convolution, and subsequently, we use 3×3 BSConv to extract channel contextual information. The advantage of using deep convolution is that it can better focus on the local context before generating the global feature map. The query (Q), key (K), and value (V) are generated from the above process. This process can be formulated as:

$$\begin{aligned} Q &= H^q_{BSconv}\left(H^q_p(LN(X))\right) \\ K &= H^k_{BSconv}\left(H^k_p(LN(X))\right) \\ V &= H^v_{BSconv}\left(H^v_p(LN(X))\right) \end{aligned} \tag{7}$$

where $H_{BSconv}(\cdot)$ denotes the BSConv, $H_p(\cdot)$ denotes the 1×1 point-wise convolution, and LN denotes the layer normalization.

Next, a transposed attention map of size $R^{C\times C}$ is generated by reshaping the query and key projection via their dot product. This process can be formulated as:

$$\begin{aligned} Attention(Q, K, V) &= V \cdot Softmax(K \cdot Q/\alpha) \\ \hat{X} &= H_p(Attention(Q, K, V)) + X \end{aligned} \tag{8}$$

where $Softmax$ denotes the function of softmax to generate the probability map, α is a learnable scaling parameter to control the size of the dot product of K and Q before applying the softmax function, and $H_p(\cdot)$ indicates the point-wise 1×1 convolution.

The above operation successfully reduces the complexity of the transformer to a linear channel complexity, effectively reducing the computational cost.

3.3.2. Gated-BSConv Feed-Forward Network (GBFN)

To further recover the structural information, the efficient transformer employs a Gated-BSConv Feed-forward Network (GBFN). This feed-forward network has two important components: a gating mechanism and BSConv. Similar to MBTA, we use BSConv to learn the local information between spatially adjacent pixels, which is very effective for learning local similarity information of images for recovery and reconstruction. The specific structure is shown in Figure 5b. This process can be formulated as follows:

$$\begin{aligned} X_{gate} &= \phi\Big(H^1_{BSconv}(H_pLN(X))\Big) \odot H^2_{BSconv}\big(H_pLN(X)\big) \\ \hat{X} &= H_p(X_{gate}) \end{aligned} \tag{9}$$

where ϕ denotes the GELU function, LN denotes the layer normalization, $\odot$ denotes element-wise multiplication, and $H_p(\cdot)$ denotes a point-wise 1×1 convolution.

Overall, the GBFN controls the information flow through the respective hierarchical levels in our pipeline, thereby allowing each level to focus on the fine details complementary to the other levels. In contrast to the MBTA, the GBFN is able to provide richer contextual information.

3.4. Loss Function

Our network is optimized with a mean absolute error (MAE, also known as L1) loss function to facilitate a fair comparison. The loss function is described as follows: given a training set $\{I^i_{HR}, I^i_{LR}\}$ that contains several pairs of LR and HR inputs, the training objective is then to minimize the L1 loss function:

$$L(\Theta) = \frac{1}{N}\sum_{i=1}^{n} \|H_{model}(I^i_{LR}) - I^i_{HR}\|_1 \tag{10}$$

where Θ denotes the parameter set of the network, and $\|\cdot\|_1$ is the L1 norm. The loss function is optimized by using Adam optimizer.

4. Experiments

In this section, we use five benchmark datasets to evaluate the performance of our proposed method. First, we introduce the datasets used along with the evaluation metrics. Then, the superiority of the proposed method is demonstrated in terms of both visualization and evaluation metrics. Finally, the complexity and computational cost of the proposed model are explored.

4.1. Experiment Setup

4.1.1. Datasets and Metrics

We used 800 images from the DIV2K dataset [42] for training. We evaluated the performance of the different methods using five standard benchmark datasets from Set5 [43], Set14 [44], BSD100 [45], Urban100 [46], and Manga109 [47]. We evaluated the mean Peak Signal-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) on the Y-channel. And we downscale HR images with the scaling factors (×2, ×3, and ×4) using bicubic degradation models.

4.1.2. Training Details

Our model is trained on the RGB channel, and we enhance the training data by randomly flipping the images 90°, 180°, 270°, and horizontally. The number of HFEBs is set to 8, and the channel number is set to 48. The kernel size of all depth-wise convolutions is set to 3. The patch size for each LR input is set to 48×48, and we randomly crop HR

patches of size 480 × 480 from the ground truth. The batch size is 32. For all scales of the model, the training process is divided into two stages. In the first stage, we train the model from the beginning. In the second stage, we use a two-time warm-start strategy. In each stage, we use the Adam optimizer [48] with $\beta_1 = 0.9$, $\beta_2 = 0.999$. the initial learning rate is set to 5×10^{-4} with cosine learning rate decay. The L1 loss function is used to optimize the model for total 1×10^6 iterations in each stage. We implement our model on a GeForce RTX 3090 GPU using PyTorch 1.9.0; the training process takes about 40 h.

4.2. Quantitative Results

4.2.1. Comparison Results

To verify the effectiveness of our RNET, we compare it with 10 advanced efficient super-resolution models: SRCNN [2], FSRCNN [18], VDSR [3], DRRN [30], LapSRN [31], LAPAR-A [19], MemNet [32], IDN [17], IMDN [22], RFDN [49], and RLFN-S [50] with scale factors of 2, 3, and 4. The quantitative performance comparison on five benchmark test sets is shown in Table 1. Compared to other advanced methods, RNET achieves the best performance for both PSNR and SSIM while reducing the parameters by 70 K compared to the second-best method RLFN-S and by 150 K parameters compared to the third-best method RFDN.

Table 1. Quantitative comparison with other SISR models. The best and second-best performances are in red and blue, respectively.

Method	Scale	Parameters	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM
Bicubic	×2	-	33.66/0.9299	30.24/0.8688	29.56/0.8431	26.88/0.8403	30.80/0.9339
SRCNN [2]		24 K	36.66/0.9542	32.45/0.9067	31.36/0.8879	29.50/0.8946	35.60/0.9663
FSRCNN [18]		12 K	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020	36.67/0.9710
VDSR [3]		666 K	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140	37.22/0.9750
DRRN [30]		298 K	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188	37.88/0.9749
LapSRN [31]		251 K	37.52/0.9591	32.99/0.9124	31.80/0.8952	30.41/0.9103	37.27/0.9740
MemNet [32]		678 K	37.78/0.9597	33.28/0.9142	32.08/0.8978	31.31/0.9195	37.72/0.9740
IDN [17]		553 K	37.83/0.9600	33.30/0.9148	32.08/0.8985	31.27/0.9196	38.01/0.9749
IMDN [22]		694 K	38.00/0.9605	33.63/0.9177	32.19/0.8996	32.17/0.9283	38.88/0.9774
RFDN [49]		534 K	38.05/0.9606	33.68/0.9184	32.16/0.8994	32.12/0.9278	38.88/0.9773
RLFN-S [50]		454 K	38.05/0.9607	33.68/0.9172	32.19/0.8997	32.17/0.9286	-/-
RNET (Ours)		385 K	38.10/0.9612	33.69/0.9190	32.23/0.9009	32.31/0.9289	39.01/0.9778
Bicubic	×3	-	30.39/0.8682	27.55/0.7742	27.21/0.7385	24.46/0.7349	26.95/0.8556
SRCNN [2]		8 K	32.75/0.9090	29.30/0.8215	28.41/0.7863	26.24/0.7989	30.48/0.9117
FSRCNN [18]		13 K	33.18/0.9140	29.37/0.8240	28.53/0.7910	26.43/0.8080	31.10/0.9210
VDSR [3]		666 K	33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279	32.01/0.9340
DRRN [30]		298 K	34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378	32.71/0.9379
MemNet [32]		678 K	34.09/0.9248	30.00/0.8350	28.96/0.8001	27.56/0.8376	32.51/0.9369
IDN [17]		553 K	34.11/0.9253	29.99/0.8354	28.95/0.8013	27.42/0.8359	32.71/0.9381
LAPAR-A [19]		544 K	34.36/0.9267	30.34/0.8421	29.11/0.8054	28.15/0.8523	33.51/.09441
IMDN [22]		703 K	34.36/0.9270	30.32/0.8417	29.09/0.8046	28.17/0.8519	33.61/0.9445
RFDN [49]		541 K	34.41/0.9273	30.34/0.8420	29.09/0.8050	28.21/0.8525	33.67/0.9449
RLFN-S [50]		-	-/-	-/-	-/-	-/-	-/-
RNET (Ours)		400 K	34.56/0.9284	30.43/0.8438	29.17/0.8080	28.36/0.8550	33.85/0.9463
Bicubic	×4	-	28.42/0.8104	26.00/0.7027	25.96/0.6675	23.14/0.6577	24.89/0.7866
SRCNN [2]		8 K	30.48/0.8626	27.50/0.7513	26.90/0.7101	24.52/0.7221	27.58/0.8555
FSRCNN [18]		13 K	30.72/0.8660	27.61/0.7550	26.98/0.7150	24.62/0.7280	27.90/0.8610
VDSR [3]		666 K	31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524	28.83/0.8870
DRRN [30]		298 K	31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638	29.45/0.8946
LapSRN [31]		813 K	31.54/0.8852	28.09/0.7700	27.32/0.7275	25.21/0.7562	29.09/0.8900
MemNet [32]		678 K	31.74/0.8893	28.26/0.7723	27.40/0.7281	25.50/0.7630	29.42/0.8942
IDN [17]		553 K	31.82/0.8903	28.25/0.7730	27.41/0.7297	25.41/0.7632	29.41/0.8942
IMDN [22]		715 K	32.21/0.8948	28.58/0.7811	27.56/0.7353	26.04/0.7838	30.45/0.9075
RFDN [49]		550 K	32.24/0.8952	28.61/0.7819	27.57/0.7360	26.11/0.7858	30.58/0.9089
RLFN-S [50]		470 K	32.23/0.8961	28.61/0.7818	27.58/0.7359	26.15/0.7866	-/-
RNET (Ours)		401 K	32.39/0.8976	28.68/0.7837	27.65/0.7403	26.24/0.7894	30.67/0.9102

4.2.2. Visual Results

To further demonstrate the superiority of our proposed RNET, we display the visual results compared with five advanced methods: SRCNN [2], IDN [17], IMDN [22], RFDN [49], and RLFN-S [50], as shown in Figure 6. The test results show that most methods cannot reconstruct the grid image clearly; by contrast, our RNET is able to obtain sharper results. Taking the top image in Figure 6 as an example, most methods of comparison output heavy aliasing. Earlier methods such as SRCNN [2], IDN [17], and IMDN [22] lose most of the structure due to limited network depth and feature extraction capability. Recent methods RFDN [49] and RLFN-S [50], on the other hand, are able to recover most of the outlines but not the texture details of the image. Compared with that, our method can reconstruct more details and obtain higher visual quality. This can be attributed to the global information extraction capability of the transformer.

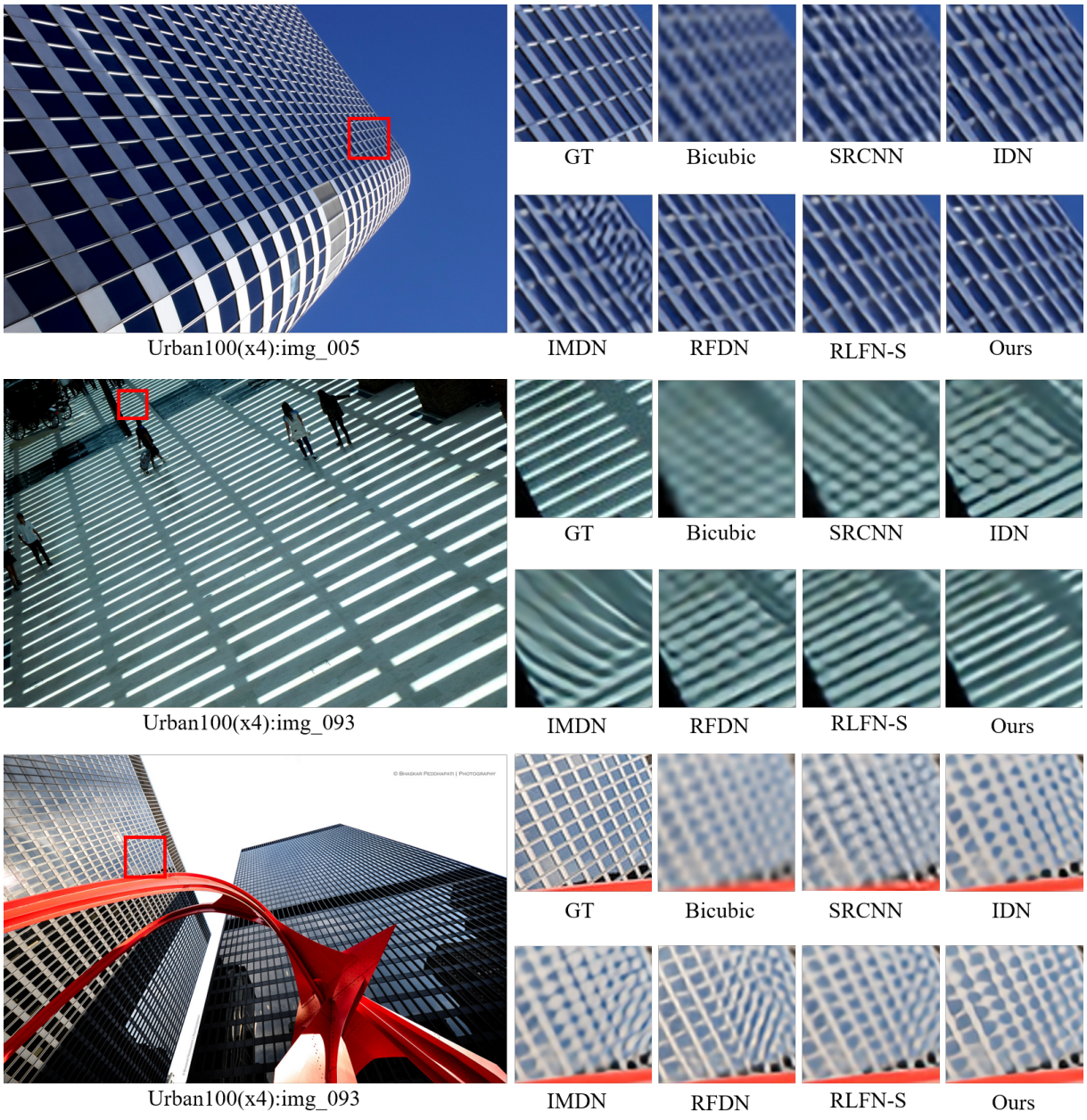


Figure 6. Visual comparison of RNET with other methods at ×4 SR. From the figure, we can see that our method can generate more details of the image.

4.2.3. LAM Results

To observe the range of pixels utilized in SR reconstruction, we resort to a diagnostic tool called Local Attribution Maps (LAMs) [51], which is an attribution method specifically designed for SR. Using the LAM approach, we can identify which pixels contribute the most to the reconstruction of the selected regions. The Diffusion Index (DI) illustrates the range of relevant and utilized pixels, with a higher DI indicating a broader range of used pixels. We compare our model with BSRN [52] and RLFN-S [50], and the LAM results are shown in Figure 7. Thanks to the transformer module, RNET exhibits the widest range of pixels inferred for SR images and achieves the highest DI value. The experimental results are highly consistent with our expectations and, from an interpretability perspective, substantiate that our proposed RNET leverages the long dependencies offered by the transformer, enabling it to utilize more pixels and thus attain better performance.

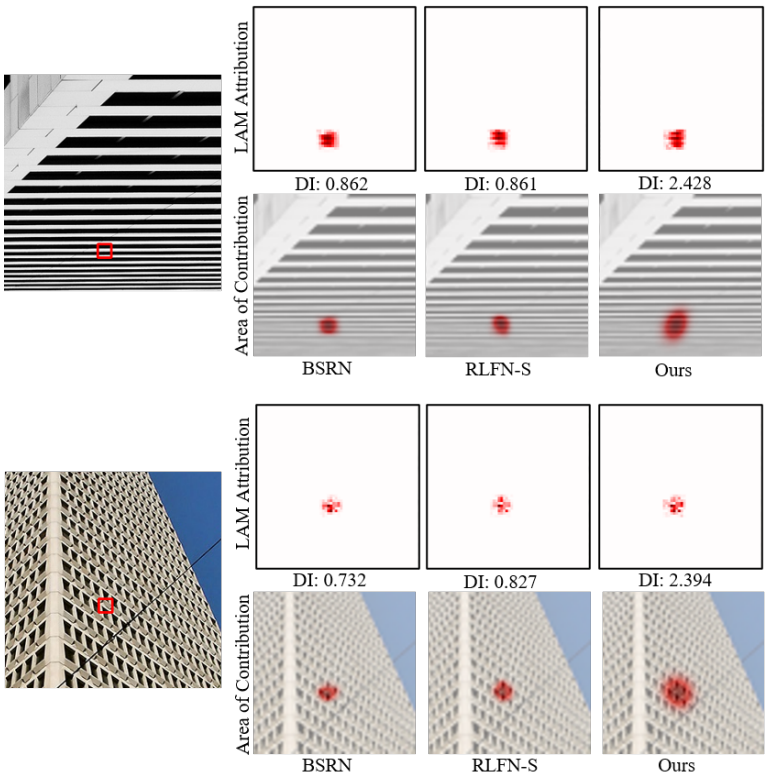


Figure 7. LAM results of our RNET, BSRN, and RLFN-S. We can see that our RNET performs SR reconstruction based on a wider range of pixels.

4.2.4. Computational Cost and Model Complexity Analysis

To fully investigate the efficiency of each model and demonstrate the advantages of our RNET in terms of complexity, we provide a detailed comparison of each model in Table 2, where FLOPs and Activations are the computational results when using the ×4 model with a 1280 × 720 image as input, and runtime and memory are the average results obtained by inference on the BSDS100 dataset tested 10 times using an RTX3070-8G GPU (Taipei, China). To make the results more intuitive, we also add the classical single-image super-resolution networks VDSR [3] and DRRN [30] as references. It can be seen that our proposed RNET has the second-fewest (401 K) number of parameters, the fewest (20.4 G) FLOPs, and the third-fewest Activations (0.17 G) thanks to our use of BSConv, which can efficiently extract effective features while mitigating the computation. Meanwhile, because our proposed RNET uses the transformer structure, it slightly lags behind other convolutional-neural-network-based methods in terms of inference speed and memory usage, but thanks to our use of local residual connections and the improved self-attention module, this gap is not very large and is perfectly acceptable. We visualize the trade-off

between the number of parameters, FLOPs, and performance of each model in Figure 8. We can see from the figure that our RNET achieves a good balance between computational complexity and performance.

Table 2. The results of model efficiency and computational cost.

Method	Parameters	Activations	FLOPs	Runtime	Memory
VDSR [3]	666 K	2.38 G	612.6 G	0.037 s	340 MB
LapSRN [31]	813 K	0.58 G	149.4 G	0.020 s	360 MB
DRRN [30]	298 K	24.89 G	6796.9 G	0.556 s	827 MB
IMDN [22]	715 K	0.15 G	40.9 G	0.015 s	228 MB
RLFN-S [50]	470 K	0.12 G	25.6 G	0.018 s	170 MB
RNET (Ours)	401 K	0.17 G	20.4 G	0.022 s	230 MB

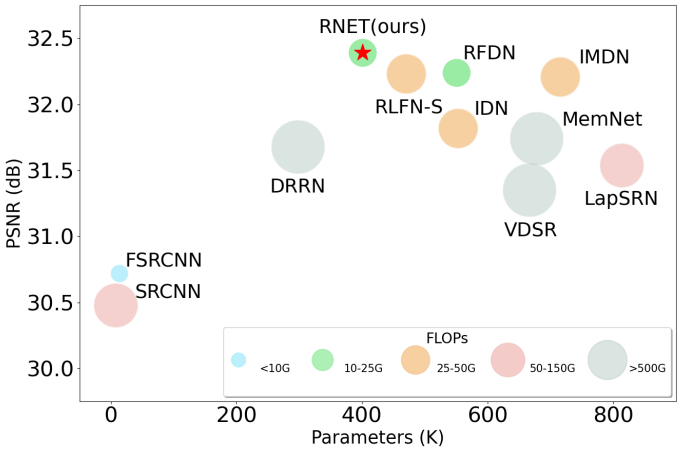


Figure 8. A comparison of performance and model complexity on Set5 dataset; the upscale factor is ×4.

4.2.5. Comparison with Other Transformer-Based Methods

Currently, there are two mainstream approaches to solve the problem of high computation brought by vision transformers: one is to use a sliding window, also known as a swin transformer, such as SwinIR [11]; the other approach is to improve the self-attention part—for example, ESRT [53] uses the splitting factor approach to reduce the computational burden brought by the self-attention. The RNET proposed in this paper uses cross-channel self-attention to achieve the same purpose. We provide a detailed comparison of these three approaches in Table 3. It can be seen that our RNET has better performance than ESRT on the Set5 dataset and achieves similar performance to SwinIR while having the least number of parameters and FLOPs.

Table 3. Detailed comparison with other transformer-based methods.

Method	Parameters	FLOPs	Set5 PSNR/SSIM	Set14 PSNR/SSIM
SwinIR [11]	897 K	49.6 G	32.44/0.8976	28.77/0.7858
ESRT [53]	751 K	67.7 G	32.19/0.8947	28.69/0.7833
RNET (Ours)	401 K	20.4 G	32.39/0.8976	28.68/0.7837

4.3. Ablation Study

In this subsection, we design a series of ablation experiments to analyze the effectiveness of our proposed network. We first investigate the effects of different network depths and widths on the experimental results. Then, we demonstrate the effectiveness of the ET module and explore the placement of the module. We also compare different activation functions.

4.3.1. The Depth and Width of the Network

The experimental results are shown in Table 4, where depth indicates the number of HFEBs used and width indicates the number of feature channels. From the experimental results, it can be seen that increasing both depth and width can enhance the model ability. And experiments show that network width influences results more than depth. When d increased from 6 to 8, the indicator of Urban100 increased by 0.21 db, but when d increased from 8 to 10, the indicator only increased by 0.05 db, while at the same time, the parameters increased to 479 K. And when w increased from 32 to 48, the indicator on Urban100 increased by 0.43 dB, but from 48 to 56, the indicator only showed a slight increase or even a decrease. But the parameters reach 513 K. From Table 4, when d = 10 and w = 56, the PSNR on all five benchmark test sets are improved, but the parameters of the network reach 638 K. In order to better balance the performance of the network with the parameters, we finally set the depth and width of the network to 8 and 48, respectively.

Table 4. Ablation studies of different depths and widths of the network, where d indicates the number of HFEBs used, and w indicates the number of feature channels.

Method	Parameters	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM
d = 6 w = 32	139 K	37.94/0.9607	33.44/0.9168	32.10/0.8994	31.72/0.9239	38.47/0.9767
d = 6 w = 48	291 K	38.04/0.9610	33.60/0.9184	32.18/0.9005	32.10/0.9270	38.78/0.9775
d = 6 w = 56	387 K	38.07/0.9611	33.63/0.9187	32.21/0.9008	32.20/0.9280	38.93/0.9777
d = 8 w = 32	183 K	37.92/0.9607	33.50/0.9172	32.13/0.8997	31.88/0.9251	38.65/0.9771
d = 8 w = 48	385 K	38.10/0.9612	33.69/0.9190	32.23/0.9009	32.31/0.9289	39.01/0.9778
d = 8 w = 56	513 K	38.11/0.9612	33.65/0.9187	32.23/0.9009	32.30/0.9293	38.99/0.9778
d = 10 w = 32	228 K	38.01/0.9609	33.51/0.9173	32.16/0.9001	32.03/0.9265	38.73/0.9773
d = 10 w = 48	479 K	38.10/0.9612	33.70/0.9188	32.24/0.9011	32.36/0.9295	39.00/0.9779
d = 10 w = 56	638 K	38.12/0.9613	33.74/0.9195	32.25/0.9014	32.47/0.9305	39.12/0.9779

4.3.2. Ablation Study of Efficient Transformer (ET) Block

We performed ablation experiments to separately verify the impact of different convolutions in our proposed ET module and the effectiveness of the ET module. Within the two key components of the ET module, MBTA and GBFN, we replaced BSConv with traditional Depth-Wise Convolution (DWConv). The experimental results are shown in Table 5. Using BSConv in the ET module introduces an insignificant increase in parameters, but it improves the model's performance on all five benchmark test sets. Therefore, we employed BSConv for channel feature extraction in the final ET module.

Subsequently, to verify the effectiveness of the ET module, we placed the ET module at Position 1 (P1 in Figure 9) or Position 2 (P2 in Figure 9) or removed the ET module altogether. The experimental results are shown in Table 5. After removing the ET module, the network's performance significantly decreased on all five benchmark datasets. When the ET module was placed at Position 2, there was a slight decline in the results on these datasets. These results demonstrate that the ET module at Position 1 can effectively enhance the capacity of the HFEB model. Consequently, in the final model, we positioned the ET module at Position 1.

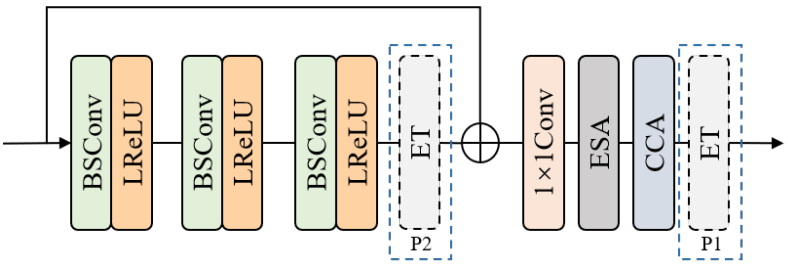


Figure 9. The structure of HFEB and possible positions of ET in HFEB.

Table 5. Quantitative comparison of the position of ET block and different convolutions used in ET.

Ablation	Variant	Parameters	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM
HFEB	ET in P1	385 K	38.10/0.9612	33.69/0.9190	32.23/0.9009	32.31/0.9289	39.01/0.9778
	ET in P2	380 K	38.06/0.9612	33.66/0.9187	32.22/0.9008	32.29/0.9289	38.96/0.9778
	without ET	135 K	37.75/0.9601	33.22/0.9152	32.01/0.8983	31.39/0.9209	38.01/0.9756
ET	BSConv	385 K	38.10/0.9612	33.69/0.9190	32.23/0.9009	32.31/0.9289	39.01/0.9778
	DWConv	384 K	38.06/0.9601	33.63/0.9189	32.22/0.9003	32.23/0.9273	38.86/0.9766

4.3.3. Different Activation Function in HFEB

We designed an ablation experiment to explore the effectiveness of the activation function in HFEB. The results in Table 6 show that different activation functions significantly affect the performance of the model. Among these activation functions, LeakyRelu obtains a significant performance gain, especially on the Urban100 and Manga109 datasets. Therefore, we chose LeakyReLU as the activation function in our model.

Table 6. Ablation study using different activation functions in HFEB.

Activate Function	Parameters	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM
GELU	385 K	38.08/0.9611	33.66/0.9184	32.21/0.9008	32.24/0.9287	38.94/0.9777
ReLU	385 K	38.08/0.9612	33.69/0.9189	32.21/0.9008	32.23/0.9285	38.90/0.9777
LeakyReLU	385 K	38.10/0.9612	33.69/0.9190	32.23/0.9009	32.31/0.9289	39.01/0.9778

4.3.4. Effectiveness of ESA and CCA Blocks

In order to verify the effectiveness of the ESA and CCA modules, we sequentially removed these two modules from the HFEB network proposed in this paper, re-trained the ×4 scale model, then tested the network on each of the five benchmark datasets. And the experimental results are shown in Table 7. The experiments show that after removing the two attention modules, the performance of our network on all the five benchmark datasets shows some degradation. At the same time, we find that both the ESA and CCA modules hardly increase the complexity of the whole model—only increasing the number of parameters by about 2%—and result in a PSNR gain of about 0.1 db on each dataset. Thus, the ESA with CCA module used in this paper is able to give the model a performance boost at a small cost.

Table 7. Ablation studies of the effectiveness of ESA and CCA blocks, where ✓ indicates the block is used, and ✗ indicates the block is not used.

ESA	CCA	Parameters	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM
✗	✗	393 K	32.29/0.8964	28.61/0.7823	27.59/0.7388	26.14/0.7863	30.47/0.9081
✗	✓	397 K	32.31/0.8967	28.62/0.7823	27.60/0.7388	26.16/0.7863	30.58/0.9094
✓	✗	397 K	32.32/0.8970	28.65/0.7829	27.61/0.7393	26.18/0.7873	30.64/0.9099
✓	✓	401 K	32.39/0.8976	28.68/0.7837	27.65/0.7403	26.24/0.7894	30.67/0.9102

5. Conclusions

In this paper, we propose a novel Residual Network with Efficient Transformer (RNET) for lightweight single-image super-resolution. RNET achieves a good balance between performance and model parameters. Specifically, we adopt a local residual connection structure as the backbone network for deep feature extraction modules. Compared to the widely used knowledge distillation structure, this structure significantly enhances inference speed. Concurrently, we utilize BSConv to replace traditional convolutional operations, which greatly reduces the number of required parameters. To fully exploit global information while avoiding excessive computational redundancy, we introduce an

efficient transformer module that comprises channel-wise self-attention mechanisms and an efficient gated feed-forward module. Experimental results demonstrate that our method can effectively utilize a broader range of pixel information, as verified by the evaluation results from LAM. To further enhance the model's representative capacity and accuracy, we also integrate a hybrid channel and spatial attention module. Extensive experiments show that our method achieves the best performance across five commonly used benchmark test sets. Concurrently, images reconstructed using our method exhibit the best visual results and richest details. Of particular note, RNET possesses only 400 K parameters and 20 G FLOPs: successfully achieving outstanding balance between performance and model complexity. Moreover, when processing single-image inference, RNET takes less than 30 milliseconds and occupies only 230 MB of memory. These characteristics hold significant value for promoting the practical application of single-image super-resolution techniques. In the future, we will further optimize the self-attention module to reduce memory usage. And we will expand the model so that it can be applied to more-challenging tasks, such as classical image super-resolution and real-world image super-resolution.

Author Contributions: Conceptualization, F.Y. and Z.Z.; methodology, F.Y.; software, F.Y.; validation, F.Y., S.L. and Z.Z.; formal analysis, F.Y. and S.L.; investigation, F.Y. and S.L.; resources, F.Y., Z.Z. and Y.S.; data curation, F.Y.; writing—original draft preparation, F.Y., Y.S. and Z.Z.; writing—review and editing, Z.Z. and Y.S.; visualization, F.Y. and S.L.; supervision, Y.S.; project administration, F.Y. and Z.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Freeman, W.T.; Pasztor, E.C.; Carmichael, O.T. Learning low-level vision. *Int. J. Comput. Vis.* **2000**, *40*, 25–47. [CrossRef]
2. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 295–307. [CrossRef] [PubMed]
3. Kim, J.; Lee, J.K.; Lee, K.M. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
4. Zhang, Y.; Tian, Y.; Kong, Y.; Zhong, B.; Fu, Y. Residual dense network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Vattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2472–2481.
5. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6000–6010.
6. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
7. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.
8. Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; Gao, W. Pre-trained image processing transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12299–12310.
9. Wang, Z.; Cun, X.; Bao, J.; Zhou, W.; Liu, J.; Li, H. Uformer: A general u-shaped transformer for image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 17683–17693.
10. Li, W.; Lu, X.; Lu, J.; Zhang, X.; Jia, J. On efficient transformer and image pre-training for low-level vision. *arXiv* **2021**, arXiv:2112.10175.
11. Liang, J.; Cao, J.; Sun, G.; Zhang, K.; Van Gool, L.; Timofte, R. Swinir: Image restoration using swin transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 1833–1844.
12. Lim, B.; Son, S.; Kim, H.; Nah, S.; Mu Lee, K. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 136–144.

13. Zhang, K.; Zuo, W.; Zhang, L. Learning a single convolutional super-resolution network for multiple degradations. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3262–3271.
14. Chu, X.; Zhang, B.; Xu, R. Multi-objective reinforced evolution in mobile neural architecture search. In Proceedings of the Computer Vision—ECCV 2020 Workshops, Glasgow, UK, 23–28 August 2020; Proceedings, Part IV; Springer: Berlin/Heidelberg, Germany, 2021; pp. 99–113.
15. Kim, J.; Lee, J.K.; Lee, K.M. Deeply-recursive convolutional network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1637–1645.
16. Liu, J.; Zhang, W.; Tang, Y.; Tang, J.; Wu, G. Residual feature aggregation network for image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2359–2368.
17. Hui, Z.; Wang, X.; Gao, X. Fast and accurate single image super-resolution via information distillation network. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 723–731.
18. Dong, C.; Loy, C.C.; Tang, X. Accelerating the super-resolution convolutional neural network. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part II 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 391–407.
19. Li, W.; Zhou, K.; Qi, L.; Jiang, N.; Lu, J.; Jia, J. Lapar: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 20343–20355.
20. Haase, D.; Amthor, M. Rethinking depthwise separable convolutions: How intra-kernel correlations lead to improved mobilenets. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 14600–14609.
21. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
22. Hui, Z.; Gao, X.; Yang, Y.; Wang, X. Lightweight image super-resolution with information multi-distillation network. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 2024–2032.
23. Tian, C.; Xu, Y.; Zuo, W. Image denoising using deep CNN with batch renormalization. *Neural Netw.* **2020**, *121*, 461–473. [CrossRef] [PubMed]
24. Pan, J.; Ren, W.; Hu, Z.; Yang, M.H. Learning to deblur images with exemplars. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 1412–1425. [CrossRef] [PubMed]
25. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
26. Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y. Image super-resolution using very deep residual channel attention networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 286–301.
27. Zhou, S.; Zhang, J.; Zuo, W.; Loy, C.C. Cross-scale internal graph neural network for image super-resolution. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 3499–3509.
28. Wang, X.; Xie, L.; Dong, C.; Shan, Y. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 1905–1914.
29. Bashivan, P.; Rish, I.; Yeasin, M.; Codella, N. Learning representations from EEG with deep recurrent-convolutional neural networks. *arXiv* **2015**, arXiv:1511.06448.
30. Tai, Y.; Yang, J.; Liu, X. Image super-resolution via deep recursive residual network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3147–3155.
31. Lai, W.S.; Huang, J.B.; Ahuja, N.; Yang, M.H. Deep laplacian pyramid networks for fast and accurate super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 624–632.
32. Tai, Y.; Yang, J.; Liu, X.; Xu, C. Memnet: A persistent memory network for image restoration. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4539–4547.
33. Zhang, X.; Zeng, H.; Zhang, L. Edge-oriented convolution block for real-time super resolution on mobile devices. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual, 20–24 October 2021; pp. 4034–4043.
34. Han, K.; Rezende, R.S.; Ham, B.; Wong, K.Y.K.; Cho, M.; Schmid, C.; Ponce, J. Scnet: Learning semantic correspondence. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1831–1840.
35. Wang, S.; Zhou, T.; Lu, Y.; Di, H. Contextual transformation network for lightweight remote-sensing image super-resolution. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–13. [CrossRef]
36. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part I 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–229.
37. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 568–578.

38. Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; Zhang, L. Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 22–31.
39. Xiao, T.; Singh, M.; Mintun, E.; Darrell, T.; Dollár, P.; Girshick, R. Early convolutions help transformers see better. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 30392–30400.
40. Wang, S.; Zhou, T.; Lu, Y.; Di, H. Detail-preserving transformer for light field image super-resolution. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36, pp. 2522–2530.
41. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.H. Restormer: Efficient transformer for high-resolution image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5728–5739.
42. Agustsson, E.; Timofte, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 126–135.
43. Bevilacqua, M.; Roumy, A.; Guillemot, C.; Alberi-Morel, M.L. Low-Complexity Single-Image Super-Resolution Based on Nonnegative Neighbor Embedding. In Proceedings of the British Machine Vision Conference, Surrey, UK, 3–7 September 2012; pp. 1–10. Available online: https://people.rennes.inria.fr/Aline.Roumy//publi/12bmvc_Bevilacqua_lowComplexitySR.pdf (accessed on 26 December 2023).
44. Zeyde, R.; Elad, M.; Protter, M. On single image scale-up using sparse-representations. In Proceedings of the Curves and Surfaces: 7th International Conference, Avignon, France, 24–30 June 2010; Revised Selected Papers 7; Springer: Berlin/Heidelberg, Germany, 2012; pp. 711–730.
45. Martin, D.; Fowlkes, C.; Tal, D.; Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Proceedings of the Proceedings Eighth IEEE International Conference on Computer Vision, ICCV, Vancouver, BC, Canada, 7–14 July 2001; Volume 2, pp. 416–423.
46. Huang, J.B.; Singh, A.; Ahuja, N. Single image super-resolution from transformed self-exemplars. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5197–5206.
47. Matsui, Y.; Ito, K.; Aramaki, Y.; Fujimoto, A.; Ogawa, T.; Yamasaki, T.; Aizawa, K. Sketch-based manga retrieval using manga109 dataset. *Multimed. Tools Appl.* **2017**, *76*, 21811–21838. [CrossRef]
48. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
49. Liu, J.; Tang, J.; Wu, G. Residual feature distillation network for lightweight image super-resolution. In Proceedings of the Computer Vision—ECCV 2020 Workshops, Glasgow, UK, 23–28 August 2020; Proceedings, Part III 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 41–55.
50. Kong, F.; Li, M.; Liu, S.; Liu, D.; He, J.; Bai, Y.; Chen, F.; Fu, L. Residual local feature network for efficient super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 766–776.
51. Gu, J.; Dong, C. Interpreting super-resolution networks with local attribution maps. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9199–9208.
52. Li, Z.; Liu, Y.; Chen, X.; Cai, H.; Gu, J.; Qiao, Y.; Dong, C. Blueprint separable residual network for efficient image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 833–843.
53. Lu, Z.; Li, J.; Liu, H.; Huang, C.; Zhang, L.; Zeng, T. Transformer for single image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 457–466.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Pixel-Level Degradation for Text Image Super-Resolution and Recognition

Xiaohong Qian [1], Lifeng Xie [1], Ning Ye [1], Renlong Le [2] and Shengying Yang [1,*]

[1] School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China
[2] Ningbo Sea Fresh Information Technology Co., Ningbo 315500, China; l.renlong@haishangxian.cn
* Correspondence: syyang@zust.edu.cn

Abstract: In the realm of image reconstruction, deep learning-based super-resolution (SR) has established itself as a prevalent technique, particularly in the domain of text image restoration. This study aims to address notable deficiencies in existing research, including constraints imposed by restricted datasets and challenges related to model generalization. Specifically, the goal is to enhance the super-resolution network's reconstruction of scene text image super-resolution and utilize the generated degenerate dataset to alleviate issues associated with poor generalization due to the sparse scene text image super-resolution dataset. The methodology employed begins with the degradation of images from the MJSynth dataset, using a stochastic degradation process to create eight distinct degraded versions. Subsequently, a blank image is constructed, preserving identical dimensions to the low-resolution image, with each pixel sourced randomly from the corresponding points across the eight degraded images. Following several iterations of training via Finetune, the LR-HR method is applied to the TextZoom dataset. The pivotal metric for assessment is optical character recognition (OCR) accuracy, recognized for its fundamental role in gauging the pragmatic effectiveness of this approach. The experimental findings reveal a notable enhancement in OCR accuracy when compared to the TBSRN model, yielding improvements of 2.4%, 2.3%, and 4.8% on the TextZoom dataset. This innovative approach, founded on pixel-level degradation, not only exhibits commendable generalization capabilities but also demonstrates resilience in confronting the intricate challenges inherent to text image super-resolution.

Keywords: degradation model; scene text image; super resolution

1. Introduction

Scene text recognition (STR) involves the process of converting images of text, such as from books, newspapers, and other sources taken in natural scenes with less than ideal lighting and focus, into digital text sequences. With the ubiquity of smartphones and similar devices, the internet has been inundated with such scene text images. However, largely due to hardware limitations and network bandwidth constraints, a significant portion of these images exist in low resolution. This low clarity presents a considerable challenge for text recognition.

While modern optical character recognition models exhibit impressive accuracy rates on high-resolution (HR) text images [1,2], they still grapple with low accuracy when it comes to low-resolution (LR) images. To ameliorate this, many in the research community have turned their focus towards applying super-resolution techniques to these low-resolution scene text images.

Several researchers have proffered diverse methodologies for text image super-resolution. Dong [3] proposed a method that synergizes deep learning with traditional sparse-coding-based super-resolution techniques. He applied the SRCNN network to scene text images and employed an end-to-end training approach, effectively enhancing the super-resolution results

and overall quality. Wang et al. [4] took a multi-pronged approach: they not only introduced the TextZoom dataset but also floated the TSRN network. This network ingeniously integrates Spatial Transformer Networks and Bi-directional LSTM (BLSTM) mechanisms, facilitating a more refined extraction of sequence information and superior low-level reconstruction. Building on TSRN's foundations, Chen et al. [5] unveiled the TBSRN network. Their novel approach, centered on a text-sensitive loss, ensures that the network's focus remains predominantly on the text rather than any irrelevant background. TSRGAN [6] employs a cutting-edge generative adversarial network, introduces a triple-attention mechanism to enhance the network's representation ability, and employs traditional wavelet loss to reconstruct sharper character edges. TATT [7] chooses the textual a priori semantic information, which is a large amount of workload but with a better yield, to assist the network for training, and utilizes a global attention mechanism to semantically steer the text prior to the text reconstruction process. These pioneering methods have undeniably introduced innovative paradigms and achieved commendable outcomes in text image super-resolution. However, challenges persist. Among them are the limited generalization capabilities of models and the constraints of having only a handful of training datasets, which hampers the robustness of these systems. These challenges highlight the need for continued optimization and refinement in the field.

In the realms of text image processing research and application, one recurring limitation stems from the paucity of suitable datasets and the intricacies of labeling. This constraint not only restricts the depth of research but also impedes the widespread adoption of super-resolution techniques. Existing datasets, such as SVT [8], CTW [9], and CUTE80 [10], have been primarily tailored for scene text recognition, rendering them valuable for their intended purposes. However, their emphasis on recognition, as opposed to super-resolution, presents a unique challenge. The absence of dedicated super-resolution datasets not only hampers algorithm generalization and transferability but also curtails their real-world practicality.

To address the dataset challenge, certain approaches strive to bolster model generalization [11], often by instructing deep denoising networks in image content reconstruction. Nevertheless, it is worth noting that the primary cause of text image degradation transcends mere noise. Therefore, these methods may exhibit limited efficacy in resolving scene-specific degradation issues. In a bid to surmount these constraints and bridge the dataset gap, synthetic datasets have emerged as a viable solution. Image degradation stands as a widely adopted synthetic method, simulating real-world text image degradation scenarios, including scenarios involving images captured out of focus or under electromagnetic influence, achieved through the introduction of blurring kernels and noise manipulation.

Nonetheless, prior endeavors in this domain have frequently relied on double or triple downsampling, or intricate blur kernels, to generate degraded images [12–15]. While these strategies provide versatility, they may not faithfully replicate real-world text image degradation. Such deviations can encumber the model's capacity to discern specific degraded image features, ultimately affecting the efficacy of super-resolution algorithms. Consequently, the accurate simulation of real-world degradation scenarios has arisen as a pressing challenge in the realm of super-resolution research. Conquering this challenge is pivotal for advancing the field and ensuring the practical applicability of super-resolution techniques across diverse text image processing applications.

In response to the aforementioned challenges, an advanced pixel-level degradation process is introduced in this study, encompassing blur, noise, and stochastic strategies. This approach micro-targets degradation to each pixel, amplifying the scope of image degradation. It aims to empower the model to holistically comprehend and enhance its super-resolution reconstruction capabilities for text images. Preliminary results indicate that datasets refined through this advanced degradation process considerably bolster the model's super-resolution capabilities, attesting to the efficacy and robustness of the degradation procedure. Comparative experiments demonstrate that training and testing using the advanced degradation process substantively boost model performance. Notably,

in managing real-world low-resolution text images, the restoration detail and recognition rates markedly surpass those of the contemporary TBSRN model.

The paper is structured as follows: Section 2 elaborates on the fundamental principles of degradation and transfer learning. In Section 3, the network's architecture is comprehensively explained. Section 4 offers visualizations and numerical comparisons of the approach's results, showcasing its viability. Lastly, Section 5 provides a concise summary of the entire paper and draws conclusions.

2. Theoretical Foundations

2.1. Noise

2.1.1. Gaussian Noise

Gaussian noise is a prevalent random interference encountered in digital signal and image processing, typically originating from imaging tools and transmission pathways. Such interference compromises image clarity, necessitating denoising interventions. Its applications span areas like image denoising, enhancement, and restoration. This study harnesses a three-dimensional Gaussian noise model with zero mean, allowing manual modulation of noise intensity by adjusting the standard deviation, symbolized by the covariance matrix Σ. It is worth highlighting that the generalized Gaussian noise model can morph into additive Gaussian white noise and grayscale Gaussian noise, though these are extreme cases. Additive Gaussian white noise is characterized by its independent components, each obeying Gaussian distribution principles, leading to its alternative name: Gaussian white noise. Its probability density function follows a normal distribution, with statistical features including zero mean and constant variance. Grayscale Gaussian noise parallels the additive Gaussian white noise in its statistical attributes, maintaining zero mean and uniform variance. Its power spectral density remains constant across frequencies, affecting only pixel grayscale values without altering their hues. Acknowledging the diverse and unpredictable nature of degradation, varying noise intensities are incorporated in the study. Noise levels are uniformly sampled within the range $\{1/255, 2/255, \ldots, 5/255\}$, generating samples under three scenarios with respective probabilities of 0.4, 0.4, and 0.2. The probability density function of Gaussian noise can be represented as follows:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-(z-\mu)^2/2\sigma^2\right\}. \quad (1)$$

where 'exp' denotes the exponential function based on the natural constant e; 'z' is the grayscale image value; 'μ' signifies the expected value of z; and 'σ' stands for the standard deviation of z.

2.1.2. JPEG Compression Noise

JPEG, short for Joint Photographic Experts Group, is an image compression standard tailored specifically for compressing natural, real-world full-color or grayscale imagery. Its primary objective is to minimize data redundancy in images, facilitating efficient data storage or transmission. The JPEG compression mechanism can introduce various types of noise, among which quantization noise and encoding noise are most prevalent. Quantization noise emerges due to errors in the quantization process post the DCT (discrete cosine transform) phase in JPEG compression. High-frequency components within images tend to be more susceptible to these quantization inaccuracies, leading to image distortions and the appearance of noise. Encoding noise, on the other hand, arises when quantized coefficients undergo encoding. To conserve the data stream, the encoding phase employs compression techniques such as Huffman coding and run-length encoding. While these methods manage to reduce the bitrate without compromising image integrity, they can inadvertently add some noise.

Recognizing that low-resolution text images in real-world settings often undergo significant compression, this study sets the image compression quality factor range between [20, 65]. This choice mirrors actual degradation conditions and accentuates the randomness

of degradation. Moreover, it introduces pronounced artifacts, expanding the degradation spectrum. The fundamental equation of the JPEG compression method can be depicted as follows:

$$F_{i,j} = \text{round}(\frac{f_{i,j}}{Q_{i,j}}), \tag{2}$$

$$x(i,j) = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)f_{u,v}\cos(\frac{(2i+1)u\pi}{16})\cos(\frac{(2j+1)v\pi}{16}). \tag{3}$$

In Equation (2),$f_{i,j}$ represents the coefficients after the discrete cosine transformation; 'round' denotes the rounding function, rounding computational outcomes to the nearest integer; $Q_{i,j}$ signifies elements within the quantization matrix; and $F_{i,j}$ designates the coefficients post-quantization. In Equation (3), $C(u)$ and $C(v)$ represent transformation coefficients; for instances when $u = 0$ or $v = 0$, both $C(u)$ and $C(v)$ equate to $1/\sqrt{2}$, otherwise, they are set to 1; $x(i,j)$ stands for pixel values of the decompressed image.

2.2. Gaussian Blur

Gaussian blurring serves as a technique in image processing, employing convolution and Gaussian kernels to perform its function. It is particularly effective at reducing high-frequency noise within images and softening finer details, contributing to a smoother and more natural appearance. Isotropic Gaussian blurring is among the most straightforward methods in this category, leveraging Gaussian filters to create a weighted average of each pixel value in conjunction with its neighbors based on a Gaussian distribution. This results in effective image smoothing. However, this approach treats details in all directions equivalently, potentially leading to loss of nuances along certain orientations. To mitigate this limitation, anisotropic Gaussian blurring is developed. It accounts for variations in image details across different directions, using an asymmetrical Gaussian filter to apply a directionally weighted average to each pixel and its surrounding pixels. This allows for better preservation of image detail and texture.

Traditional SISR (Single-Image Super-Resolution) often employs isotropic Gaussian kernels with standard deviations as its primary blurring mechanism, though there is substantial room for improvement in terms of mimicking degradation. For a more realistic simulation of natural degradation processes, this study incorporates both isotropic and anisotropic Gaussian blurring kernels. As for the selection of kernel sizes, this study randomly samples within a range of $\{9 \times 9, 12 \times 12, \ldots 36 \times 36\}$, while the rotation angles for isotropic Gaussian kernels are randomly sampled from the interval $[0, \pi]$. The equations representing isotropic and anisotropic Gaussian blurring are as follows:

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{4}$$

In Equation (4), x and y denote the spatial coordinates of the Gaussian blurring kernel, and σ signifies its standard deviation.

$$G(x,y,\sigma_x,\sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y}e^{-\frac{1}{2}(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2})}. \tag{5}$$

Equation (5) specifies x and y as the spatial coordinates, while σ_x and σ_y represent the standard deviations of the Gaussian kernel along the x and y axes, respectively.

2.3. Transfer Learning

Transfer learning is a widely acknowledged technique across numerous machine learning domains, most notably in computer vision and natural language processing. The core essence lies in harnessing the parameters of a sufficiently trained model and integrating them into a fresh model, expediting the learning process of the latter. This tactic can not only accelerate the learning pace but also enhance the generalization capabilities of the

new model. This is especially advantageous when computational resources are scarce, optimizing the training efficiency. The "pretrain-finetune" method represents a popular form of transfer learning, and it can be dissected further into several subcategories.

Li and colleagues [16] devised a strategy that utilizes an L2 regularization loss to manage the finetuning phase. By ensuring the model parameters across both stages are closely aligned, it prevents potential overfitting during the fine-tuning period. The L2 regularization loss can be depicted as:

$$\Omega(w) = \frac{\alpha}{2}\left\|w - w^0\right\|_2^2, \tag{6}$$

In Equation (6), w embodies the parameter vector encompassing all network parameters designed for the target task. α is a regularization factor that dictates the intensity of the penalty term, while w^0 signifies the parameter vector of the pretrained model on the source problem.

On the other hand, Jang et al. [16] introduced a weighted feature matching loss, articulated as:

$$\mathcal{L}_{wfm}^{m,n}(\theta|x, w^{m,n}) = \frac{1}{HW}\sum_c w_c^{m,n}\sum_{i,j}\left(r_\theta(T_\theta^n(x))_{c,i,j} - S^m(x)_{c,i,j}\right)^2. \tag{7}$$

Equation (7) outlines $H \times W$ as the spatial dimensions of $S^m(x)$ and $r_\theta(T_\theta^n(x))$. $w_c^{m,n}$ represents the non-negative weight of channel c, with $\sum_c w_c^{m,n} = 1$. r_θ stands as a linear transformation parameter; $T_\theta^n(x)$ represents the intermediate feature map of the n-th layer in the target network, while $S^m(x)$ denotes the intermediate feature mapping of the m-th layer in the pre-trained source network. This particular loss accentuates channels based on their utility in the target task.

Through the aforementioned loss functions, models can autonomously enact the rules of knowledge transfer, accommodating the discrepancies in architecture and tasks between the source and target, eliminating the necessity for manual transfer adjustments. Ultimately, by discerning the weight of each feature and the corresponding weights of source and target layers, one can achieve selective transfer of the pretrained model.

3. Methods

The overall process of text image super-resolution based on pixel-level degradation can be visualized as depicted in Figure 1. Initially, raw images from the input dataset undergo high-order pixel degradation to produce corresponding low-resolution images. The aim here is to emulate the image quality typically found in real-world scenarios. These images are then passed through a super-resolution network for training, designed to capture high-frequency details and augment image resolution. Later, the low-resolution images from the TextZoom dataset are fed into the trained super-resolution network for fine-tuning, enhancing the model's performance in practical applications. The outcome is a refined super-resolution image. The primary components of this method include the super-resolution network module and the pixel-level high-order degradation module.

3.1. Super Resolution Network Module

The super-resolution network module employs the Hybrid Attention Transformer (HAT) network [17], as visualized in Figure 2. The network mainly consists of a shallow feature extraction layer, a deep feature extraction layer, and an image reconstruction layer.

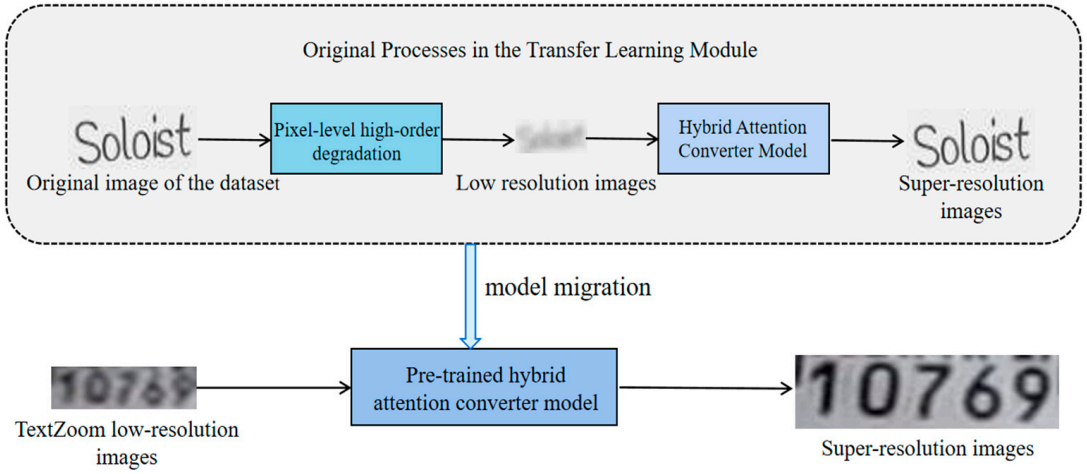


Figure 1. Overall process of text image super-resolution method.

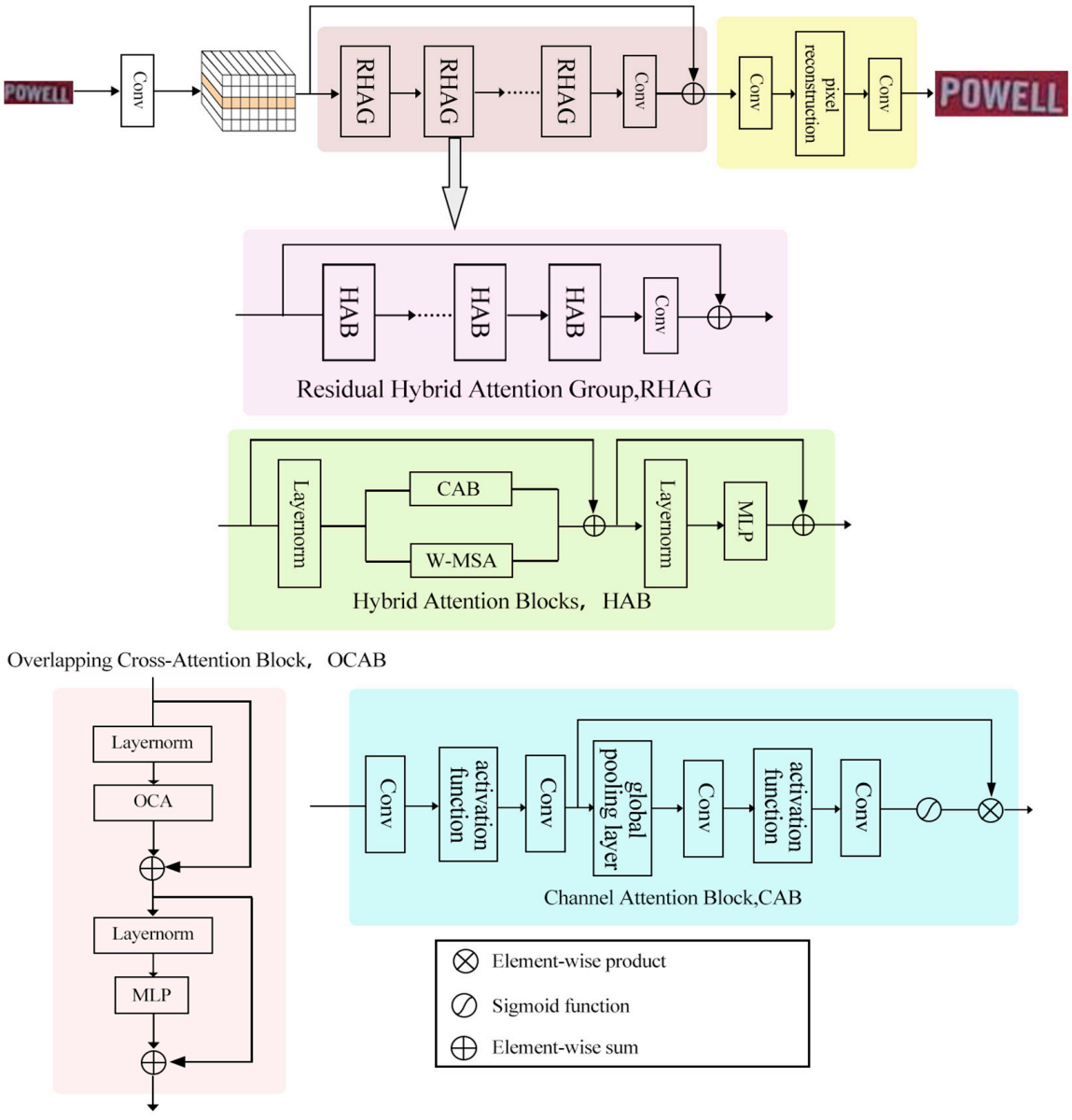


Figure 2. Hybrid Attention Transformer model diagram.

The deep feature extraction layer is composed of N Residual Hybrid Attention Groups (RHAG) and a convolutional layer. The addition of an extra convolutional layer at the conclusion of the deep feature extraction layer serves a pivotal role. This strategic placement enhances the amalgamation of deep feature information, yielding more comprehensive and representative feature maps. It introduces non-linearity, empowering the model to capture intricate patterns and relationships within the features. This, in turn, bolsters its ability to recognize and restore intricate text details during super-resolution. For any given low-resolution image, P_{LR}, an initial convolution layer named C_{SF} extracts its shallow features F_1:

$$F_1 = C_{\mathrm{SF}}(P_{\mathrm{LR}}), \tag{8}$$

This shallow feature is subsequently passed into the deep feature extraction layer C_{DF} yielding the deep feature F_2:

$$F_2 = C_{\mathrm{DF}}(F_1). \tag{9}$$

In the final step, these deep features are summed element-wise with the initial shallow-level features and processed in the image reconstruction layer to generate the final super-resolution image.

Within the deep feature extraction layer, the Residual Hybrid Attention Group comprises M Hybrid Attention Blocks (HAB) and an Overlapping Cross-Attention Block (OCAB). Inside the HAB, incoming feature maps undergo a series of transformations. They initially traverse a Layer Normalization (LN) layer, ensuring input stability for subsequent computations. Parallel operations unfold within the Channel Attention Block (CAB) and the Window-based Multi-Head Self-Attention (W-MSA) module. The CAB captures global information by computing channel attention weights, facilitating the extraction of structured features within the image. Concurrently, the W-MSA divides the image into non-overlapping windows and executes self-attention operations within each window, effectively capturing local dependencies. The resultant outputs from both branches are combined element-wise with the initial feature maps, augmenting their representational capacity. Subsequently, another normalization layer and a Multi-Layer Perceptron (MLP) layer further refine the features before conveying them for subsequent processing. This computation sequence can be represented as:

$$F_m = \mathrm{LN}(F_l), \tag{10}$$

$$F_n = \mathrm{W-MSA}(F_m) + \alpha \mathrm{CAB}(F_m) + F_l, \tag{11}$$

$$Y = \mathrm{MLP}(\mathrm{LN}(F_n)) + F_n. \tag{12}$$

Equation (10) signifies that F_m denotes intermediary features while F_l is the incoming feature map. In Equation (11), F_n represents intermediary features, and α is a minor constant. Equation (12) illustrates that Y indicates the HAB's output and MLP corresponds to the Multi-Layer Perceptron module. The OCAB, on the other hand, incorporates the Overlapping Cross-Attention (OCA) layer along with an additional MLP layer. This particular design choice facilitates cross-window connections, thereby enhancing self-attention mechanisms within each window. The OCA layer partitions windows into larger sizes while maintaining a consistent step size, allowing for overlapping regions in windows that would otherwise remain non-overlapping. This innovation encourages key/value computation from more extensive fields and reinforces interactions between features in neighboring windows. Through the integration of these advanced mechanisms, the deep feature extraction layer achieves a delicate equilibrium between capturing global context and preserving local intricacies. This, in turn, significantly augments the model's capacity to restore intricate details in text images during the super-resolution process. The actions of the Residual Hybrid Attention Group, facilitated by the modules mentioned earlier, can be summarized as:

$$F_{i-1,j} = C_{\mathrm{HAB}_{i,j}}(F_{i-1,j-1}), j = 1, 2 \ldots, M\,, \tag{13}$$

$$F_i = C_{\mathrm{conv}_i}(C_{\mathrm{OCAB}_i}(F_{i-1,M})) + F_{i-1}. \tag{14}$$

Equation (13) elaborates that $F_{i-1,0}$ represents the input features of the i-th RHAG, with C_{HAB} symbolizing the Hybrid Attention Block. In Equation (14), $F_{i-1,j}$ is the j-th output feature of the j-th Hybrid Attention Block within the i-th Residual Hybrid Attention Group.

3.2. Pixel-Level High-Order Degradation Module

In an effort to simulate intricate real-world degradation while also taking into account the shooting scenario of low-resolution images, this study employed a combination of Gaussian noise, Joint Photographic Experts Group (JPEG) noise, and Gaussian blur kernel to establish the overall degradation process. Figure 3 showcases the results generated after the implementation of a pixel-level high-order degradation module. The magnified image in the center represents the final degraded image, while the surrounding eight images represent eight different low-resolution images produced from a single high-resolution image using a randomized degradation approach. Initially, an isotropic or anisotropic Gaussian blur kernel was selected via a random strategy to blur the image, followed by the addition of either standard Gaussian noise, grayscale Gaussian noise, or Additive white Gaussian noise (AWGN). Subsequently, JPEG noise was incorporated to simulate noise introduced during network transmission.

Figure 3. Result graph generated after pixel-level degradation module based.

A random strategy was employed to expand the degradation space, determining the degree of degradation via random number generation. Diverging from conventional degradation techniques, this study extended the degradation of an entire image to individual pixels. A blank image of the same size as the original low-resolution image was first generated, into which pixels degraded at random were then populated, further enhancing the degradation space and achieving pixel-level high-order degradation. The corresponding formula can be expressed as follows:

$$P_{(i,j)} = \left(A_{1(i,j)}, A_{2(i,j)}, A_{3(i,j)}, \ldots, A_{8(i,j)}\right). \tag{15}$$

In Equation (15), $P_{(i,j)}$ denotes the pixel located at the i-th row and j-th column in P, composed of corresponding pixels from randomly chosen images A_1, A_2, ..., A_8. From both a performance and computational speed perspective, utilizing the eight degraded images to populate pixels was deemed an optimal choice.

4. Results

4.1. Datasets

4.1.1. TextZoom Dataset

The TextZoom dataset stands out as the inaugural dataset focusing on real-world text super-resolution, encompassing a series of paired scene text data. Traditional super-resolution methods, which typically employ simple bicubic interpolation or blur kernel for generating low-resolution images, fall short when applied to scene text, given its arbitrary shapes, varying backgrounds, and distinct lighting conditions. To address this more challenging issue, the TextZoom dataset incorporates LR-HR image pairs captured with digital cameras, sourced from RealSR [18] and SR-RAW [19]. It comprises 17,367 LR-HR image pairs for training. The test set is divided into three categories. The categorization is based on the insight that "the smaller the focal length, the blurrier the image is at the same height". In consideration of the recognition accuracy of the low-resolution images, the test set comprises 1619 image pairs cropped from RealSR, classified as belonging to the 'easy' difficulty category. These images are characterized by focal lengths greater than 100 mm, and notably, the recognition rate of low-resolution images from RealSR, under the ASTER recognition model, surpasses that of SR-RAW. A 'medium' difficulty category includes 1411 pairs of low-resolution images with focal lengths exceeding 50 mm. Finally, 1343 pairs of low-resolution images featuring focal lengths less than 50 mm are categorized as 'hard' difficulty. During experiments, LR image resolutions were adjusted to 16 × 64 pixels, while HR images were adjusted to 32 × 128 pixels. Figures 4–6, respectively, present samples from the TextZoom dataset spanning the simple, medium, and difficult subsets. As evident, while the simple difficulty LR images can be discerned with ease, those of high difficulty are profoundly blurred.

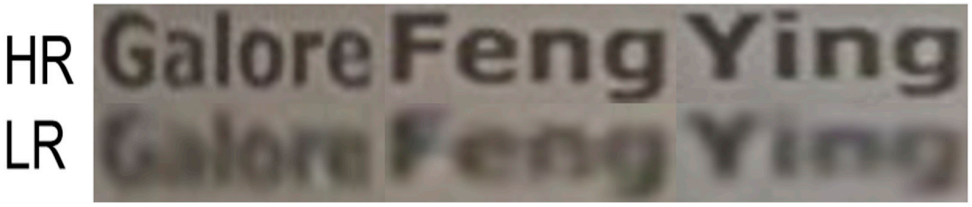


Figure 4. Easy difficulty images in the TextZoom dataset.

Figure 5. Medium difficulty images in the TextZoom dataset.

Figure 6. Hard difficulty images in the TextZoom dataset.

4.1.2. MJSynth Dataset

Introduced by Jaderberg et al. [20], the MJSynth dataset encompasses 9 million images and 90,000 English words. The dataset's creation involved a meticulous and deliberate process, ensuring its suitability for a broad array of applications. Font selection within the dataset is marked by its dynamic nature, with each image being meticulously generated through random font selection from a pool of over 1400 fonts sourced from Google Fonts. This deliberate randomness in font selection significantly contributes to the dataset's diversity and realism, guaranteeing that the text images portray an extensive array of font styles.

Furthermore, the MJSynth dataset is distinct in its blending of data from various sources. The dataset enriches its corpus by combining each image layer with random crops extracted from images featured in the training dataset of SVT. This blending process introduces a substantial degree of diversity into the dataset. The blending operations incorporate a range of alpha blend modes, including 'normal', 'add', 'multiply', 'max', and others, further amplifying the dataset's complexity and richness.

Given its vast data volume and the relevance of its text content to the experiments, this study employs this dataset for model pre-training, aiming to enhance the model's performance ceiling. Figure 7 exhibits a selection of images from the MJSynth dataset.

Figure 7. MJSynth dataset.

4.2. Implementation Details

Throughout the research, PyTorch served as the primary framework for method implementation. High-resolution images were resized to dimensions of 128 × 32 pixels, while the degraded low-resolution images were adjusted to 64 × 16 pixels. All experiments were conducted on a NVIDIA GeForce RTX 3090 GPU equipped with 24 GB of memory. The Adam optimizer was employed for model training, with a batch size set to 16.

Model training utilized the Adam optimizer with a batch size of 16. The training procedure was divided into two distinct phases. In the initial phase, the training dataset comprise the extensive MJSynth dataset, which consists of 9 million images. Low-resolution images were generated using the pixel-level degradation process introduced in this research. Subsequently, the test dataset encompassed the three challenging test sets from the TextZoom dataset. In the second phase of training, the dataset comprised 17,367 images from the TextZoom training set. Importantly, no modifications were made to the low-resolution images during this stage. The test dataset remained consistent with the first phase evaluation.

When training on the MJSynth dataset, the learning rate was established at 1×10^{-4}, while for fine-tuning using the TextZoom dataset, it was set at 7×10^{-4}. For evaluating recognition accuracy, OCR models such as ASTER, MORAN, and CRNN were employed, assessed using the official Pytorch code released on GitHub. To ensure fairness, the study adhered to prior practices in text image super-resolution research, converting all uppercase letters to lowercase. Experimental outcomes were gauged using OCR recognition rates to evaluate the model's performance.

In the experiments of this paper, four Residual Hybrid Attention Groups (RHAGs) were utilized, each equipped with six Hybrid Attention Blocks (HABs). Furthermore, the local window size was set to 7, enabling the model to focus on nearby pixel regions for super-resolution. Additionally, a 4:1 ratio was maintained for the MLP hidden dimension to the embedding dimension, determining the model's non-linear transformation capacity.

4.3. Experiment Result

In this section, the study delves into a comprehensive assessment of the text image super-resolution model based on pixel-level degradation processes on the TextZoom dataset. A meticulous comparison with prevailing super-resolution models is presented, encompassing EDSR [21], RDN [22], SRCNN, SRResNet [23], ESRGAN [24], TSRN, TSRGAN, and TBSRN. The results illustrate that the model proposed in this study outperforms others across all recognition rate metrics of various recognizers. It is pivotal to emphasize that the comparison was predominantly centered on recent models like TSRN, TSRGAN, and TBSRN, which are particularly tailored for text image super-resolution. Tables 1 and 2 contrast the outcomes of the methodology with other techniques based on ASTER and MORAN recognition models. It is evident that, in the ASTER recognition model, the methodology's recognition rates across simple, medium, and difficult levels reached 78.7%, 63.3%, and 45.5%, respectively. Compared to the current most proficient TBSRN technique, the proposed model enhances the average accuracy on ASTER by 2.4% and on MORAN by 2.3%. In terms of recognition rates of images with high difficulty, both ASTER and MORAN exhibited the most substantial growth at 3.9% and 4.2%, respectively, underscoring the significant advancements of the approach in super-resolving particularly blurred images.

Table 1. Comparison of the results of this paper's method with other methods on ASTER recognition.

Backbone	Easy/%	Medium/%	Hard/%
BICUBIC	64.7	42.4	31.2
SRCNN	69.4	43.4	32.2
SRResNet	69.6	47.6	34.3
EDSR	72.3	48.3	34.3
RDN	70.0	47.0	34.0
ESRGAN	68.4	49.5	35.6
TSRN	75.1	56.3	40.1
TSRGAN	75.7	57.3	40.9
TBSRN	75.7	59.9	41.6
Ours	**78.7**	**63.3**	**45.5**

Table 2. Comparison of the results of this paper's method with other methods on MORAN recognition.

Backbone	Easy/%	Medium/%	Hard/%
BICUBIC	60.6	37.9	30.8
SRCNN	63.2	39.0	30.2
SRResNet	60.7	42.9	32.6
EDSR	63.6	45.4	32.2
RDN	61.7	42.0	31.6
ESRGAN	63.4	43.2	34.3
TSRN	70.1	53.3	37.9
TSRGAN	72.0	54.6	39.3
TBSRN	74.1	57.0	40.8
Ours	**75.7**	**61.5**	**45.0**

Table 3 showcases a comparative analysis of the outcomes achieved by the proposed approach and other established methods when assessed with the CRNN recognition model. It is evident that the methodology achieves recognition accuracies of 62.7%, 55.0%, and 41.1% across the three defined levels of difficulty. When juxtaposed with the current state-of-the-art TBSRN method, the proposed approach exhibits an increment in recognition

rate by 3.1%, 7.9%, and 5.8% across these difficulties, respectively. Notably, when drawing a comparison between contemporaneous models, the most pronounced improvement between our model and the TBSRN is evident in the medium difficulty level, underscoring our technique's superior visual quality in text image super-resolution.

Table 3. Comparison of the results of this paper's method with other methods on CRNN recognition.

Backbone	Easy/%	Medium/%	Hard/%
BICUBIC	36.4	21.1	21.1
SRCNN	38.7	21.6	20.9
SRResNet	39.7	27.6	22.7
EDSR	42.7	29.3	24.1
RDN	41.6	24.4	23.5
ESRGAN	50.2	33.0	28.9
TSRN	52.5	38.2	31.4
TSRGAN	56.2	42.5	32.8
TBSRN	59.6	47.1	35.3
Ours	**62.7**	**55.0**	**41.1**

Tables 4 and 5 depict the comparison of the proposed method with other models in terms of PSNR and SSIM values. Notably, the approach yields lower scores in these metrics. However, this discrepancy is explainable. It is imperative to emphasize that the unconventional approach, which results in a reduction in PSNR and SSIM scores, stems from the specific nuances of super-resolution in text image restoration and the subsequent improvements in recognition accuracy.

Table 4. Comparison of this paper's method with other models in terms of PSNR values.

Backbone	PSNR		
	Easy	Medium	Hard
EDSR	24.26	18.63	19.14
RDN	22.27	18.95	19.70
LapSRN	24.58	18.85	19.77
ESRGAN	24.01	**19.62**	**20.30**
TSRN	**25.07**	18.86	19.74
TSRGAN	24.22	19.17	19.99
TBSRN	23.82	19.17	19.68
Ours	20.64	18.88	19.20

Table 5. Comparison of this paper's method with other models in terms of SSIM values.

Backbone	SSIM		
	Easy	Medium	Hard
EDSR	0.8633	0.6440	0.7108
RDN	0.8249	0.6427	0.7113
LapSRN	0.8556	0.6480	0.7087
ESRGAN	0.8489	0.6569	0.7290
TSRN	**0.8897**	0.6676	0.7302
TSRGAN	0.8791	**0.6770**	0.7420
TBSRN	0.8660	0.6533	**0.7490**
Ours	0.8029	0.6292	0.6655

Beyond Pixel-Level Metrics: While PSNR and SSIM are valuable in various image processing tasks, they primarily operate at the pixel level. In tasks where fine details and specific content, such as text, are of paramount importance, these metrics might not comprehensively reflect the true super-resolution results. Text image restoration demands

an exceptional level of detail and legibility, which goes beyond the scope of pixel-level assessments. The impact of super-resolution on character recognition, text clarity, and OCR accuracy is more effectively captured by a metric that assesses these higher-level qualities.

Evaluating Super-Resolution Results: Although PSNR and SSIM are widely recognized metrics in the super-resolution community, it is essential to acknowledge that they are not the ultimate authority on image quality. In practice, super-resolution results are often subjectively evaluated. Visual assessment by human observers remains a valuable approach to rank the visual reproduction effects of super-resolution result maps. This underscores the recognition that super-resolution quality is not solely determined by mathematical metrics but also by the perceptual experience.

Emphasizing Recognition Rate: In the context of scene text image super-resolution, where the legibility of characters and OCR performance are critical, the recognition rate stands out as a more reflective and practical performance metric. This is particularly pertinent when using the same pre-training OCR model. The recognition rate intuitively reflects the impact of character enhancement, making it an essential measure for assessing the real-world benefits of super-resolution in text image clarity and legibility. It is also aligned with the broader goal of super-resolution in facilitating downstream applications, where text recognition is often the ultimate objective.

In essence, the approach prioritizes the enhancement of text image legibility and character recognition, aligning with the practical applications of super-resolution in the domain of text image restoration. While PSNR and SSIM results may appear lower, the emphasis on recognition accuracy in this paper highlights the tangible and real-world applicability of the super-resolution technique. This reflects the commitment to optimizing text clarity and OCR performance, which is paramount in many text processing and analysis tasks.

To provide a more lucid comparative visualization of the super-resolution results across different models, Figure 8 delineates the enhanced super-resolution images from various models. For the purpose of offering improved clarity regarding finer details, Figure 9 displays an enlarged section of the model. Upon combining Figure 8 with Figure 9, it becomes evident that for the label "VARIETY", TBSRN incorrectly restores the letter "i" as "v". In the context of the "SWEEPING" label, TBSRN fails to recover the letter "E", interpreting it as "C" instead. In stark contrast, the proposed super-resolution model impeccably reconstructs these characters. Observing the labels "STORY" and "CONSTRUCTION", while other models manage a rudimentary contour restoration, their results are plagued with artifacts, elongated trailing distortions, and suboptimal reconstructions. This approach, on the other hand, delivers artifact-free and precise reconstructions, closely mirroring the original content. Thus, when juxtaposed with other text-focused super-resolution methodologies, the model not only evinces fewer artifacts and distortions but also maintains fidelity to the original content. To sum up, the proposed pixel-level degradation-based text image super-resolution technique holds promising potential for practical real-world applications in text image super-resolution.

When assessing the complexity of super-resolution models for text images, the number of parameters emerges as a crucial metric. It signifies the model's capacity to capture intricate features and details while maintaining efficiency. Table 6 presents a comparison of the model utilized in this study with the number of parameters in other models. With 5.2 million parameters, the model effectively balances complexity and utility. Remarkably, it achieves the highest recognition rate among the compared models, highlighting its suitability for optical character recognition (OCR)-oriented tasks. This refined feature extraction capability positions the model as a compelling choice for enhancing the legibility of text images in practical applications, solidifying its prominence in the field.

Figure 8. Comparison of super-resolution images of each model after visualization.

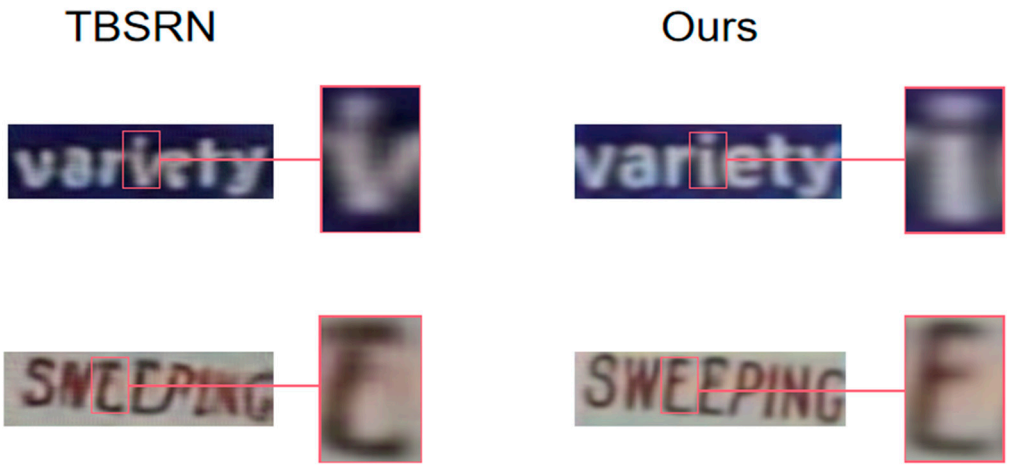


Figure 9. Comparison image with details enlarged.

Table 6. Comparison of this paper's method with other models in terms of the number of parameters.

Backbone	SRCNN	EDSR	ESRGAN	TSRN	TBSRN	Ours
Parameters	1.8 M	43 M	16 M	2.6 M	3.2 M	5.2 M

4.4. Limitation

Although the method in this paper can effectively improve the recognition rate of scene text images, there remain areas that warrant further attention, particularly in addressing highly blurred images. While recognition rates have exhibited substantial improvements, conventional metrics like PSNR and SSIM may still suggest room for optimization.

5. Conclusions

This paper has presented a novel approach to text image super-resolution, which amalgamates pixel-level degradation techniques, transfer learning, and conventional degradation methods. The methodology has yielded substantial improvements in recognition

rates, underscoring its practical applicability in enhancing text image clarity and legibility, particularly in OCR and text processing applications. Looking ahead, there is a compelling need to address the identified limitations, particularly in augmenting super-resolution efficacy for highly blurred images and optimizing performance across various contexts, including traditional quality metrics such as PSNR and SSIM. Moreover, extending the applicability of this approach to multi-language and multi-script scenarios holds significant promise.

Author Contributions: Conceptualization, X.Q., L.X. and S.Y.; methodology, L.X. and N.Y.; formal analysis, X.Q.; validation, N.Y. and R.L.; writing—original draft, X.Q. and L.X.; writing—review and editing S.Y. and R.L.; supervision, S.Y. and R.L.; funding acquisition, X.Q. and S.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Scientific Research Fund of Zhejiang Provincial Education Department (No. Y202352263) and the National Natural Science Foundation of China (No. 61972357).

Data Availability Statement: Data available on request from the authors. Our code is available at https://github.com/syyang2022/PDTS (accessed on 1 November 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Long, S.; He, X.; Yao, C. Scene text detection and recognition: The deep learning era. *Int. J. Comput. Vis.* **2021**, *129*, 161–184. [CrossRef]
2. Naiemi, F.; Ghods, V.; Khalesi, H. Scene text detection and recognition: A survey. *Multimed. Tools Appl.* **2022**, *81*, 20255–20290. [CrossRef]
3. Dong, C.; Zhu, X.; Deng, Y.; Loy, C.C.; Qiao, Y. Boosting optical character recognition: A super-resolution approach. *arXiv* **2015**, arXiv:1506.02211.
4. Wang, W.; Xie, E.; Liu, X.; Wang, W.; Liang, D.; Shen, C.; Bai, X. Scene text image super-resolution in the wild. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 650–666, Part X 16.
5. Chen, J.; Li, B.; Xue, X. Scene text telescope: Text-focused scene image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12026–12035.
6. Fang, C.; Zhu, Y.; Liao, L.; Ling, X. TSRGAN: Real-world text image super-resolution based on adversarial learning and triplet attention. *Neurocomputing* **2021**, *455*, 88–96. [CrossRef]
7. Ma, J.; Liang, Z.; Zhang, L. A text attention network for spatial deformation robust scene text image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5911–5920.
8. Wang, K.; Belongie, S. Word spotting in the wild. In Proceedings of the Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010; pp. 591–604, Part I 11.
9. Yuan, T.-L.; Zhu, Z.; Xu, K.; Li, C.-J.; Mu, T.-J.; Hu, S.-M. A large chinese text dataset in the wild. *J. Comput. Sci. Technol.* **2019**, *34*, 509–521. [CrossRef]
10. Risnumawan, A.; Shivakumara, P.; Chan, C.S.; Tan, C.L. A robust arbitrary text detection system for natural scene images. *Expert Syst. Appl.* **2014**, *41*, 8027–8048. [CrossRef]
11. Chen, H.; Gu, J.; Liu, Y.; Magid, S.A.; Dong, C.; Wang, Q.; Pfister, H.; Zhu, L. Masked Image Training for Generalizable Deep Image Denoising. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 1692–1703.
12. Lan, R.; Sun, L.; Liu, Z.; Lu, H.; Pang, C.; Luo, X. MADNet: A fast and lightweight network for single-image super resolution. *IEEE Trans. Cybern.* **2020**, *51*, 1443–1453. [CrossRef] [PubMed]
13. Ates, H.F.; Yildirim, S.; Gunturk, B.K. Deep learning-based blind image super-resolution with iterative kernel reconstruction and noise estimation. *Comput. Vis. Image Underst.* **2023**, *233*, 103718. [CrossRef]
14. Wang, X.; Xie, L.; Dong, C.; Shan, Y. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 1905–1914.
15. Niu, A.; Zhu, Y.; Zhang, C.; Sun, J.; Wang, P.; Kweon, I.S.; Zhang, Y. Ms2net: Multi-scale and multi-stage feature fusion for blurred image super-resolution. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 5137–5150. [CrossRef]
16. Xuhong, L.; Grandvalet, Y.; Davoine, F. Explicit inductive bias for transfer learning with convolutional networks. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 2825–2834.
17. Chen, X.; Wang, X.; Zhou, J.; Qiao, Y.; Dong, C. Activating more pixels in image super-resolution transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 22367–22377.

18. Cai, J.; Zeng, H.; Yong, H.; Cao, Z.; Zhang, L. Toward real-world single image super-resolution: A new benchmark and a new model. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3086–3095.
19. Zhang, X.; Chen, Q.; Ng, R.; Koltun, V. Zoom to learn, learn to zoom. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3762–3770.
20. Jaderberg, M.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv* **2014**, arXiv:1406.2227.
21. Lim, B.; Son, S.; Kim, H.; Nah, S.; Mu Lee, K. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 136–144.
22. Zhao, C.; Feng, S.; Zhao, B.N.; Ding, Z.; Wu, J.; Shen, F.; Shen, H.T. Scene text image super-resolution via parallelly contextual attention network. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, China, 20–24 October 2021; pp. 2908–2917.
23. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
24. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Change Loy, C. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Defect Detection Method of Phosphor in Glass Based on Improved YOLO5 Algorithm

Yong Qin, Zhenye Pan * and Chenhao Shao

College of Measurement and Control Technology and Communication Engineering; Harbin University of Science and Technology, Harbin 150086, China; qinyongmail@163.com (Y.Q.); 15524215017@163.com (C.S.)
* Correspondence: 18845631247@163.com

Abstract: Phosphor in Glass (PiG) is easy to be stirred unevenly during production and processing, and improper use of instruments and other factors lead to defective products. In this paper, we propose an improved YOLOv5 target detection algorithm. Firstly, the Coordinate Attention (CA) is introduced into the backbone network to enable the network to notice detect targets in a larger range. Secondly, the Bidirectional Feature Pyramid Network (BiFPN) is used to fuse different scale information in the neck part to obtain the output feature map with rich semantic information. At the same time, the weighted bidirectional feature fusion pyramid structure adjusts the contribution of different scale input feature maps to the output by introducing weights. This optimization enhances the feature fusion effect, reduces the loss of feature information in the convolution process, and improves detection accuracy. Then, the GIOU_Loss function is replaced with the EIOU_Loss function to speed up the convergence. Finally, the comparative experiment is carried out with the self-made PiG dataset. The experimental results show that the average accuracy mAP of this method is 12.35% higher than that of the original method (YOLOv5s), with a detection speed is 53.92 FPS, aligning with the actual needs of industrial detection.

Keywords: Phosphor in Glass; deep learning; industrial detection

1. Introduction

Phosphor in Glass (hereinafter referred to as PiG) is an all-inorganic composite fluorescent material sintered using phosphors and ceramic glass materials. It has high heat resistance, thermal conductivity, refractive index and light transmittance. It can effectively improve the excitation efficiency of phosphors while minimizing light attenuation [1]. During the process of preparing LED crystals from fluorescent ceramic glass materials [2], it is necessary to undergo high-temperature sintering, segmentation, dicing and other processes. These processes can lead to surface imperfections, including burns, holes, scratches, and defects like missing corners and deformation, all of which impact product yield [3].

In the selection process of fluorescent glass LED crystals, artificial visual methods and scanning electron microscopy are mostly used. However, the artificial visual method will cause people to have health problems such as visual decline and mental fatigue during prolonged periods of work [4]. At the same time, due to the limitations of human eye resolution, missed detection and false detection occur from time to time. Because scanning electron microscope [5] is expensive, it is often impractical for batch use on the production line. In recent years, the vigorous development of machine vision inspection and continuous updating and optimization of deep learning algorithms have provided a new way for industrial inspection [6].

Compared with traditional defect detection methods, defect detection based on deep learning trains the model through convolutional neural networks and uses the trained model to detect defects. This approach is more efficient compared to traditional methods, offering improved calculation speed and recognition accuracy [7]. According to the model

training method, it can be divided into two types: one-stage detection algorithms and two-stage detection algorithm [8]. Among them, the two-stage algorithm first generates a series of candidate boxes from the model, and then classifies them through a subsequent classification network. Common methods include R-CNN [9], Faster-RCNN [10], Cascade RCNN [11], etc. The network principle is mainly that a large number of Windows are generated in the first stage [12]. Windows uses a binary classification method to distinguish foreground and background. In the second stage, the region of interest (ROI) for target detection is used to deduct features from the feature map extracted by the convolutional neural network, followed by classification, which is a different process compared to that of the first stage. The second stage of classification work is multi-target classification, which is used to distinguish the categories of different targets and to predict the position of an object through regression [13]. While these methods greatly improve accuracy, they may suffer from lower detection rates. In contrast, single-stage algorithms such as SSD and YOLO [14] series do not need to generate regional candidate boxes, and feature extraction based on regional regression in the network results in high positioning accuracy and detection speed.

Therefore, in order to smoothly realize the rapid detection of PiG defects, this paper selects the most mature YOLOv5s algorithm in the YOLO series to complete the defect detection task, streamlining and improving some modules of the algorithm. Firstly, the coordinate attention mechanism CA is introduced into the Backbone network, and the coordinate information is embedded on the basis of channel attention to improve the detection ability of small defect targets. Secondly, different scale information is fused into the Bidirectional Feature Pyramid Network (BiFPN) at the neck layer to obtain an output feature map with rich semantic information. At the same time, BiFPN adjusts the contribution of different scale input feature maps to the output by introducing weights to optimize the feature fusion effect. Finally, the GIOU_Loss function is replaced by the EIOU loss function, which solves the problem of a certain amount of ambiguity in the GIOU_Loss function and the imbalance of difficult and easy samples.

2. Methods and Materials

2.1. Introduction of YOLOv5 Principle

YOLOv5 consists of three parts: Backbone, Neck and YOLO head (illustrated in Figure 1).

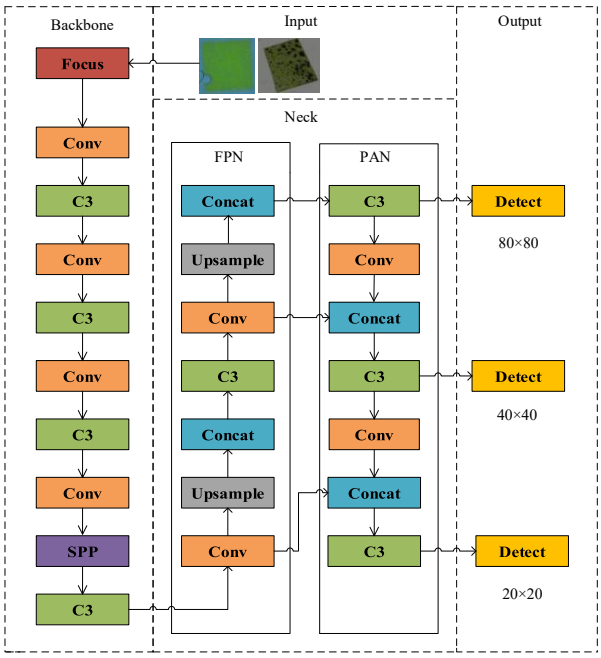


Figure 1. The structure of YOLOv5s network.

In the Backbone stage, various features are extracted, including the Focus, Conv, C3 and SPP modules. The Focus module reduces the amount of calculation and the number of network layers through slicing operations, thus improving the reasoning speed. Conv is a convolution module to extract features. The C3 module represents CSPNet, which splits the underlying feature map into two parts according to the channel information. One part passes through dense blocks and transition layers, while the other part is combined with the transmitted feature map. The SPP module is transmitted and fused downward through four different pooling layers.

In the Neck part of the network, features are fused and then transmitted to the output. The FPN structure constructs a high-level semantic feature map through a top-down approach, while the PAN adds a bottom-up route to complement and strengthen positioning information.

In the prediction layer, YOLOv5s uses the GIOU_Loss function as the loss function for bounding boxes. This effectively addresses issues where predicted boxes do not align precisely with actual boxes, leading to improved speed and accuracy of the prediction box regression. Finally, weighted non-maximum suppression is employed to enhance the recognition ability for multiple targets.

2.2. Adding Coordinate Attention

When the proportion of pixels in the image is small, the original YOLOv5 algorithm continuously extracts features through the convolutional layer, which is often prone to information loss, resulting in poor detection of small targets [15].

In order to solve the above problems, the Coordinate Attention (referred to as CA) is added to the network structure [16]. CA not only captures information between channels but also embeds direction-related positional information within channel attention. This effectively mitigates the issue where the channel attention mechanism tends to overlook positional information. Moreover, CA can encode information over a larger area, enabling the network to detect small targets across a broader range [17]. The structure of the CA module is shown in Figure 2, where "Residual" represents the residual structure, and "X Avg Pool" and "Y Avg Pool" represent average pooling in the X and Y directions, respectively.

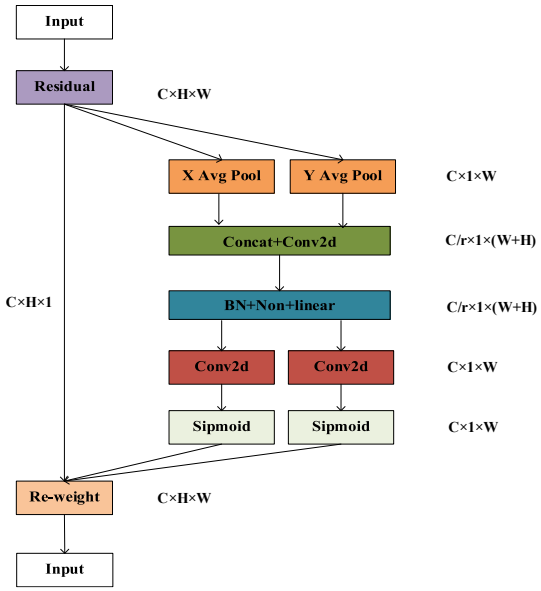


Figure 2. The structure of CA module.

The improved algorithm accurately captures and classifies defects such as holes and black spots, which are common in the production process of ceramic phosphors. As shown

in Figure 3, the CA module is introduced into the backbone network to enhance the model's ability to extract location information and feature expression.

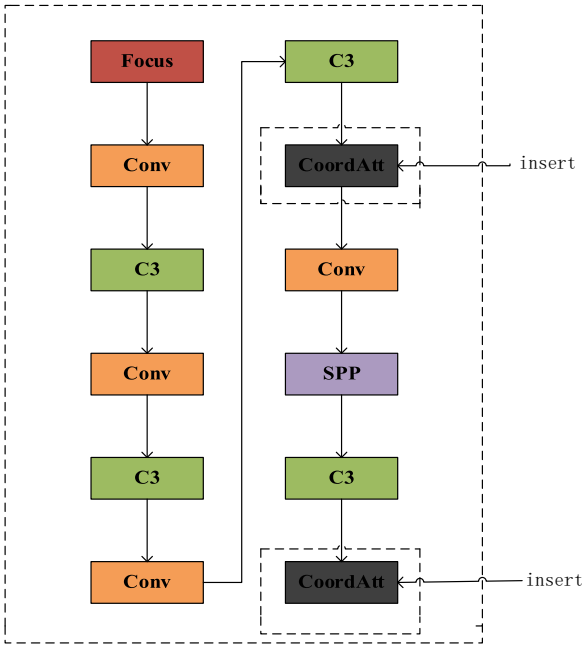


Figure 3. Improved backbone network.

2.3. Using A Bidirectional Feature Pyramid Network

Based on the standard feature pyramid, BiFPN achieves the purpose of simplifying the feature network by deleting the intermediate nodes with only one output edge [18]. At the same time, additional edges are added at the same level between the input and output nodes to fuse more features at a low cost. Finally, BiFPN processes each bidirectional path (top-down and bottom-up) as a single feature network layer and repeats the layer multiple times to achieve higher-level feature fusion. As shown in Figure 4, P3–P7 in the figure represent different levels of fused features. The blue arrow represents a top-down path, the red arrow represents a bottom-up path, and the purple arrow represents adding other edges to the input and output nodes of the same level.

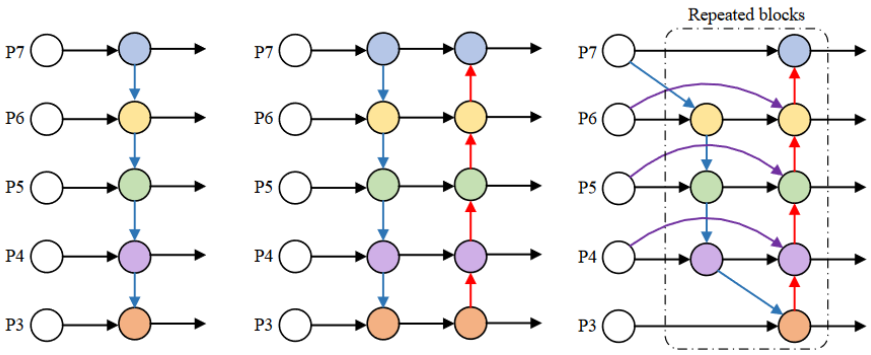


Figure 4. Two cross-scale connection optimizations implemented by BiFPN.

Since different input features have different resolutions, different weights should be attributed to the final output at each node performing feature fusion. Therefore, BiFPN introduces training weights to add additional weights to each input to adjust the contribution of different inputs to the output feature map. BiFPN uses fast normalized fusion to

select weights, which directly divides the sum of all values by weights and normalizes the normalized weights to the range [0, 1]. The calculation formula is shown in Formula (1):

$$o = \sum_i \frac{\frac{w_i}{n}}{\varepsilon + \sum_j w_j} I_i \tag{1}$$

where *Ii* represents the input characteristic, ε is a smaller value to avoid numerical instability; $\omega_i \geq 0$ is a small value that avoids numerical instability by applying ReLU after each ω_i to ensure that ε = 0.0001 [19].

As shown in Figure 5, the BiFPN network with repeated three layers is used as the improved method of the Neck part of the network in this paper. This adaptation enhances the network's capacity to fuse different input features and improve feature extraction across various scales.

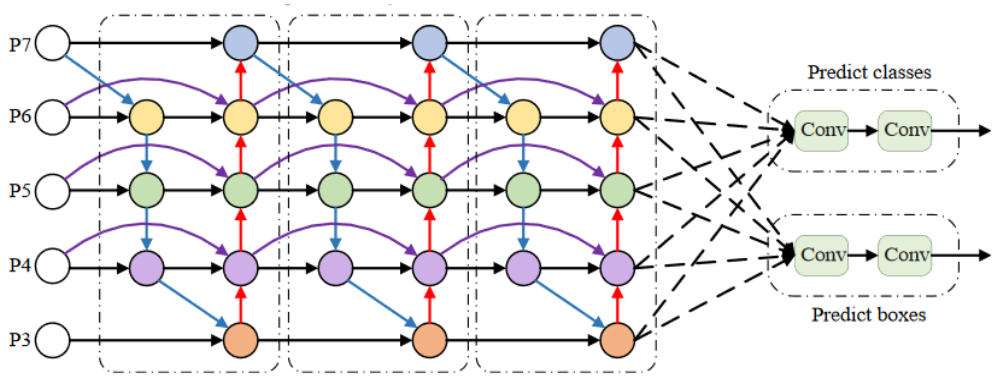


Figure 5. Repeated three layers BiFPN structure.

The improvements to the feature fusion structure in this paper are as follows: (1). Deepen the feature pyramid depth and increase detection head matching. The receptive field of the high-level feature map is larger, and the semantic information contained in it is richer. The fusion of low-level location feature information and high-level semantic information is more conducive to the identification and detection of small targets. (2). Delete unnecessary nodes and increase cross-layer connections. The nodes that do not participate in feature fusion contribute minimally to the feature network's ability to integrate different features. The nodes with only one input edge in the network are deleted, which simplifies the feature fusion network and effectively reduces the amount of model calculation. For the feature map of the same size, two additional cross-layer connections are added, which can fuse more feature information and improve the network detection accuracy with a slight increase in computational complexity. (3). Incorporate weighted feature fusion. Introduce learnable weights to learn the importance of different input features, similar to the attention mechanism. This enables the model to reduce the weight of secondary features and focus on the learning of more important and key features.

2.4. Improvement of Positioning Loss Function

The function of the loss function is to measure the distance between the neural network prediction information and the expected information. The closer the prediction information is to the expected information, the smaller the loss function value is. The loss function of YOLOv5 training includes Location Loss(e_d), Classification Loss(e_s) and Confidence Loss(e_k), which define the total loss of the network (l):

$$l = e_d + e_s + (a_1 + a_2 + a_3)e_k \tag{2}$$

where e_d is obtained by the GIOU loss, denoted by:

$$L_{GIOU} = 1 - IOU + \frac{S_2 - S_1}{S_2} \tag{3}$$

As shown in Figure 6, IOU is the intersection ratio of the prediction box and the real box, S_1 refers to the overlapping area of the two boxes, and S_2 represents the minimum rectangular area of the prediction box and the real box.

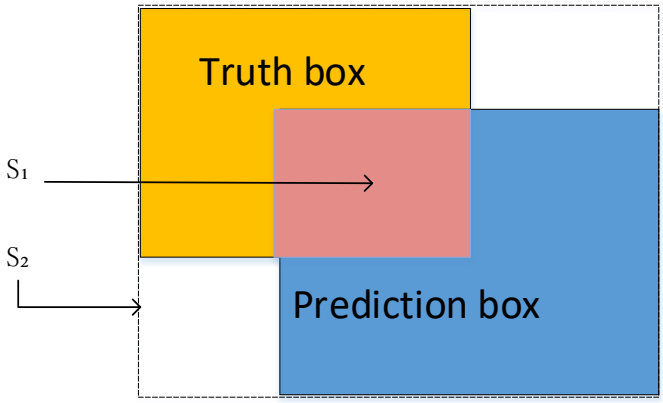


Figure 6. GIOU schematic diagram.

The original Yolov5 algorithm solves the problem that the IOU value is 0 when the two boxes have no intersection area by introducing the GIOU_Loss function [20]. However, when the aspect ratio of the two boxes is linear, there will be some ambiguity, so that the regression optimization cannot be continued. At the same time, this method does not consider the balance between difficult and easy samples. In order to solve these two problems, the EIOU loss function is introduced to split the influence factor of the aspect ratio from the CIOU function. It calculates the width and height of the prediction box and the real box are calculated, respectively. Based on the two losses of overlap area and center point distance of GIOU, EIOU changes the uniform penalty of aspect ratio to separate penalty, so that the difference between the width and height of the target box and the anchor box is minimized, and the convergence speed of the loss function and the regression accuracy of the prediction box are improved [21].

The EIOU_Loss principle is shown in Formula (4), where C_W and C_K represent the width and height of the minimum circumscribed rectangle of the prediction box and the real box.

$$EIOU_Loss = 1 - IOU + \frac{\rho^2(b^A, b^B)}{c^2} + \frac{\rho^2(w^A, w^B)}{c_w^2} + \frac{\rho^2(h^A, h^B)}{c_h^2} \tag{4}$$

EIOU_Loss, SIOU_Loss and AlphaIOU_Loss are all improved versions of IOU, which are used to calculate late the overlap between the detection box and the real box. Their differences are mainly reflected in the calculation methods and optimization objectives. In this paper, EIOU_Loss is selected according to the characteristics of the material itself. The optimization goal of EIOU_Loss is to maximize the overlap between the prediction box and the real box, and to consider the influence of angle differences. Subsequently, the superiority of its improvement is demonstrated through experimental data.

2.5. Causes of Defects

With the continuous optimization of material properties, the luminous efficiency of PiG-based white LED has been able to meet the requirements of general lighting boasting excellent heat and moisture resistance. This solid foundation enables PiG material to replace the organic resin-based fluorescent conversion layer in traditional white LEDs.

However, the application of PiG materials in the field of white LEDs is still constrained by the following three key scientific issues.

2.5.1. Poor Luminescence Performance

The luminescent properties of PiG materials are derived from the incorporation of commercial LED phosphors. However, the structural integrity of these phosphors is easily destroyed by the heat treatment and the formation of glass melt during the preparation process. Consequently, PiG materials cannot completely retain the initial luminescent properties of the phosphors. In addition, when a variety of phosphors are randomly dispersed in the same glass matrix, reabsorption occurs between the phosphors due to spectral overlap. This leads to a decrease in the luminous efficiency and color purity of PiG.

2.5.2. Low Transparency

The transparency of PiG materials is related to the scattering of light at the phosphor/glass interface. The scattering of light through PiG material, caused by scattering sources such as commercial LED phosphor grains and residual holes, conforms to Mie scattering. Based on the Van de Hulst approximation theory, the closer the refractive index of the glass matrix is to that of the scattering source, the lower the light scattering efficiency becomes. In fact, the poor transparency of the material caused by refractive index mismatch and holes significantly reduces the luminous efficiency of PiG-based white LED devices.

2.5.3. Mechanical Strength and Mass Production

The thickness of the fluorescent layer is usually controlled at a few hundred microns when high-efficiency white light emission is achieved in LED devices. However, how to ensure the mechanical strength of PiG materials and mass production in practical applications is a problem that needs to be continuously explored. Coating the fluorescent layer on the surface of the transparent glass substrate to convert it into a single functional layer can effectively solve the above problems. However, research in this area is rare.

There are six main defects caused by the above reasons. By categorizing the types of defect detection, the defects are classified into two types: surface damage and shape damage. Surface damage includes pollution, char, hole, bubble, and other types, while shape damage includes missing angles and deformation, among others. The annotation software LabelImg 1.4.3 is used to annotate the data, classifying defects into two categories: surface damage and shape damage. Figure 7a–d depict common surface damage, while Figure 7e,f depict common shape damage. The following are all defects of the original sample.

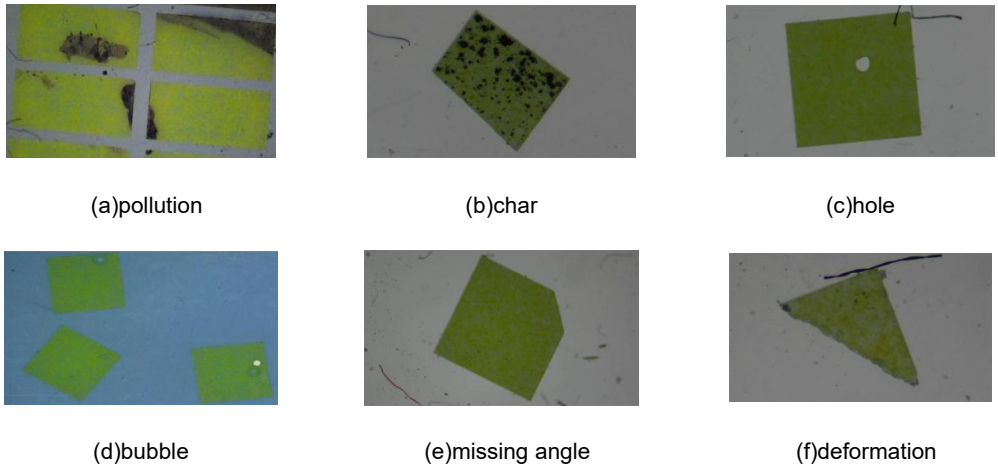


Figure 7. Common defect types.

2.6. Dataset Acquisition

In this paper, the data collection method for the fluorescent ceramic glass material dataset is collected by enterprise supplies. A high-definition CCD microscope, GP-530H of Kunshan High-quality Precision Instrument Co., Ltd, was produced in Kunshan City, Jiangsu Province, China. It is used for shooting. The camera magnification offers a magnification of 50 times, an object distance is 110 mm, and a shooting image resolution of 1920 × 1080. A vertical illumination scheme is adopted. The annular light source is placed vertically relative to the horizontal plane. The color temperature of light is 7000 k, which is white light illumination. The advantages of this scheme are uniform illumination, large illumination area and vertical light.

A total of 1000 defect samples, each sized 1920 × 1080, were obtained using electron microscopy. In order to meet the training requirements of deep learning, it is necessary to enhance the data. The above fault samples are flipped, cropped, translated and so on. A total of 3000 pictures are obtained and divided into training set and test set according to the ratio of 4:1 to establish PiG dataset. The dataset of surface defects is 1800, and the dataset of shape defects is 1200.

3. Results

3.1. Experimental Environment Settings

This experiment uses the deep learning platform of the cloud server, and the Linux operating system. The GPU graphics card model is NVIDIA Container Toolkit with 64 GB of memory. The deep learning framework employed is Pytorch 1.11.0, and the coding environment runs on Ubuntu 18.04 using Python 3.9. The CUDA version used is 11.3.

The experimental setting is basically based on the official recommended parameter setting of YOLOv5s, using adaptive anchors and mosaic data enhancement. The input image size is set to 640 × 640, the initialization learning rate is 0.01, the weight attenuation is set to 0.05, and the maximum number of iterations is set to 32, maximizing the memory usage of GPU devices. The maximum number of iterations for training is 200, and the validation dataset is evaluated after each epoch.

3.2. Experimental Evaluation Index

The model size, Prediction (P), Recall (R), mean Average Precision (mAP), and the number of frames per second (FPS) are used to evaluate the model. The model size can reflect the lightweight degree of the model. The accuracy represents the number of objects detected by the object detection system, while the recall rate signifies the probability of correctly detecting real objects. The mAP measures the model's recognition accuracy, and FPS represents the number of frames that can be detected per second. A higher FPS value indicates faster detection speed. The calculation process for P, R, mAP is shown in Formulas (5)–(7):

$$P = \frac{TP}{TP + FP} \times 100\% \, , R = \frac{TP}{TP + FN} \times 100\% \tag{5}$$

$$AP = \int_0^1 P(R) dR \tag{6}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{7}$$

In these formulas, TP represents the number of positive samples determined, FP represents the number of samples that should be negative but judged to be positive, FN represents the number of samples that should be positive but judged to be negative, AP represents the average accuracy of each category, and N is the number of classifications.

3.3. Experimental Result Analysis

In order to verify the feature extraction ability of the CA module, the current mainstream attention mechanism is added to the network, and the other parts remain unchanged. A comparative experiment is performed using the self-made dataset. The results are shown in Table 1.

Table 1. Comparing the results.

Algorithm	Model Size (M)	P (%)	R (%)	mAP@0.5 (%)	FPS
YOLOV5s	14.4	91.10	67.42	77.90	55.45
YOLOv5s-CEAM	14.5	92.85	69.34	79.34	44.56
YOLOv5s-SE	14.6	91.52	68.56	78.54	48.18
YOLOv5s-CA	14.6	92.61	72.22	81.61	49.42

Through analysis, it can be seen that when the detection speed is almost the same, the accuracy rate increases the most after adding the CA module, making it the most suitable for this dataset.

In order to test the effectiveness of the improved algorithm, ablation experiments were performed on the basis of the original algorithm. The same hyperparameters and training strategies were used in each group of experiments. The experimental results are shown in Table 2.

Table 2. Ablation experiment.

CA	BiFPN	EIOU	AP (%) Surface	AP (%) Shape	P (%)	R (%)	mAP@0.5 (%)	FPS	CA
1	—	—	—	67.85	88.15	91.10	67.42	77.90	55.45
2	—	√	—	69.53	89.07	91.82	73.46	79.30	55.46
3	√	—	—	73.61	89.62	92.61	72.22	81.61	49.42
4	—	—	√	73.34	89.36	91.34	74.58	81.35	57.42
5	√	√	—	82.52	90.66	92.52	78.68	86.59	50.97
6	—	√	√	82.26	91.26	91.26	77.27	84.76	58.65
7	√	—	√	84.80	90.84	91.80	83.45	87.82	53.41
8	√	√	√	88.94	91.56	93.45	84.56	90.25	53.82

Experiment 1 is the original algorithm. Experiment 2 only optimizes BiFPN. The results show that the accuracy and speed are slightly improved. In experiment 3, only the CA module was added on the basis of the first experiment resulting in a significant improvement in detection accuracy. However, this improvement came at the cost of reduced detection speed. Experiment 4 optimizes the loss function only on the basis of experiment 1, solving the issue of ambiguity in the original GIOU_Loss function, resulting in increased speed. Experiments 5, 6 and 7 adopt two kinds of optimization, demonstrating greater improvements in detection accuracy. Finally, all parts are improved in Experiment 8. Compared with the original algorithm, mAP is increased by 13.25%, while FPS is only decreased by 1.63%, which meets the requirements of industrial detection speed and greatly improves the accuracy.

In the case of setting the same training parameters, the improved YOLOv5s defect detection model and the original YOLOv5s model were trained for 200 rounds. The training results were saved once after each round of training, resulting in 200 saved training results. The obtained mAP change curve is shown in Figure 8 (the red curve represents the improved model, and the blue curve represents the original model). It can be seen from Figure 8 illustrates that the mAP value of the improved detection model is greatly improved compared with the original YOLOv5 model. The final mAP value is the best mAP value in 200 rounds of training.

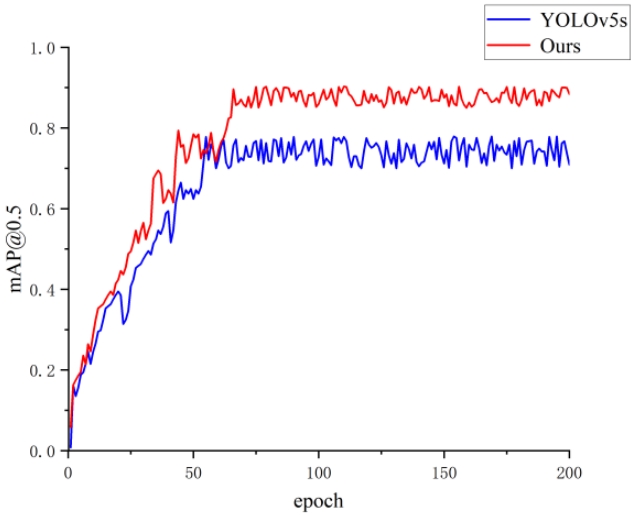
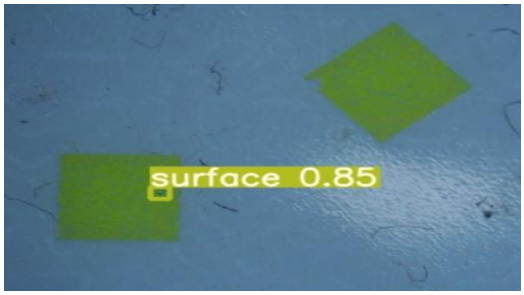
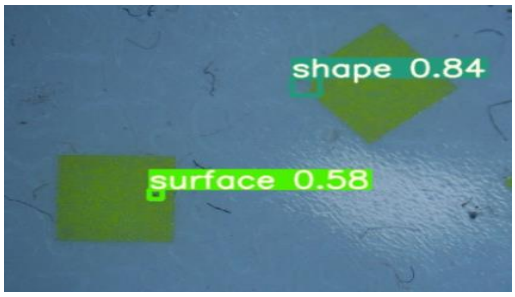


Figure 8. The mAP comparison diagram of the improved algorithm and the original algorithm.

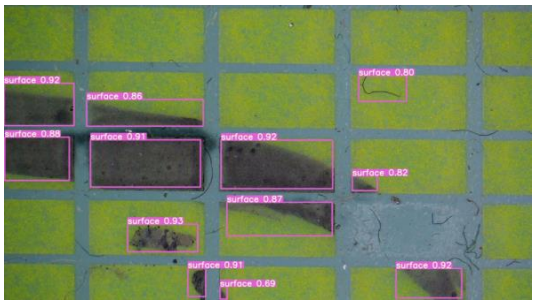
In order to verify the performance of the algorithm in practical applications, both the algorithm and the original YOLOv5s algorithm are used to detect the defects in the captured images. The detection results are shown in Figure 9.



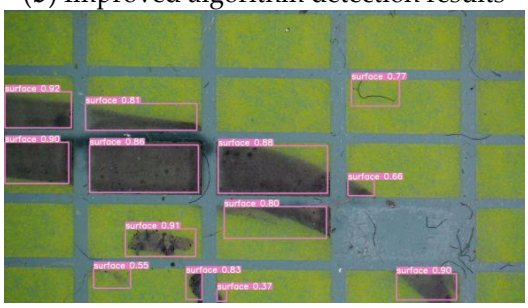
(**a**) Original YOLOv5s test results



(**b**) Improved algorithm detection results



(**c**) Original YOLOv5s test results



(**d**) Improved algorithm detection results

Figure 9. Comparison of test results.

It can be seen from the processing results in Figure 9 that in the first group of pictures, the PiG materials are disorderly arranged, and both algorithms detect the fluorescent bodies with surface damage. However, the original algorithm misses the detection of fluorescent bodies with shape defects, while the improved algorithm is more in line with the actual size of the defects. In the second group of images, for the pollution that is not obvious in the lower left corner, the original algorithm has missed detection. On the other hand, the improved algorithm enhances the detection of small target defects and less obvious target defects. Despite the differences in the two scene targets, the improved algorithm exhibits strong robustness.

In order to further verify the performance of the model in the same type of algorithm, the popular single-stage algorithm and the two-stage algorithm in the field of the deep

learning are used to detect defects in the same dataset under the same parameters. The detection results are compared with the results of the improved algorithm. The comparison algorithms include the classical Faster-RCNN, SSD algorithm, YOLOv3, YOLOv5 and lightweight algorithms such as YOLOv4-tiny and YOLOv5-mobileNet. The comparison results are shown in Table 3.

Table 3. Comparison between improved algorithm and deep learning mainstream algorithm.

Algorithm	Model Size (M)	P (%)	R (%)	mAP@0.5 (%)	FPS
Faster-RCNN	185.34	91.05	86.52	84.21	20.05
SSD	31.39	88.46	87.52	86.26	44.56
YOLOV3	61.56	85.05	66.58	75.55	38.50
YOLOv5s	14.40	91.10	67.42	77.90	55.45
YOLOv4-tiny	76.79	80.42	70.88	75.54	59.18
YOLOv5-mobileNet	77.44	75.79	65.24	70.83	64.20
YOLOv7s	94.8	92.14	85.65	91.32	42.65
Ours	18.15	93.45	84.56	90.25	53.82

It can be seen from Table 3 that the improved YOLOv5 algorithm is obviously superior to other algorithms in detection accuracy. Compared with the lightweight algorithms YOLOv4-tiny and YOLOv5-mobileNet with faster detection speed, mAP is 14.71% and 19.42% higher, respectively. For the defect detection of PiG, the improved YOLOv5 algorithm is superior to other models in comprehensive performance and meets the actual industrial production needs. compared to YOLOv7s, the improved algorithm achieves nearly the same mAP but with a slight increase in FPS, making it more suitable for the current dataset.

4. Discussion

In this chapter, the original algorithm has been improved in three aspects.

Firstly, by adding CA coordinate attention mechanism, the detection accuracy of small defects in fluorophore is enhanced. Secondly, the introduction of weights in the BiFPN feature fusion process optimizes the fusion of feature maps from different scales, reducing the loss of feature information in the convolution process, and improving the detection accuracy. Lastly, by introducing the EIOU_Loss function, the model is more focused on the influence of the prediction box with higher coincidence degree, and the accuracy of defect detection is improved again.

5. Conclusions

This paper proposes a defect method for YOLOv5s fluorescent ceramic glass material, focusing on enhancing loss functions and spatial attention mechanisms. By adding the spatial attention mechanism module CA, the detection layer obtains more semantic information and location information. By improving the feature fusion network BiFPN to optimize the feature fusion effect, the loss of feature information in the convolution process is reduced and the detection accuracy is improved. By replacing the loss function, the loss converges faster, and the model training time is reduced. The experimental results show that compared with the original YOLOv5s algorithm, the improved algorithm improves mAP by 12.35%, and the improvement effect is extremely obvious. Additionally, the algorithm exhibits enhanced performance in detecting small targets and demonstrates robustness in complex scene recognition.

Author Contributions: Conceptualization, Z.P.; methodology, Z.P.; software, Z.P.; validation, Z.P., Y.Q. and C.S.; formal analysis, Z.P.; investigation, Z.P.; resources, Z.P.; data curation, Z.P.; writing—original draft preparation, Z.P.; writing—review and editing, Y.Q.; visualization, Z.P.; supervision, Z.P.; project administration, Y.Q.; funding acquisition, C.S. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Data are contained within the article; The code is confidential.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ma, T.; Chen, H.; Yin, L. A novel Eu^{3+}-doped phosphor-in-glass for WLEDs and the effect of borophosphate matrix. *J. Rare Earths* **2023**, *41*, 190–199. [CrossRef]
2. Zhao, H.Z.; Liu, J.; Xu, H.W. Research on the quality reliability of LED lighting engineering based on life cycle management. *J. Qual. Stand.* **2021**, *5*, 42–45.
3. Zhu, S.C.; Luo, J.X.; Chen, H.B. The influence of LED position error on the imaging quality of Fourier laminated microscopy and its correction. *J. Foshan Univ. Sci. Technol.* **2022**, *40*, 9–16.
4. Utzet, M.; Ayala, G.A.; Benavides, F.G.; Basagaña, X. Extreme temperatures and sickness absence in the Mediterranean province of Barcelona: An occupational health issue. *J. Front. Public Health* **2023**, *11*, 1129027. [CrossRef] [PubMed]
5. Sathish, K.T.; Ezhil, P.P.; Makesh, M.; Poornima, M.; Jithendran, K.P. Artificial germination of *Enterocytozoon hepatopenaei* (EHP) spores induced by ions under the scanning electron microscope. *J. Invertebr. Pathol.* **2022**, *194*, 107820. [CrossRef]
6. Han, Y.; Li, G.Z.; Qin, Q.; Wang, S.P.; Li, Y.H. Research on Machine Vision Detection Method of Ship Sulfur Emission Based on Convolutional Neural Network. *J. Phys. Conf. Ser.* **2022**, *2171*, 012071. [CrossRef]
7. Habibullah, A.; Nanna, S.; Fikri, A. Surface Defect Detection and Classification Based on Statistical Filter and Decision Tree. *Int. J. Comput. Theory Eng.* **2013**, *5*, 774–779.
8. Pironti, P.; Ambrosanio, A.; Vismara, V. One-stage vs two-stage bilateral THA in Lombardy: A cost-effectiveness analysis. *Cost Eff. Resour. Alloc.* **2023**, *21*, 3. [CrossRef]
9. Jiang, X.J.; Du, X.L. Railway Catenary Insulator Recognition Based on Improved Faster R-CNN. *Autom. Control. Comput. Sci.* **2023**, *56*, 553–563. [CrossRef]
10. Youssouf, N. Traffic sign classification using CNN and detection using faster-RCNN and YOLOV4. *Heliyon* **2022**, *8*, e11792. [CrossRef]
11. Ajay, M.; Sanjeev, S. A Novel Color Coherence Vector Based Obstacle Detection Algorithm for Textured Environments. *Int. J. Comput. Theory Eng.* **2013**, *5*, 81–84.
12. Du, F.-J.; Jiao, S.-J. Improvement of Lightweight Convolutional Neural Network Model Based on YOLO Algorithm and Its Research in Pavement Defect Detection. *Sensors* **2022**, *22*, 3537. [CrossRef] [PubMed]
13. Li, F.; Zhang, X.Y.; Liang, S.C. Vascular interventional guidewire detection based on YOLO algorithm. *Beijing Biomed. Eng.* **2023**, *42*, 341–347.
14. Li, W.; Zhang, L.; Wu, C.; Cui, Z.; Niu, C. A new lightweight deep neural network for surface scratch detection. *Int. J. Adv. Manuf. Technol.* **2022**, *123*, 1999–2015. [CrossRef]
15. Mahaur, B.; Mishra, K.K. Small-object detection based on YOLOv5 in autonomous driving systems. *Pattern Recognit. Lett.* **2023**, *168*, 115–122. [CrossRef]
16. Wu, J.; Dong, J.; Nie, W.Y.; Ye, Z.W. A Lightweight YOLOv5 Optimization of Coordinate Attention. *Appl. Sci.* **2023**, *13*, 1746. [CrossRef]
17. Wang, Z.K.; Cao, Y.; Yu, H.F.; Sun, C.H.; Chen, X.J.; Jin, Z.G.; Kong, W.L. Scene Classification of Remote Sensing Images Using EfficientNetV2 with Coordinate Attention. *J. Phys.* **2022**, *2289*, 012026. [CrossRef]
18. Xie, X.; Li, H.P.; Hu, H.P. The Flocs Target Detection Algorithm Based on the Three Frame Difference and Enhanced Method of the Otsu. *Int. J. Comput. Theory Eng.* **2015**, *7*, 197–200.
19. Plonka, G.; Riebe, Y.; Kolomoitsev, Y. Spline representation and redundancies of one-dimensional ReLU neural network models. *Anal. Appl.* **2023**, *21*, 127–163. [CrossRef]
20. Ye, N.; Wang, R.G.; Li, N. A Novel Active Object Detection Network Based on Historical Scenes and Movements. *Int. J. Comput. Theory Eng.* **2021**, *13*, 79–83. [CrossRef]
21. Zeng, W.; Huang, J.J.; Wen, S.P.; Fu, Z.J. A masked-face detection algorithm based on M-EIOU loss and improved ConvNeXt. *Expert Syst. Appl.* **2023**, *225*, 120037. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Fully Automatic Approach for Smoke Tracking Based on Deep Image Quality Enhancement and Adaptive Level Set Model

Rimeh Daoudi [1,2], Aymen Mouelhi [1], Moez Bouchouicha [2,*], Eric Moreau [2] and Mounir Sayadi [1]

[1] Université de Tunis, ENSIT, Laboratory SIME, Tunis 1008, Tunisia; rimehdaoudi@gmail.com (R.D.); aymen_mouelhi@yahoo.fr (A.M.); mounir.sayadi@ensit.u-tunis.tn (M.S.)
[2] LIS-CNRS, Université de Toulon, Université Aix-Marseille, 83130 Toulon, France; eric.moreau@univ-tln.fr
* Correspondence: moez.bouchouicha@univ-tln.fr

Abstract: In recent decades, the need for advanced systems with good precision, low cost, and high-time response for wildfires and smoke detection and monitoring has become an absolute necessity. In this paper, we propose a novel, fast, and autonomous approach for denoising and tracking smoke in video sequences captured from a camera in motion. The proposed method is based mainly on two stages: the first one is a reconstruction and denoising path with a novel lightweight convolutional autoencoder architecture. The second stage is a specific scheme designated for smoke tracking, and it consists of the following: first, the foreground frames are extracted with the HSV color model and textural features of smoke; second, possible false detections of smoke regions are eliminated with image processing technique and last smoke contours detection is performed with an adaptive nonlinear level set. The obtained experimental results exposed in this paper show the potential of the proposed approach and prove its efficiency in smoke video denoising and tracking with a minimized number of false negative regions and good detection rates.

Keywords: convolutional autoencoder; smoke detection; image quality enhancement; deep learning models; level set models

1. Introduction

Wildfires are major natural disasters that can cause extensive damage to ecosystems and threaten human lives. It is an uncontrollable and destructive fire that rapidly spreads through vegetation, grasslands, or other flammable areas. Wildfires are typically triggered by a combination of factors, including the presence of abundant dry vegetation and favorable weather conditions like high temperatures, low humidity, and strong winds. The sources of ignition for wildfires are diverse and can range from natural causes like lightning strikes to human activities such as campfires, careless disposal of cigarettes, or even intentional acts of arson. Besides the destructive nature of wildfires, the smoke from wildfires can have severe human health risks and environmental consequences as it can contribute to air quality degradation, disrupt the balance of ecosystems, and even impact the behavior and survival of wildlife.

Early fire and smoke detection are crucial; some recent methods and tools focus on smoke and fire detectors [1] due to their low cost. However, they have some limitations, such as poor performance in open and wide areas and the detection time, which is dependent on the distance between the detector and the smoke source [2]. Recent monitoring systems such as remote sensing technologies, satellite imagery, and ground-based sensors rely on artificial intelligence-based algorithms in detecting and monitoring the outbreak of fires and their spread. In those systems, vision software runs at its core and automatically detects the presence of an event.

In the last years, like other fields, computer vision-based fire and smoke detection methods have attracted the attention of researchers [3–6]. Compared to conventional

methods [7–11], they have many advantages like the large area coverage, the real-time response, and the high accuracy.

Some of the existing approaches integrated various combinations of hand-crafted traditional techniques to detect smoke image features like shape, color, and texture [12–14]. At the same time, others exploit the moving nature of smoke to extract the motion characteristics [13,15,16] (motion value, direction, energy, convex hull), achieving good results considering some particular cases.

In recent years, deep learning-based algorithms, especially CNNs [17–21], have gained so much attention, and many researchers investigated those models and achieved good precision results. Hence, the use of those methods is still challenging since they rely on supervised learning and require huge databases for training, and their effectiveness depends on various factors, including the strong hardware systems for implementation and the quality and diversity of the dataset used for training. A database is one of the bases of a deep learning model, and the provided results largely depend on the comprehensiveness and the authenticity of training data. Industries and institutions involved in deep learning spend a lot of time and energy collecting datasets. The significant investment required to acquire these datasets adds value to deep learning training and underscores their importance. The availability of public smoke and fire databases with their ground truth labels is one of the biggest challenges related to deep learning models. In addition, smoke and wildfire images captured from fixed cameras, drones in motion, or other imaging devices are often affected by noise, artifacts, or other imperfections that can affect their interpretation and analysis. To address these challenges, Image processing techniques, such as noise reduction, image enhancement, and image restoration, are employed to improve the quality of acquired images, making them more suitable for interpretation.

In contrast to these previous methods and related challenges, a further solution is described in this paper. In this study, we propose a fully automatic hybrid method designated for noise removal and image quality improvement followed by smoke tracking in video sequences. The proposed benchmark consists of two stages. In the first stage, we construct an image denoising and quality enhancement path based on lightweight convolutional encoder-decoder new architecture. In the second stage, we propose a novel scheme for smoke tracking in videos provided by cameras in motion. The proposed scheme consists of tracking the evolution of smoke that appears in different shapes and from different angles in consecutive frames. First, a decision function to indicate the appearance of smoke in the first frames in a recorded video based on hand-crafted features like energy and color. The goal of those primary steps is to extract the foreground image that contains possibly smoke region candidates. Second, smoke motion tracking with a nonlinear adaptive level set framework. In this work, our main contributions are three folds: (i) we develop a novel fully unsupervised pipeline for smoke tracking in videos captured from a source camera in motion, (ii) we analyze the convolutional encoder-decoder structures, and we suggest a novel view of its architecture that is designated especially to reconstruct and denoise wildfire images, and (iii) we track the smoke regions evolution starting from the small appearance regions in the first video frames. Besides giving good detection results, the advantage of the proposed method is to be fully unsupervised, which avoids the need for labeled, big, and high-resolution databases. In addition, in theory, it is apt for deployment in real-world situations with low computational cost.

Extensive experiments and comparisons with the state-of-the-art methods confirm the good performance of the proposed benchmark. In addition, our method is able to detect the smoke at early stages with a small appearance from a moving camera, which, to our best knowledge, has not been discussed in related research.

The rest of the paper is organized as follows: in Section 2, we provide an overview of recent methods related to fire and smoke detection; Section 3 exposes the proposed approach; Section 4 discusses the obtained results and evaluation of our method; and in Section 5, we end up with conclusions and future works.

2. Related Works

Computer vision and deep neural networks have emerged as powerful strategies for studying the behavior of fire and smoke. Numerous methods have been introduced in the literature to leverage these technologies. Kaabi et al. [22] introduced a method for smoke detection. First, a motion-based feature extraction with the GMM algorithm was employed; then, a trained DBN classifier based on the feature of smoke was used to detect the smoke region in videos. Xu et al. [23] proposed an end-to-end method for smoke detection based on a deep saliency network; the framework was used to extract the smoke saliency map via a pixel level and object level salient CNN. Yifan et al. [24] proposed a real-time detector network for fire and smoke based on a light YOLOv4. The model is based mainly on three modules: the MobileNetv3, the BiFPN, and a new feature extraction technique with depthwise separable convolution and attention block. The method attends good performance with a minimum of trainable parameters. Cao et al. [25] proposed a bidirectional LSTM network for forest fire and smoke detection in videos, and it consists of bidirectional learning of the discriminative spatiotemporal features. Ali Khan et al. [26] exploit the advantages of transfer learning to train the VGG19 model to localize wildfires; then, they build a network composed of unmanned aerial vehicles communicated to an assistance center to simplify and increase the data transmission process. Frizzi et al. [27] proposed a new CNN architecture for fire and smoke segmentation and classification. The network is composed of coding and decoding paths and achieves better accuracy compared to similar methods with low false positives (clouds, haze) and in optimal segmentation time. Aymen et al. [28] integrate the non-linear adaptive level set method with an artificial neural network model to track the forest fire regions in wildfire videos; the method consists of estimating and localizing the possible fire contours by analyzing the chromatic and statistical features with the linear discriminant analysis combined with an artificial neural network. Then, a level-set algorithm was applied to refine the segmentation results. The method gives good results in terms of speed and accuracy. An interesting approach proposed by [29] concentrates and studies the architecture and the training of networks; it combines the self-attention mechanism with a multi-scale feature connection for real-time fire and smoke detection. The authors first fused the feature maps of the network into a radial connection; then, they applied a permutation attention mechanism to gather the relevant information, and designed a feature fusion block to increase the detection efficiency. The method gives good results compared to standard proposed methods.

Recent technologies based on advanced artificial neural network architecture achieve good results, but they still have limitations, especially when trying to implement them for daily use because of the need for huge databases for training and the strong/expensive computation resources. From this, many methods attempt to detect fire and smoke by building a pipeline integrating the image processing techniques and by exploiting the smoke textural features (Table 1). Ref. [30] built a statistical model combined with an optical flow algorithm for real-time fire and smoke detection. The method first extracts the smoke and fire-like regions with frame differential steps, a color model of fire and smoke, and a foreground accumulation technique. A motion feature discriminating model with the optical flow was applied to the first resulting image to extract the final fire and smoke regions. In [31], the fire presence decision in video frames is reached by analyzing the color variations and periodic behavior of the flame with the temporal and spatial wavelet transform algorithm. Ref. [32] proposed a fire detection system consisting of modeling the color information in the CIE L^*a^*b color space and detecting the motion of fire pixels with a background subtraction technique.

Table 1. Overview of main methods and findings in the smoke detection literature.

	Color Space	Features	Method	Accuracy (%)
Hashemzadeh [7]	RGB	Motion	CNN SVM	97.6
Pundir [33]	RGB	Motion Texture Color	Deep CNN	97.4
Yin [34]	RGB	Motion	CNN	97.0
Toreyin [35]	YUV	Motion Energy Disorder	Wavelet transform	-

3. The Proposed Smoke Tracking Method

The proposed approach presented in Figure 1 is based mainly on two stages: the first stage is a reconstruction path, which aims to enhance the image quality and remove noise, and the second stage consists of smoke detection and tracking using a new algorithm based on smoke image features extraction steps combined with the nonlinear adaptive level set method.

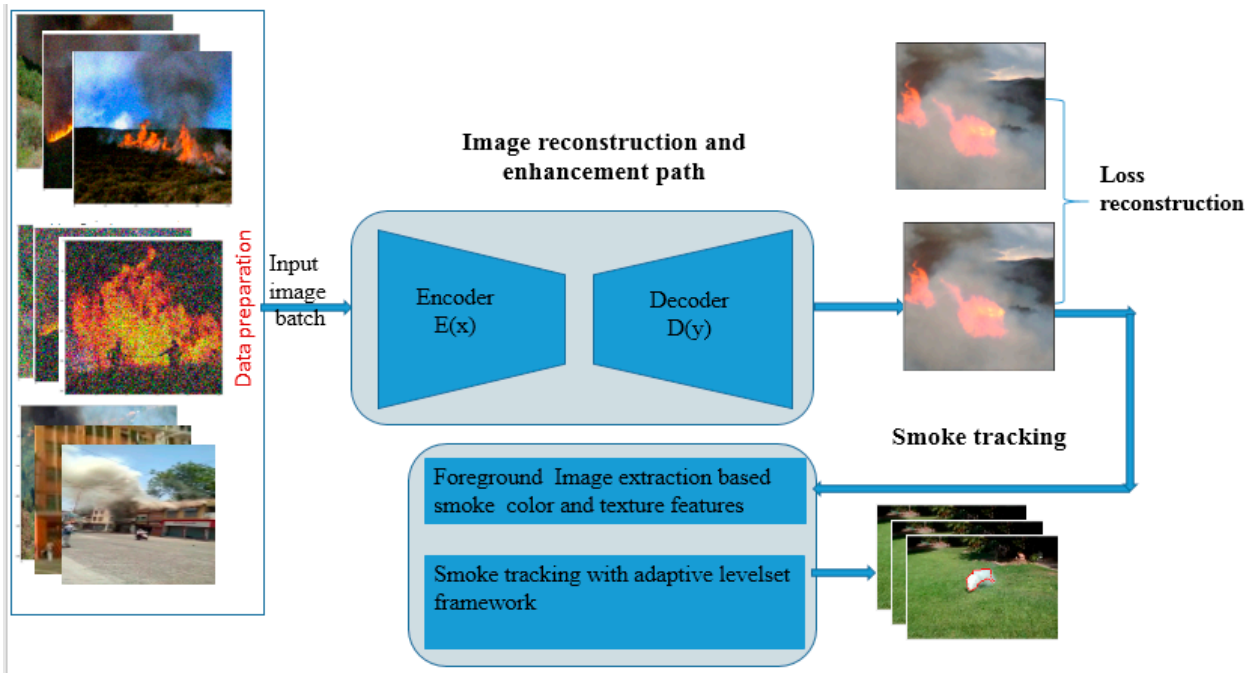


Figure 1. Flowchart of the proposed method.

3.1. Image Enhancement Path

3.1.1. Dataset Preparation

In this work, to train and validate the proposed convolutional autoencoder CAE architecture, we used a dataset collected by our laboratory research team. The dataset contains 450 RGB images, and it was divided into 80% used for training and 20% for validation of the network. Note that to optimize the network learning, the dataset used to train the network contains a diverse combination of clean and noisy images with varying noise levels, as shown in Figure 2. Subsequently, every inputted image was normalized and resized to a consistent size ($256 \times 256 \times 3$).

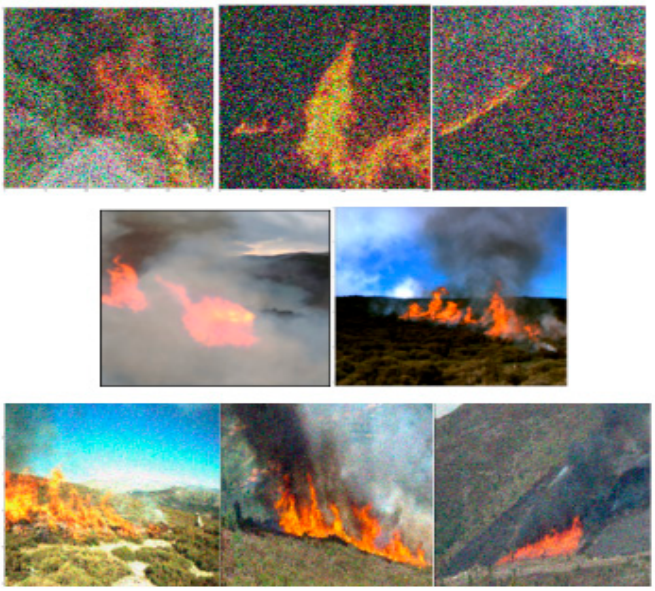


Figure 2. Frames from the training database. Row (1): noisy images with Gaussian noise: $\mu = 0$, $\sigma = 1$, $\rho = 0.7$. Row (2): clean original images. Row (3): noisy images with $\mu = 0$, $\sigma = 1$, $\rho = 0.1$.

3.1.2. The Proposed Lightweight CAE Network Architecture

Deep learning dimensionality reduction methods such as Variational auto-encoders and their variants and generative adversarial networks have gained so much attention. Some methods exploit the obtained latent space representation to help classification and detection models better learn the distribution of information to achieve high-precision results, while others tend to generate new data, for example, the data understanding approaches.

Many feature fusion and selection techniques employ neural architecture for better cross-scale feature network topology. Auto-encoders [36] have demonstrated their efficacy in mapping and modeling the data and proved their performance in data mapping. They have been successfully applied in various tasks, including data manipulation, image generation, and reconstruction.

Convolutional auto-encoders (CAE) [37] build upon the concept of auto-encoders by incorporating a consecutive stacked convolutional layers, CAE is designed to learn a specialized representation of image data, it involves mapping an input image data x to a latent space representation h via an Encoder $h = E(x)$ and then reconstruct it to obtain the output image x via a decoder D with $x = D(h)$.

To improve the learning process and help the model predict the missing values in a corrupted image $\widetilde{x}$ with noise, we employed a data mixing technique, which involved injecting a batch of data that was corrupted with varying noise levels into the model. It is important to note that this task is completely unsupervised, meaning that labeled data are not required to train and fine-tune the CAE model.

Let us represent the feature representation of $\widetilde{x}$ as h by denoting it as:

$$h = \sigma(w_\vartheta \widetilde{x} + b_\vartheta) \tag{1}$$

where $\widetilde{x}$ is the input image to be reconstructed.

The reconstructed image z is defined by:

$$z = \delta(W_k h + b_k) \tag{2}$$

where W_v, b_v, W_h, b_h, are the weights and biases of the CAE model.

Our designed CAE architecture operates by first flattening the input image data into a feature representation vector. It incorporates a sliding window and multiple hidden layers consisting of convolution, pooling, and batch normalization operations. This architecture

enables the network to learn distinctive contextual patterns and global and local features from various regions within the image.

Figure 3 illustrates the detailed architecture of the proposed CAE model. It consists of two paths: the encoding path begins with two convolution layers with a kernel size of 3×3 and 64 filters. This is followed by a normalization layer, a max-pooling operation, and ends with a 3×3 convolution layer followed by a max-pooling operation.

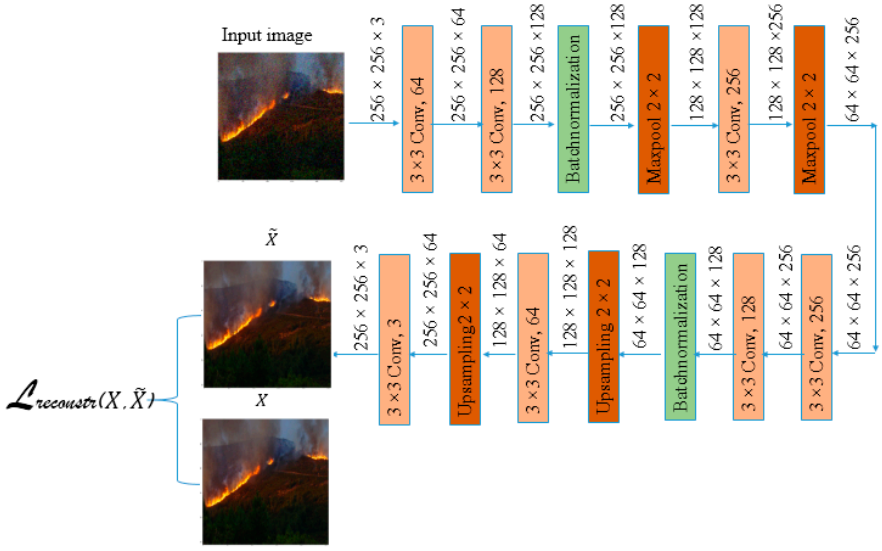


Figure 3. An overview of the proposed architecture of the CAE network.

The decoding path of our model consists of two convolution operations. The first operation utilizes 256 kernels, while the second operation employs 128 kernels. These blocks are followed by a batch-normalization and up-sampling operations. Next, the feature map is passed through a convolution block and another up-sampling operation. The decoding path concludes with a convolution layer.

The architecture we propose is designed to be lightweight and promotes local connectivity between neurons across different layers. This enables a hierarchical decomposition of fire and smoke images, representing low and high-level features at varying scales and levels. Additionally, it effectively blurs the input image data and maintains fine details like contours and corners.

Note that the model was trained and validated on images for computation and time simplicity reasons. After training the network and saving the best-achieved weights and biases, the video smoke sequences are decomposed into frames, and then those frames are fed one by one into the network for quality improvement and noise removal.

3.2. The Tracking Path

3.2.1. Color Modeling for Smoke Detection

Our aim in this study is to detect the smoke regions in video sequences acquired from a camera with motion. The video dataset used for testing the tracking algorithm is available at http://signal.ee.bilkent.edu.tr/VisiFire/Demo/SampleClips.html (accessed on 12 April 2023). The smoke in the chosen videos is characterized by its variable shape, appearance, and motion.

The smoke usually appears with grayish colors [38], either dark or light levels of gray. The appearance of smoke depends on the combustion process. In this work, we consider the light smoke since we want to find out the smoke at earlier stages. In terms of computation, the RGB color model has less complexity than other models, but in smoke color recognition, the extraction of smoke color-related information in RGB domain space is not suitable. In this paper, the HSV color domain is used, which is a more people-oriented

information model. Our experimental results show that to obtain the primary foreground image, every selected HSV smoke pixel should satisfy the following conditions:

$$\text{(i) } 0.26 \leq H \leq 0.53 \tag{3}$$

$$\text{(ii) } 0.01 \leq S \leq 0.13 \tag{4}$$

$$\text{(iii) } 0.7 \leq V \leq 1 \tag{5}$$

Foreground thresholded images may contain both smoke and disturbances, which are to be differentiated in the following phases.

The pixels that satisfy the above color decision rule are set as smoke region candidates/foreground images, and the rest are the background. Afterward, the foreground image underwent a refinement process that involved removing regions with small areas. Additionally, the smoke region candidates were smoothed using a morphological closing operation, and any holes within the regions were filled. These pre-processing steps significantly improved the quality of the obtained mask and reduced the number of false detections.

3.2.2. Smoke Feature Extraction

To enhance the threshold result of the color model, we need to ensure that the grayish detected region is a smoke region. Many methods exploit the motion feature by using a differential method, but since we are detecting smoke regions from a moving camera, the subtraction frames technique is not efficient. We studied the different image features between consecutive frames, such as correlation, energy, mean, entropy, homogeneity, and contrast, etc., and we found that in a video, the energy and the homogeneity of an image increase with the presence of smoke. So, to consider a region that appears during consecutive frames as a smoke region, the α and β terms must satisfy the conditions:

$$\alpha(I_K, I_{K-1}) > 0 \tag{6}$$

$$\beta(I_K, I_{K-1}) > 0 \tag{7}$$

where α is the difference value of the energy feature between the current frame I_K and the previous frame I_{K-1} two consecutive frames and β is the difference value of the homogeneity feature between the current frame I_K and the previous frame I_{K-1}.

3.2.3. Level Set Segmentation

To proceed with the smoke image quality enhancement results, we propose in this paper to adapt a nonlinear adaptive level set for smoke tracking refinement in video sequences. The level set-based method is an extension of the active contour without edges model proposed by Chan–Vese [39]. The simplified Chan–Vese model gives an approximation to an image in a piecewise constant piece; then, the same authors designed a penalty term by measuring the distance between the approximation and the original image, which helps the model be robust with noise. The model has been extended into many methods, e.g., the vector-valued images for color images [40], the multiphase level set [41], Tsai et al. [42] gave a piecewise smooth approximation for image segmentation and denoising derived from [39]. Ref. [43] proposed a maximum posterior (MAP) function to model the external energy. These methods are robust against the noise since they exploit the regional image features. However, they are inefficient in segmenting images with inhomogeneous regions due to the ignoring of local image features.

The nonlinear adaptive level set function presented by [44] evolves an initial curve by minimizing an energy function. The evolution of the curve takes into consideration local and global region features.

Let Φ be the level set function of an inputted gray level image defined on image domain Ω with $C = \{(x, y) \mid \Phi(x, y) = 0\}$ the zero-level set at the boundary region(C), $\Phi > 0$ inside the contour C and $\Phi < 0$ outside the contour region C.

The formulation of the energy function to be minimized for a gray-level image is as follows:

$$E_g(\Phi) = \mu \int_{\Omega} \frac{1}{2}(|\nabla\Phi| - 1)^2 \, dxdy + \lambda \int_{\Omega} \delta_\varepsilon(\Phi) g |\nabla\Phi| \, dxdy + \nu \int_{\Omega} g \, H_\varepsilon(-\Phi) dxdy \quad (8)$$

where the three terms of the right-hand side of (8) are, respectively, the penalization term, which controls the property of the signed distance during the evolution process of C, the weighted length of the curve, and the weighted area inside Φ. μ, λ and ν are the parameters controlling the effects of the three terms of the energy function.

Large μ and small λ help the algorithm to filter high-frequency noise and detect objects with finer details. $H(.)$ is a regularized Heaviside function, and its derivative is the regularized Dirac function $\delta(.)$. g is an edge detector function.

The evolution direction function takes into account the global and local statistical features and is computed with a posterior probability obtained by the Bayesian rule. The conditional probability is obtained with a Gaussian function with color channels of an RGB image. The direction evolution function was determined as follows:

$$Adf(x,y) = \frac{1}{3} \sum_{j=1}^{3} \frac{(I_j(x,y) - \mu_{1j})^2}{2\sigma_{1j}^2} - \frac{1}{3} \sum_{j=1}^{3} \frac{(I_j(x,y) - \mu_{2j})^2}{2\sigma_{2j}^2} - \ln\left(\frac{1}{3}\sum_{j=1}^{3} \frac{\sigma_{1j}}{\sigma_{2j}}\right) \quad (9)$$

where $j = 1, 2, 3$ are the RGB channels, respectively, of a processed frame.

To increase the evolution speed of the level set function in the homogeneous regions and decrease it when crossing the real boundaries, a velocity parameter was adapted by [44]:

$$v(x,y) = 2\left[\frac{1}{1 + \exp(-\zeta \, Adf(x,y))} - 0.5\right] \quad (10)$$

where ζ is a constant parameter that controls the nonlinear degree of the velocity. This parameter is computed by estimating the absolute difference between the current frame I_k and the previously processed frame I_{k-1}. Each iteration is updated using the smoke motion feature to speed up the level set evolution [28].

The identification of the smoke regions is difficult since the model can not distinguish between smoke regions and clouds, and the task became more difficult in images with haze or fog or in daytime captured images. To overcome this challenge, it is crucial to accurately well estimate and automatically initialize the parameters to ameliorate the segmentation process accuracy. The proposed scheme contributes by ameliorating the obtained masks and decreasing the number of false negatives ROIs.

This first stage generates a binary mask for the candidate smoke area. This mask is used for the level set algorithm initialization. The adaptive level set proved his performance in detecting the smoke borders efficiently in different frames and segmenting the image regions into smoke and background. Hence, this model requires a good parameter estimation and a precise initialization of the level set functions. The parameters of the model are determined with an empirical assessment. An overview of the scheme is presented in Figure 4.

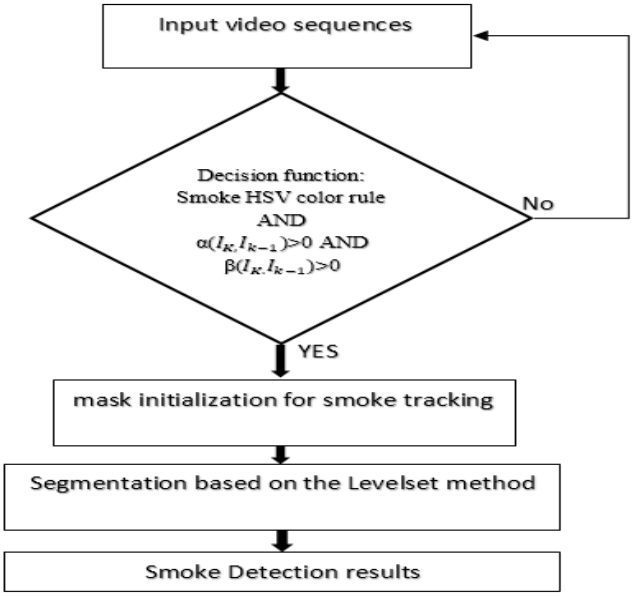


Figure 4. Workflow of the proposed smoke detection stage in video sequences.

4. Experimental Results and Discussion

4.1. CAE Configuration and Image Quality Enhancement Results Evaluation

The training and validation of the CAE model was implemented using Python programming language and with keras and tensorflow backend. Fixing and fine-tuning the hyperparameters enables us to have a robust network with good results. We tested several values to fix the parameters and opted to set the learning rate to 0.001. The trained model outputted a reconstructed video image with some quality improvement. Then, the reconstructed video image was fed to the segmentation stage to detect smoke regions.

Table 2 exposes different parameter configurations.

Table 2. Parameters configuration of the CAE model.

Input Dimension	(256, 256, 3)
Epochs number	130
Loss	Formulation combines the SSIM, MSE, MAE
Weight decay	1×10^{-5}
Optimizer	Adam
Trainable parameters	299,139

To optimize the learning process of the CAE network, a loss formulation introduced by Roy et al. (2021) [45] has been integrated. It is a weighted sum of three losses: the MSE loss, the MAE loss, and the SSIM loss.

The equation of reconstruction loss is defined by:

$$l = \alpha \times l_{MSE} + \beta \times l_{SSIM} + \gamma \times l_{MAE} \tag{11}$$

where α, β, γ are hyper-parameters determined by accurate testing $\alpha = \beta = \gamma = 0.5$ [19].

The SSIM loss is defined by:

$$l_{SSIM}(x_{ij}, \widetilde{x}_{ij}) = 1 - SSIM(x_{ij}, \widetilde{x}_{ij}) \tag{12}$$

where the formulation of the structural similarity index measure is:

$$SSIM(x, \widetilde{x}) = \left[l(x, \widetilde{x})^{\alpha} \cdot c(x, \widetilde{x})^{\beta} s(x, \widetilde{x})^{\gamma} \right] \tag{13}$$

where $l(.)$, $c(.)$, and $s(.)$ are the luminance, contrast, and structural comparison functions, respectively.

The loss function plays a crucial role in image generation and manipulation problems as it guides the model in optimizing its weights and provides insight into its performance in data modeling. Various approaches have been proposed in the literature, often relying on the mean squared error (MSE) loss for evaluation. In our approach, we employed both losses, the reconstruction loss, and the MSE loss, and we found that the reconstruction loss yielded the best performance. As depicted in Figure 5, the model achieved its lowest loss value of 0.05 at the 120th epoch. Notably, the proposed architecture of the convolutional autoencoder (CAE) yielded excellent results even with a small dataset and without data augmentation techniques.

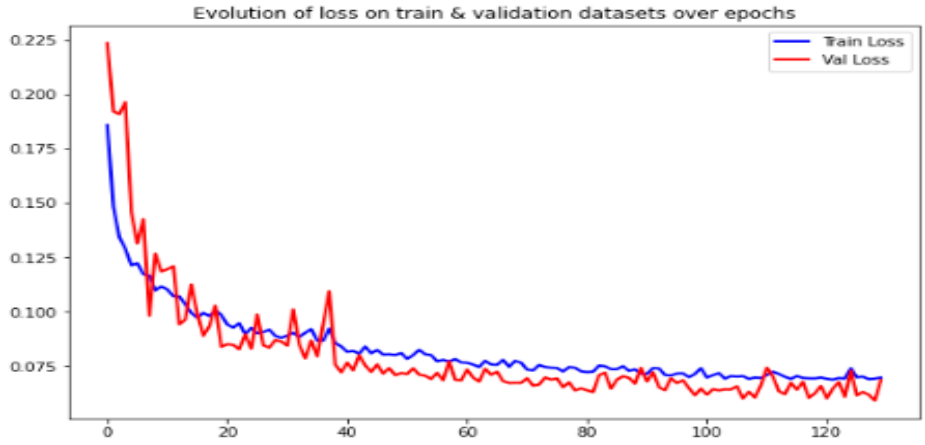


Figure 5. The reconstruction loss evolution curves.

In order to objectively evaluate the proposed CAE model in image denoising and reconstruction, SSIM, PSNR, and MSE evaluation criteria were employed.

The outcomes of the experiment conducted on a randomly selected image from the test samples are provided in Table 3. Additionally, Table 4 displays the denoising results achieved using our method in comparison to various conventional filters and a technique proposed in the literature. The results undeniably illustrate the superiority of the CAE over other denoising methods. Our CAE model excels in eliminating noise points while simultaneously retaining fine details and enhancing the overall image resolution. These findings serve as compelling evidence of the effectiveness and efficiency of our proposed approach in denoising images, highlighting its potential for practical applications. The enhanced image quality resulting from our approach will greatly facilitate the analysis of the reconstructed video images in the subsequent stages. It is important to note that our dataset comprises both clean images and noisy images with varying levels of noise. As a result, our proposed CAE is capable of performing both image reconstruction and denoising tasks simultaneously. This dual functionality enables the CAE to effectively address the challenges posed by noise in the dataset and produce high-quality, noise-free images for further investigation and analysis.

Table 3. Quantitative results of the CAE network.

Image	PSNR	SSIM	MSE	Processing Time(s)
Filtered with the proposed CAE	73.2	0.91	0.0030	0.6
Filtered with a median filter	72.3	0.79	0.0040	0.04
Noisy image	69.04	0.33	0.0081	-

Table 4. Comparative evaluation results of the proposed CAE architecture with different filters and methods using the SSIM evaluation criterion.

Filtering Type	SSIM
Noisy	0.33
Median filter	0.79
Gaussian filter	0.90
Gondara [46]	0.89
The proposed CAE	0.91

In Figure 6, a set of reconstructed resulting images obtained from the designed CAE architecture is displayed. Through visual evaluation, we observed a slight improvement in image contrast. The CAE demonstrated its capability to effectively eliminate artifacts in noisy images and accurately reconstruct the fire and smoke regions, as well as other parts of the images.

Figure 6. Construction results of the fire and smoke images from the test samples with the proposed lightweight CAE architecture.

4.2. Video Smoke Tracking Path

In this section, we provide detection results evaluation of the reconstructed smoke video images using the proposed approach, which involves foreground extraction followed by the adaptive level set algorithm. Figure 7 illustrates a selection of sample frames from the video test set. These video frames were passed through the CAE model to eliminate noise, reduce image size while preserving relevant features, and enhance the overall video image quality. Subsequently, the samples were preprocessed to extract the initial mask of the smoke region candidates. This mask was further refined using an adaptive level set model to obtain the final detection of smoke regions in each frame.

A subjective evaluation can be observed from Table 5, which shows that the proposed method successfully separated the smoke area from the background image. To verify the effectiveness of the method, we compared the segmentation results with those obtained using Fuzzy c-means (FCM), K-means, and spatial fuzzy c-means combined with the level-set algorithm (Spatial FCM). Both our method and the three algorithms used for comparison were able to segment the smoke area. However, only the proposed algorithm yielded enhanced segmentation results with a minimum of false detections. The clustering algorithms used were unable to differentiate between regions with the same texture features. For example, in images (a), (c), and (d), our benchmark was able to differentiate between different regions with the same appearance and extract the smoke area from the complex background image. It is worth noting that to identify the weaknesses of our method, Table 5 showcases some critical particular cases from the test dataset with low accuracy in detection. The results are promising and demonstrate the superiority of our method.

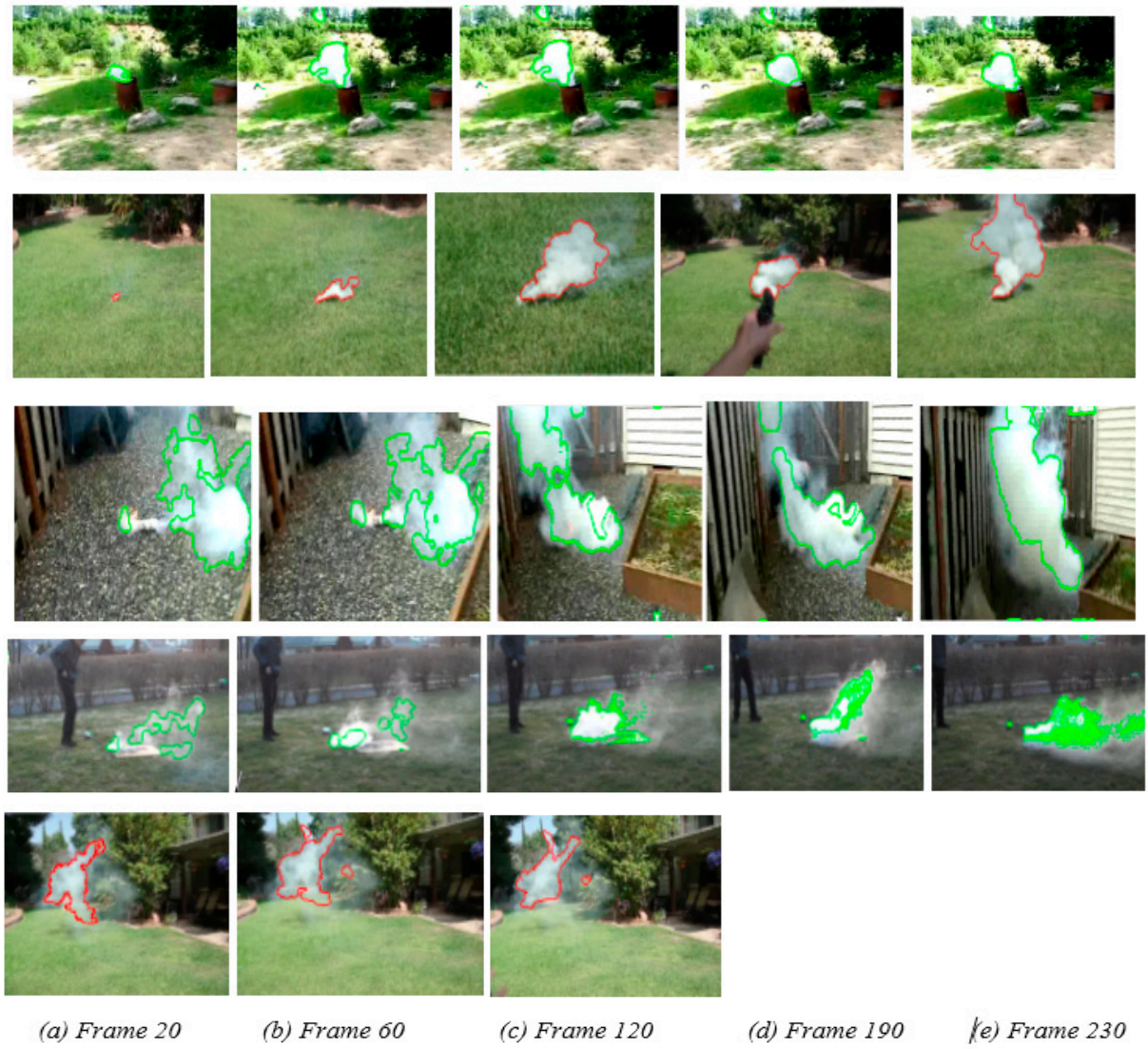
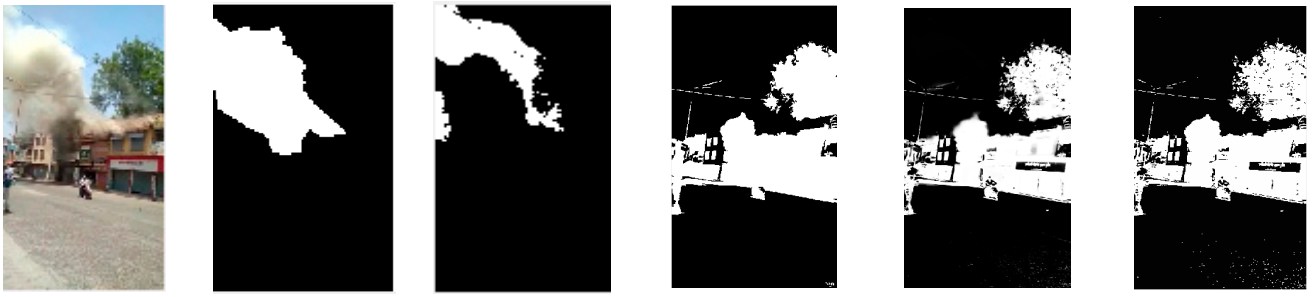


Figure 7. Smoke tracking results with the proposed method with illustration of subsequent frames extracted from five different videos from row (1) to row (5): final smoke boundaries detection results in (**a**) Frame 20, (**b**) Frame 60, (**c**) Frame 120, (**d**) Frame 190, and (**e**) Frame 230.

However, the edges of segmented regions are still not smooth enough, and the reason for this is that the model still confuses clouds with cluttered skies. This is somewhat justified because we use a smoke color model in our preprocessing steps, and the choice of the smoke color model depends on the dataset used in this work (nearby smoke and a moving camera in an environmental scene). Another reason for the low accuracy is that smoke, depending on its combustion level and density, can exhibit an array of grayish color shades, including the darkest ones, which can affect the segmentation results of our algorithm.

We believe that the strength of the proposed method lies in being a fully unsupervised benchmark by integrating a new workflow for smoke denoising and tracking, resulting in very promising segmentation accuracy and detection time. Moreover, it is accessible for employment in real-world situations.

Table 5. Visual comparison of segmentation results with different algorithms.

Original Image	Ground Truth	Our Method	FCM	Spatial FCM [47]	K-means
(a)					
(b)					
(c)					
(d)					
(e)					
(f)					

In order to illustrate quantitatively the effectiveness of the proposed benchmark, we selected 24 image frames from 7 smoke videos. These frames were chosen to represent various smoke locations under different weather conditions and situations. We calculated the Jaccard similarity index and the Dice coefficient between the segmentation results and the manually annotated images. The experimental results for the test set, including the mean, maximum, and minimum evaluation criterion values, are reported in Table 6. In addition, Table 7 exposes a quantitative evaluation of the proposed method on four videos from the test set. F_{tot} is the total number of frames in a video sequence, F_s is the number of frames containing smoke regions in a video, F_t is the number of frames detected as smoke with the proposed method, R_d is the detection rate of a video. The detection rate formula is defined as follows:

$$R_d = \frac{F_c}{F_t} \tag{14}$$

where F_c is the number of frames that are correctly classified with the proposed method and F_t is the number of frames in a video sequence.

Table 6. Quantitative evaluation of the smoke segmentation using the adaptive level set-based algorithm in terms of accuracy and processing time.

	Max	Min	Mean
Jaccard index (%)	92.1	80.5	90.1
Dice coefficient (%)	90.0	79.6	89.5
Processing time/frame (s)	5.3	4.9	5.0
Processing time/video (s)	484.9	235.4	376.3

Table 7. Experimental results of smoke detection in videos using the proposed scheme.

Video Sequences	F_{tot}	F_s	F_t	$R_d(\%)$
Video 1	675	674	670	99.20
Video 2	407	403	403	99.02
Video 3	360	360	360	100.00
Video 4	310	308	308	99.35
Video 5	150	150	150	100.00

To proceed with the evaluation process in our experiments, we evaluated the efficiency of the CAE architecture, and we compared the segmentation accuracy values obtained with the proposed pipeline with the segmentation accuracy obtained with the traditional filtering method, which is the median filter and the segmentation accuracy obtained by applying the level set method on noisy images. Table 8 reports measurements performed on 24 image frames in terms of Jaccard similarity index and dice coefficient. Based on these experiments, we can deduce that the proposed pipeline provides encouraging segmentation results by reaching more than 90% accuracy and effectively identifies smoke regions within video frames, accomplishing this task within a reasonable processing time, particularly when the regions are uniform and exhibit statistical differences from the background. The purpose of employing this approach is to eliminate the need for manual parameter configuration and enhance the segmentation process with the level set method.

Table 8. Performance evaluation of the smoke segmentation pipeline compared to other segmentation methods in terms of accuracy segmentation and processing time.

	Jaccard Similarity Index (%)		Dice Coefficient (%)		Detection Time/ Image(s)	
	Min	Max	Min	Max	Min	Max
segmentation of raw images	70.6	89.9	73.2	89.8	8.2	9
segmentation of denoised images with a median filter	64.7	87.3	65.5	86.9	4.6	5.1
segmentation of denoised images with the proposed pipeline	80.5	92.1	79.6	90.0	4.9	5.3

The experimental results show that those steps were able to help the level set function track the evolution of the smoke borders in a video sequence in an optimal time and with a minimum of false detections.

The detection of smoke is highly important in practical scenarios, particularly when dealing with fire and smoke in wildfire video images. Both qualitative and quantitative observations clearly demonstrate that the model performs well in identifying wildfire smoke in reconstructed video images, especially in detecting the appearance of small

smoke regions in the first frames. However, the lower accuracy segmentation values in particular frames are justified by the similarity between intensity regions. Due to the appearance of smoke in various color combinations and different spatial domains, making it challenging to identify using traditional conventional methods. Another contributing factor to the lower accuracy values is the manual segmentation process. In certain cases, it can be difficult to determine whether a pixel belongs to the fire, smoke, or background class. For instance, when smoke is present against a cloudy sky or when flames are obscured by smoke and segmented as background, the model detected it because of its characteristics. While the resulting segmentation is not really false, it can influence the model's performance evaluation.

5. Conclusions

Efforts to improve early detection and response to wildfires and smoke continue to evolve with advancements in technology and the use of data-driven approaches. By understanding the behavior and characteristics of smoke, it becomes possible to develop more effective strategies to prevent and mitigate their devastating effects. In this work, we introduced a new lightweight convolutional autoencoder architecture and incorporated an automatic scheme based on color and texture features characteristics of smoke and a nonlinear adaptive level set algorithm. Also, we provided a detailed explanation of the implementation and training process, as well as the experimental results and evaluation for both the reconstruction and segmentation stages in video sequences. Our proposed workflow yielded visually and quantitatively satisfactory results, effectively improving image quality, eliminating noise, and accurately detecting smoke boundaries in RGB video frames.

In future works, our objective is to conduct a thorough investigation into feature extraction and selection techniques, as well as generative models, in order to generate high-resolution and large-scale databases. Also, extend this methodology to track fires and smoke in more complex scenarios. Additionally, we plan to adapt this approach to detect fires and smoke with small appearances, as they serve as the first clue of a wildfire recipient, and implement the algorithm in an embedded system to use it for real-world applications. Ultimately, we intend to generalize our method and reproduce our findings with a diversified data pool.

Author Contributions: R.D. conceived the presented approach. R.D., M.B. and A.M. developed the theory and performed the computations. M.S., M.B. and A.M. verified the analytical methods and were involved in planning and supervising the work. M.S. and E.M. encouraged R.D. to design, and they supervised the findings of this work. All authors discussed the results and contributed to the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Experimental video data covered in this paper are available at the following http://signal.ee.bilkent.edu.tr/VisiFire/Demo/SampleClips.html accessed on 12 April 2023. For the image dataset provided by our laboratory research please contact the authors by email.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Garcia-Jimenez, S.; Jurio, A.; Pagola, M.; De Miguel, L.; Barrenechea, E.; Bustince, H. Forest fire detection: A fuzzy system approach based on overlap indices. *Appl. Soft Comput.* **2017**, *2*, 834–842. [CrossRef]
2. Tian, H.; Li, W.; Wang, L.; Ogunbona, P. Smoke detection in video: An image separation approach. *Int. J. Comput. Vis.* **2014**, *106*, 192–209. [CrossRef]
3. Sun, Y.; Feng, J. Fire and smoke precise detection method based on the attention mechanism and anchor-free mechanism. *Complex Intell. Syst.* **2023**, 1–14. [CrossRef]
4. Wang, X.; Li, M.; Gao, M.; Liu, Q.; Li, Z.; Kou, L. Early smoke and flame detection based on transformer. *J. Saf. Sci. Resil.* **2023**, *4*, 294–304. [CrossRef]

5. Sathishkumar, V.E.; Cho, J.; Subramanian, M.; Sai Naren, O. Forest fire and smoke detection using deep learning-based learning without forgetting. *Fire Ecol.* **2023**, *19*, 9. [CrossRef]
6. Lou, L.; Chen, F.; Cheng, P.; Huang, Y. Smoke root detection from video sequences based on multi-feature fusion. *J. For. Res.* **2022**, *33*, 1841–1856. [CrossRef]
7. Hashemzadeh, M.; Zademehdi, A. Fire detection for video surveillance applications using ICA K-medoids-based color model and efficient spatio-temporal visual features. *Expert Syst. Appl.* **2019**, *130*, 60–78. [CrossRef]
8. Zhong, Z.; Wang, M.; Shi, Y.; Gao, W. A convolutional neural network-based flame detection method in video sequence. *Signal Image Video Process.* **2018**, *12*, 1619–1627. [CrossRef]
9. Khatami, A.; Mirghasemi, S.; Khosravi, A.; Lim, C.P.; Nahavandi, S. A new PSO-based approach to fire flame detection using K-medoids clustering. *Expert Syst. Appl. Vol.* **2017**, *68*, 69–80. [CrossRef]
10. Sousa, M.J.; Moutinho, A.; Almeida, M. Wildfire detection using transfer learning on augmented datasets. *Expert Syst. Appl.* **2020**, *142*, 112975. [CrossRef]
11. Zhao, Y.; Tang, G.; Xu, M. Hierarchical detection of wildfire flame video from pixel level to semantic level. *Expert Syst. Appl.* **2015**, *2*, 4097–4104. [CrossRef]
12. Filonenko, A.; Hernández, D.C.; Jo, K. Fast smoke detection for video surveillance using CUDA. *IEEE Trans. Ind. Inf.* **2018**, *14*, 725–733. [CrossRef]
13. Barmpoutis, P.; Dimitropoulos, K.; Grammalidis, N. Smoke detection using spatio-temporal analysis, motion modeling and dynamic texture recognition. In Proceedings of the 2014 22nd European Signal Processing Conference, EUSIPCO, Lisbon, Portugal, 1–5 September 2014; pp. 1078–1082.
14. Liu, Z.; Yang, X.; Liu, Y.; Qian, Z. Smoke-detection framework for high-definition video using fused spatial- and frequency-domain features. *IEEE Access* **2019**, *7*, 89687–89701. [CrossRef]
15. Wang, Y.; Wu, A.; Zhang, J.; Zhao, M.; Li, W.; Dong, N. Fire smoke detection based on texture features and optical flow vector of contour. In Proceedings of the 2016 12th World Congress on Intelligent Control and Automation, WCICA, Guilin, China, 12–15 June 2016; pp. 2879–2883.
16. Appana, D.K.; Islam, R.; Khan, S.A.; Kim, J.-M. A video-based smoke detection using smoke flow pattern and spatial–temporal energy analyses for alarm systems. *Inf. Sci.* **2017**, *418–419*, 91–101. [CrossRef]
17. Novac, I.; Geipel, K.R.; Gil, J.E.D.; Paula, L.G.D.; Hyttel, K.; Chrysostomou, D. A Framework for Wildfire Inspection Using Deep Convolutional Neural Networks. In Proceedings of the IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2020; pp. 867–872.
18. He, L.; Gong, X.; Zhang, S.; Wang, L.; Li, F. Efficient attention based deep fusion CNN for smoke detection in fog environment. *Neurocomputing* **2021**, *434*, 224–238. [CrossRef]
19. Bouguettaya, A.; Zarzour, H.; Taberkit, A.M.; Kechida, A. A Review on Early Wildfire Detection from Unmanned Aerial Vehicles Using Deep Learning-Based Computer Vision Algorithms. *Signal Process.* **2022**, *190*, 108309. [CrossRef]
20. Gagliardi, A.; de Gioia, F.; Saponara, S. A real-time video smoke detection algorithm based on Kalman filter and CNN. *J. Real-Time Image Process.* **2021**, *18*, 2085–2095. [CrossRef]
21. Lu, K.; Xu, R.; Li, J.; Lv, Y.; Lin, H.; Liu, Y. A Vision-Based Detection and Spatial Localization Scheme for Forest Fire Inspection from UAV. *Forests* **2022**, *13*, 383. [CrossRef]
22. Kaabi, R.; Bouchouicha, M.; Mouelhi, A.; Sayadi, M.; Moreau, E. An Efficient Smoke Detection Algorithm Based on Deep Belief Network Classifier Using Energy and Intensity Features. *Electronics* **2020**, *9*, 1390–1411. [CrossRef]
23. Xu, G.; Zhang, Y.; Zhang, Q.; Lin, G.; Wang, Z.; Jia, Y.; Wang, J. Video smoke detection based on deep saliency network. *Fire Saf. J.* **2019**, *105*, 277–285. [CrossRef]
24. Wang, Y.H.; Hua, C.; Ding, W.; Wu, R. Real-time detection of flame and smoke using an improved YOLOv4 network. *SIViP* **2022**, *16*, 1109–1116. [CrossRef]
25. Cao, Y.; Yang, F.; Tang, Q.; Lu, X. An Attention Enhanced Bidirectional LSTM for Early Forest Fire Smoke Recognition. *IEEE Access* **2019**, *7*, 154732–154742. [CrossRef]
26. Khan, A.; Hassan, B.; Khan, S.; Ahmed, R.; Abuassba, A. Deep Fire: A Novel Dataset and Deep Transfer Learning Benchmark for Forest Fire Detection. *Mob. Inf. Syst.* **2022**, *2022*, 5358359. [CrossRef]
27. Frizzi, S.; Bouchouicha, M.; Ginoux, J.; Moreau, E.; Sayadi, M. Convolutional neural network for smoke and fire semantic segmentation. *IET Image Process.* **2021**, *15*, 634–647. [CrossRef]
28. Mouelhi, A.; Bouchouicha, M.; Sayadi, M.; Moreau, E. A Neural Adaptive Level Set Method for Wildland Forest Fire Tracking. *Int. J. Comput. Appl. Technol. Indersci.* **2021**, *67*, 289–302. [CrossRef]
29. Jin, C.; Zheng, A.; Wu, Z.; Tong, C. Real-Time Fire Smoke Detection Method Combining a Self-Attention Mechanism and Radial Multi-Scale Feature Connection. *Sensors* **2023**, *6*, 3358. [CrossRef]
30. Yu, C.; Mei, Z.; Zhang, X. A real-time video fire flame and smoke detection algorithm. *Procedia Eng.* **2013**, *62*, 891–898. [CrossRef]
31. Yigithan, D.; Behçet, T.; Gudukbay, U.; Cetin, A. Real-Time Fire and Flame Detection in Video, 1988 International Conference on Acoustics, Speech, and Signal Processing. *ICASSP-88* **2005**, *2*, 669–673.
32. Turgay, C. Fast and Efficient Method for Fire Detection Using Image Processing. *ETRI J.* **2010**, *32*, 881–890. [CrossRef]
33. Pundir, A.S.; Raman, B. Dual Deep Learning Model for Image Based Smoke Detection. *Fire Technol.* **2019**, *55*, 2419–2442. [CrossRef]

34. Yin, Z.; Wan, B.; Yuan, F.; Xia, X.; Shi, J. A Deep Normalization and Convolutional Neural Network for Image Smoke Detection. *IEEE Access* **2017**, *5*, 18429–18438. [CrossRef]
35. Toreyin, B.U.; Dedeoglu, Y.; Cetin, A.E. Contour Based Smoke Detection in Video Using Wavelets. In Proceedings of the 14th European Signal Processing Conference, EUSIPCO 2006, Florence, Italy, 4–8 September 2006; pp. 1–5.
36. Vincent, P.; Larochelle, H.; Bengio, Y.; Manzagol, P.-A. Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning, ACM, New York, NY, USA, 5 July 2008.
37. Viren, J.; Seung, S. Natural image denoising with convolutional networks. In Proceedings of the 21st International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 8–10 December 2008.
38. Chen, T.; Wu, P.; Chiou, Y. An Early Fire-Detection Method Based on Image Processing. In Proceedings of the 2004 International Conference on Image Processing, 2004. ICIP'04, Singapore, 24–27 October 2004; pp. 1707–1710.
39. Chan, T.F.; Vese, L.A. Active contours without edges. *IEEE Trans. Image Process.* **2001**, *10*, 266–277. [CrossRef] [PubMed]
40. Chan, T.F.; Sandberg, B.Y.; Vese, L.A. Active contour without edges for vector-valued images. *J. Vis. Commun. Image Represent.* **2000**, *11*, 130–141. [CrossRef]
41. Vese, L.A.; Chan, T.F. A multiphase level set framework for image segmentation using the Mumford and Shah model. *Int. J. Comput. Vis.* **2002**, *53*, 271–293. [CrossRef]
42. Tsai, A.; Yezze, A., Jr.; Willsky, A.S. Curve evolution implementation of the Mumford–Shah functional for image segmentation, denoising, interpretation, and magnification. *IEEE Trans. Image Process.* **2001**, *10*, 1169–1186. [CrossRef]
43. Rousson, M.; Brox, T.; Deriche, R. Active unsupervised texture segmentation on a diffusion based feature space. In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, USA, 18–20 June 2003; pp. 699–704.
44. Wang, B.; Gao, X.; Tao, D.; Li, X. A nonlinear adaptive level set for image segmentation. *IEEE Trans. Cybern.* **2014**, *44*, 418–428. [CrossRef]
45. Roy, M.; Kong, J.; Kashyap, S.; Pastore, V.P.; Wang, F.; Wong, K.C.L.; Mukherjee, V. Convolutional autoencoder based model HistoCAE for segmentation of viable tumor regions in liver whole-slide images. *Sci. Rep.* **2021**, *11*, 139. [CrossRef]
46. Gondara, L. Medical image denoising using convolutional denoising autoencoders. *arXiv* **2016**, arXiv:1608.04667v2.
47. Li, B.N.; Chui, C.K.; Chang, S.; Ong, S.H. Integrating spatial fuzzy clustering with level set methods for automated medical image segmentation. *Comput. Biol. Med.* **2011**, *41*, 1–10. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

A Multi-Stage Adaptive Copy-Paste Data Augmentation Algorithm Based on Model Training Preferences

Xiaoyu Yu [1,*], Fuchao Li [2], Yan Liu [1] and Aili Wang [3]

[1] College of Electron and Information, University of Electronic Science and Technology of China, Zhongshan Institute, Zhongshan 528402, China
[2] Department of Computing, Guangdong University of Science and Technology, Dongguan 523083, China; 2020023766@m.scnu.edu.cn
[3] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; aili925@hrbust.edu.cn
* Correspondence: yuxy@zsc.edu.cn

Abstract: Datasets play an important role in the field of object detection. However, the production of the dataset is influenced by objective environment and human subjectivity, resulting in class imbalanced or even long-tailed distribution in the datasets. At present, the optimization methods based on data augmentation still rely on subjective parameter adjustments, which is tedious. In this paper, we propose a multi-stage adaptive Copy-Paste augmentation (MSACP) algorithm. This algorithm divides model training into multiple training stages, each stage forming unique training preferences for that stage. Based on these training preferences, the class information of the training set is adaptively adjusted, which not only alleviates the problem of class imbalance in training, but also expands different sample sizes for categories with insufficient information at different training stages. Experimental verification of the traffic sign dataset Tsinghua–Tencent 100K (TT100K) was carried out and showed that the proposed method not only can improve the class imbalance in the dataset, but can also improve the detection performance of models. By using MSACP to transplant the trained optimal weights to an embedded platform, and combining YOLOv3-tiny, the model's accuracy in detecting traffic signs in autonomous driving scenarios was improved, verifying the effectiveness of the MSACP algorithm in practical applications.

Keywords: object detection; data augmentation; class imbalance; YOLOv3-tiny

1. Introduction

The total number of different classes in the real world is different, which leads to the class imbalance of datasets when collecting data [1]. Currently, most competitive models rely on huge amounts of data for training. However, the large amount of data can lead to more extreme class imbalances; that is, a few classes have a very large sample size, while most classes have a small sample size. The sample size of the classes showed a long-tail distribution [2]. When the model is trained, it is easy to overfit the classes with insufficient samples.

Because datasets are too expensive to produce, researchers focus on algorithms to optimize datasets. Two common data processing methods for class imbalance are sample sampling and cost-sensitive learning. For the sample sampling technology, the main idea of this method is to directly change the feature distribution of the dataset by adding or deleting the sample size of the class, specifically by reducing the sample size of the larger class or increasing the sample size of the smaller class. Methods that reduce a large number of classes are often referred to as down-sampling, and methods that increase a small number of classes are often referred to as up-sampling. Random sampling [3] is one of the most commonly used methods of sampling techniques. Many new sampling methods have been proposed in recent years. Dynamic curriculum learning (DCL) [4] achieves the rebalance of

classes by dynamically sampling data. If a class is sampled many times during training, it will reduce the probability that the class will be sampled in the following training. This sampling strategy allows the model to focus more on learning the tail classes. Online hard example mining (OHEM) [5] performs additional training by online selecting candidate regions of interest with high loss values. The classes in the tail usually produce high loss values during training. Although this method has a good improvement effect, it also takes up more memory. At the same time, it is difficult to transfer the optimization algorithm to different models.

Different from sampling techniques, data augmentation [6] can improve the size and quality of the training dataset, which allows the model to achieve higher performance after training. It is simple for data augmentation to be applied to different models. Chen P [7] proposed an erasure data augmentation method: GridMask, which discarded some regions on the image to generate new data. The discarded regions are equivalent to adding a regular term on the network to reduce the occurrence of overfitting. But the dropped areas actually change the gray level of pixels in that area to zero. These regions still need to participate in the training of the model, which will reduce the training efficiency to a certain extent. Bochkovskiy [8] proposed a data augmentation called Mosaic, which mixed four training images with different contexts to generate one training image, reducing the need for a large batchsize. In the process of training, the model seeks the optimal data augmentation strategy, but the optimization method has a large demand on computing power. Huang Siw [9] proposed a cross-domain adaptive data augmentation method based on generative adversarial networks (GAN). This method can generate images of different areas of the same scene, such as spring and winter, day and night. Among various data augmentation methods, Ghiasi [10] has proved that Copy-Paste augmentation is a simple and effective optimization method. This method randomly copies and pastes the instances in the training dataset to generate a large amount of data for training. Copy-Paste augmentation methods have been used to optimize data in a variety of situations. Dwibedi [11] applies the Copy-Paste augmentation method to generate data for indoor scenes. Although this method can greatly improve the performance of the model, this process is not conducive to large-scale data generation due to the use of more complex optimization algorithms and other datasets that are used in the data generation process.

Most of the previous data augmentation strategies were designed and implemented manually. Different data augmentation strategies have different effects on the same model. The parameters that match the data augmentation work for one model, but may not work for another. There is an optimal match between the model and the data augmentation strategy, while manually designed or randomly matched data augmentation strategies ignore the characteristics between the model and the dataset. The data augmentation strategy does not achieve an optimal match between the model and the dataset. Most of the optimization based on Copy-Paste augmentation adopts manual design or random matching strategy, which is not conducive to the optimal effect of this method. Automatic augmentation [12,13] proposes optimized data augmentation strategies for the field of image classification. The main idea of automatic augmentation is to assemble data augmentation strategies into a search space. According to the current data, the model uses the search algorithm to find the best strategy between the model and the data in the search space, which further improves the accuracy of the model. Although automatic augmentation solves the phenomenon that data augmentation relies too much on manual adjustment of parameters to a certain extent, the process of automatic augmentation has high requirements on computing power.

In this paper, based on the work of CPA [14], we further propose further propose a multi-stage adaptive adjustment Copy-Paste augmentation (MSACP) algorithm. This algorithm can ease the dependence of Copy-Paste augmentation algorithm on artificial adjustment parameters, and make the amplification degree of data augmentation more flexible. This algorithm uses the training preferences of the model at different stages to

expand different sample numbers for different classes. The main contributions of this paper are summarized as follows:

1. A multi-stage adaptive Copy-Paste augmentation algorithm based on training preferences of different stages is proposed. By extracting the training preferences of different stages, the algorithm can adaptively expand the appropriate sample size for the class with insufficient information in different training stages. The amplified information can be adjusted according to the training information of different stages, which can avoid the phenomenon of overfitting or underfitting after the amplification of unreasonable information in a certain stage.

2. To prove the effectiveness and generality of MSACP, the models in the field of object detection such as Grid RCNN [15], ATSS [16], and YOLOv3 [17] are used for comparison. At the same time, the algorithm is also applied to the embedded platform of automatic driving. From the experimental results, the MSACP algorithm has a better AP after embedding in different object detection models. In the actual automatic driving scenario, the algorithm can improve the detection accuracy of the embedded platform.

2. Related Work

2.1. Cost-Sensitive Learning Techniques

Class imbalance is a common problem in imbalance problems [1]. Class imbalance means that there is a significant quantitative gap among classes. In the process of model training, the class with a small sample size tends to produce a larger loss value, but the model does not pay much attention to this part of the loss value. Cost-sensitive learning techniques mainly change the training principle of the model. The class with a small number of samples is more likely to produce difficult samples. Therefore, the penalty generated by the samples of a larger number of classes is reduced, and the penalty generated by the samples of a smaller number of classes is increased. Focal Loss [18] used weight parameters to increase the loss value generated by difficult samples and reduce the loss value generated by simple samples, so that the model paid more attention to the learning of difficult samples. Adaptive class suppression loss (ACSL) [19] uses the confidence of the model prediction to decide whether to suppress the gradient produced by the negative label, where the negative label refers to the class with a large number of samples. J Tan [20] proposed an equalization loss function that ignores the gradient produced by the head class. This is equivalent to enhancing the gradient generated by the rare class in the tail, increasing the model's attention to the rare class. Aditya [21] proposed a logical adjustment based on long-tail learning, which uses the label distribution of the training set to modify the logit output in the loss function. Although this method can improve the prediction effect of the model for rare classes, the prediction effect of the majority class samples will be slightly reduced. Li [22] proposed a Targeted supervised contrastive (TSC) based on the similarity of features to improve the performance of the model's long-tail visual detection. Yin Cui [23] proposed a new class balance loss based on the concept of effective samples to improve the prediction effect of minority samples. Although these methods often outperform sampling techniques in terms of performance improvement, they are more complex in principle and design.

2.2. Data Augmentation

Compared to the cost-sensitive learning techniques, this is a simple method for data augmentation to migrate to a different model. The methods can generate more and different data for model training under the condition of limited data, and even generate high-quality images. This can restrain the phenomenon of overfitting during the training of the model and improve its own generalization ability. The commonly used data augmentation methods include symmetry transformation, gray change, image color, brightness, saturation adjustment, etc. In recent years, more effective methods have been proposed. Yun [24] proposed a data augmentation method: CutMix, which pastes other category information on the randomly erased region and effectively solves the problem of low efficiency of the region discarding strategy. Kisantal [25] copied and pasted small objects in the training

dataset many times to increase the number of such samples. It is effective for models that are good at detecting small objects. Dvornik [26] increased the number of training samples with manual production, so as to reduce the overfitting phenomenon of the model and improve its generalization ability. Li CL [27] generated a large amount of abnormal data based on the Copy-Paste augmentation method. Georgakis G [28] pasted the location of the object reasonably according to the semantic information and context of the scene. But this process requires a large amount of computing power. Yun [29] optimized the Copy-Paste augmentation algorithm by a generative adversarial network to alleviate the gap between different domains and generate more diverse images. But this also increased the computing power required by the model. Zhang [30] proposed a new Copy-Paste augmentation method, Multi-Modality Cut and Paste, for generating 3D images to alleviate the problem of insufficient training data under multi-modal conditions. Cubuk [31] proposed a new automatic augmentation algorithm (RandAugment), which changed the augmentation intensity to a fixed level to reduce the size of the search space. Lin [32] proposed Patch AutoAugment to divide the original image into multiple regions. The patches method is used to achieve the best enhancement strategy for automatic search and reduce the computational cost of the model.

To a certain extent, the application of automatic augmentation reflects that the idea of adaptive adjustment based on the model training state is helpful in optimizing data augmentation. For Copy-Paste augmentation algorithms, manual design or random matching optimization methods are not scientific enough. These methods amplify an unreasonable number of samples, which can lead to overfitting or underfitting.

3. Methods

3.1. Overview

The MSACP training process is shown in Figure 1. First, there are two key pre-processing steps. The first pre-processing is to build an online database of class information. This library was built to reduce the time spent on each training dataset update and to simplify the process of updating the training dataset. The online class information database contains the information of all the classes in the training dataset. The semantic information of these classes is collected and classified by label files. When the training dataset needs to be updated, the MSACP algorithm only needs to extract the corresponding information from the online class information library. The first update of the training dataset is based on the evaluation results of the pre-training. The model begins formal training using the training dataset. In formal training, the MSACP algorithm divides the training amount of the model into n training stages according to the hyper-parameter n.

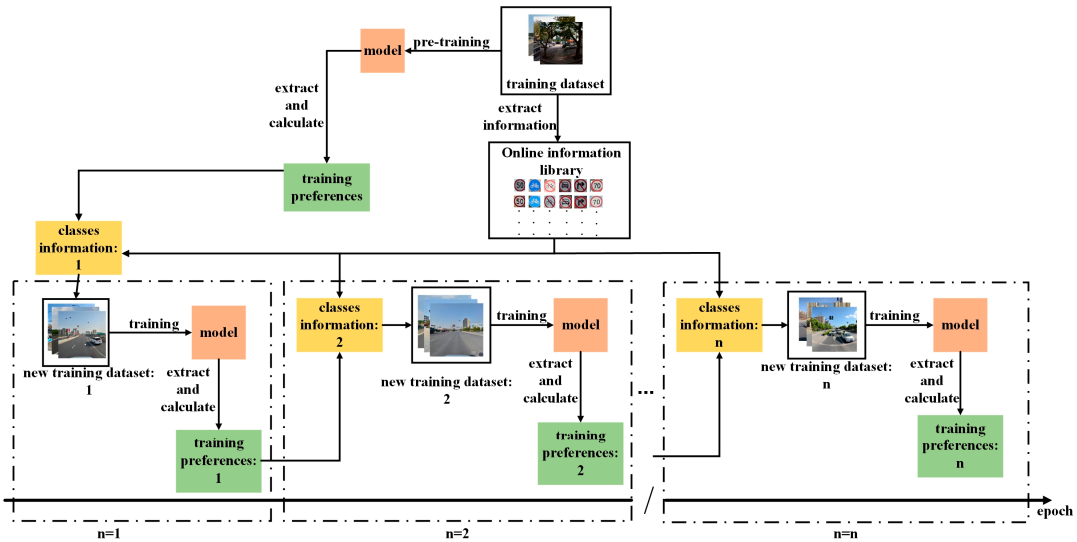


Figure 1. The training process of MSACP algorithm embedded in the model is shown.

MSACP extracts the training information of the current stage before starting the next stage. The training information is turned into training preferences for the current stage, and then an expanded number of categories is generated. According to this result, the MSACP algorithm extracts the corresponding class information from the online class information library. This can not only alleviate the class imbalance problem by expanding the class information, but also adjust the class information after each stage of training, forcing the model to pay more attention to the class with poor training effect in the next stage.

3.2. The Multi-Stage Training Preference of the Model

In the work of CPA [14], the training preference of the model is proposed. The training preference of the model refers to the model better grasping the features of some classes in training, and giving higher confidence in predicting these classes. Meanwhile, different models are also good at detecting different classes. That is to say, in the same datasets, different models are good at detecting different classes. However, it is found that the training preference of the model is different in different training stages by further experiments. In the CPA algorithm, the model training preference is mainly based on the evaluation results of pre-training. According to the training preference, the CPA algorithm will provide information compensation for each class to alleviate the impact of class imbalance on model training. But this training preference cannot reflect the whole training situation of the model. The expanded sample size may not be suitable for the entire training process of the model.

To provide an intuitive explanation of the above analysis, the model YOLOv3 was trained with the original COCO datasets. Figure 2 shows the evaluation results of these 5 nodes. The figure only shows the top 5 categories with the highest values. These 5 nodes are the 50th, 100th, 150th, 200th, and 250th epoch, respectively. From the figure, it can be seen that the evaluation results of the 200th epoch show a significant decrease compared to the results of the 100th and 150th epoch, and the training of the model is unstable. For the top five classes, there are differences in the evaluation results between the five epochs. The class giraffe is highest at epoch 50, but does not appear in the top five highest classes at epoch 100. The class bear does not appear in the 50th epoch, but appears in the next four extracts. The class bus does not appear in the top five highest epochs at the 200th epoch, but appears in the other four extractions. The four classes of toilet, train, frisbee, and stop sign appear only once in these five extractions.

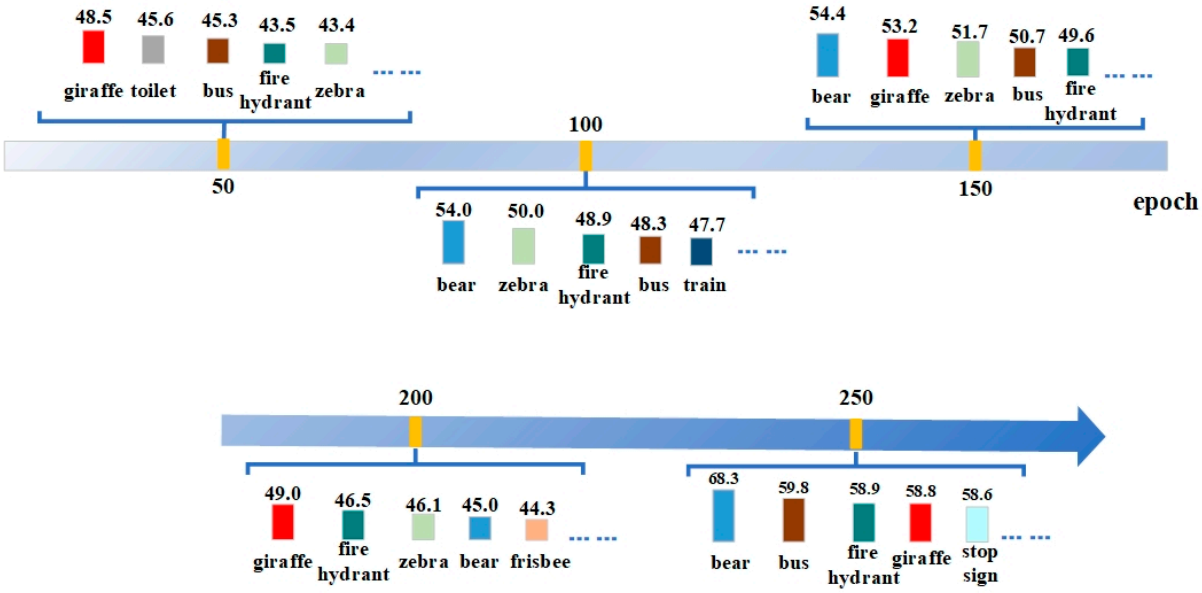


Figure 2. The evaluation results of model YOLOv3 in different stages on COCO dataset.

The training preferences of the model will be different in different training stages. Therefore, it is not appropriate to reflect the training preference of the model only from the evaluation results of pre-training. In order to master the training preferences of different stages, the commonly used improvement method is to calculate the training preferences based on the evaluation results in real time, and then update the training dataset constantly. This is similar to difficult sample mining, which checks loss values or other indicators in real time to mine the information that the model needs to focus on learning. For the Copy-Paste augmentation algorithm, the main data amplification method is to insert the synthesized image in the training dataset, where the new training dataset is used to train the model. If real-time sampling is applied to the Copy-Paste augmentation algorithm, it means that the training dataset needs to be updated frequently, and it takes time to generate images and import a new training set. The frequent updating of the datasets results in a long training time. If the dataset is updated frequently, it does not give the model much additional class information to learn.

3.3. The Design of MSACP Algorithm

The model in the field of deep learning is usually regarded as a black box because the parameters of the model are uncertain during the training process. How does class imbalance affect model training? For supervised learning, the number of samples in the datasets and the model structure are known, which is a priori knowledge. The probability that the model predicts samples is the posterior probability. The class information such as speed limit signs, warning signs, and so on are in the TT100K dataset and are assumed to be independent of each other. The class information in the training dataset can be written as D = {(x_1,y_1), (x_2,y_2), (x_3,y_3), ... , (x_n,y_n)}, where x is the sample with class label y, and n is the number of classes. If the training dataset contains samples with {m_1, m_2, m_3, ... , m_m}, the probability that a predicted sample m belongs to class x_i can be written as Equation (1):

$$p(x_i|\mathrm{m}) = \frac{p(\mathrm{m}|x_i) \times p(x_i)}{\sum_1^n p(\mathrm{m}|x_i) \times p(x_i)} \tag{1}$$

where $p(x_i|\mathrm{m})$ represents the probability that sample m is predicted by the model as class x_i. $p(x_i)$ is the prior probability of this class, which represents the proportion of the number of samples in the training dataset. $p(\mathrm{m}|x_i)$ is the conditional probability, representing the class x_i of the training dataset. Equation (1) can be further simplified, as shown in Equation (2):

$$p(x_i|\mathrm{m}) = \frac{p(\mathrm{m}|x_i) \times p(x_i)}{p(m)} \tag{2}$$

Among them, $p(x_i)$ is the proportion of the number of samples of class x_i and the total number of samples in the training dataset. The conditional probability $p(\mathrm{m}|x_i)$ and prior probability $p(x_i)$ in the Bayesian model are related to the number of samples in the training dataset, so the class imbalance can affect the magnitude of these two probabilities. If c_- is used to represent the class with a large number of samples, and c_+ is used to represent the class with a small number of samples, then the posterior probabilities of these two classes can be calculated by Bayes, as shown in Equations (3) and (4):

$$p(\mathrm{c}_-|\mathrm{m}) = \frac{p(\mathrm{m}|\mathrm{c}_-) \times p(\mathrm{c}_-)}{p(m)} \tag{3}$$

$$p(\mathrm{c}_+|\mathrm{m}) = \frac{p(\mathrm{m}|\mathrm{c}_+) \times p(\mathrm{c}_+)}{p(m)} \tag{4}$$

Because the number of samples for class c_+ in the training dataset is small, the values of $p(\mathrm{m}|\mathrm{c}_+)$ and $p(\mathrm{c}_+)$ are also small, which results in the posterior probability $p(\mathrm{c}_+|\mathrm{m})$ becoming smaller. Because of the influence of class imbalance, the posterior probability of class c_ becomes larger, so the prediction result of the model will be biased to c_. To reduce

the effect of class imbalance, the Copy-Paste enhancement algorithm is used to expand the class information with insufficient samples. This method is used to improve the prior probabilities of these classes and modify the prediction results of the model.

Due to its unique design, the model is easily able to grasp certain categories during training, which forms the training preferences of the model. Under the same dataset, the detection performance of the model is not positively correlated with the number of samples in the dataset. Although the number of samples in the training set is not very large for some categories, the model can accurately detect these categories. Therefore, simply using Copy-Paste enhancement algorithms to increase the sample size of a class lacks the support of principled approaches. After the analysis in Figure 2, the training preferences will change at different training nodes. These changes are used to update the category information of the training set, correct the training direction of the model, and further improve the detection performance of the model.

How to choose an appropriate time point to update the training dataset and open a new training stage? To ensure the simplicity and generality of MSACP algorithm, the training volume of the model is equally divided into stages. The hyper-parameter n is to represent the number of stages. Before starting the next training stage, MSACP algorithm adaptively adjusts the class information of the training dataset according to the training situation of the current stage. The key of MSACP algorithm is to amplify the appropriate class information according to the training preference of different stages. The computed expression for this training preference is given in Equation (5):

$$Class\ preference\ (\mathrm{CP}_i) = p\left(AP_i^{sf} - AP_i^{sl}\right) \times \left(AP_i^{sl} - mAP^{sl} + T\right) \tag{5}$$

where CP_i reflects the degree of preference of the model in the stage. AP_i is the evaluation result for each class. i is id of the class. sf is the evaluation result of the first epoch in this stage. Then sl is the evaluation result of the last epoch in this stage. $p(AP_i^{sf} - AP_i^{sl})$ serves as an enhancement coefficient to dynamically adjust the preference degree of classes, written as follows:

$$p\left(AP_i^{sf} - AP_i^{sl}\right) = \begin{cases} 1,\ AP_i^{sf} - AP_i^{sl} < 1 \\ AP_i^{sf} - AP_i^{sl},\ AP_i^{sf} - AP_i^{sl} \geq 1 \end{cases} \tag{6}$$

When $AP_i^{sf} - AP_i^{sl}$ is greater than 1, it means that in this stage, the detection effect of the model for the class is reduced and the training preference is enhanced. When $AP_i^{sf} - AP_i^{sl}$ is less than 1, it means that the detection effect of the model for the class remains stable or has been improved in this stage. The training preference is then kept constant. After obtaining the training preferences for the current stage, the MSACP algorithm calculates the number of samples that need to be expanded in this stage. According to the results, the appropriate amount of amplification is provided for the copy–move augmentation algorithm to adjust the class information in the training dataset. The expression is shown in Formula (7):

$$n_i' = CP_i \times n_i \times f\left(AP_i^{sl} - mAP^{sl} + T\right) \tag{7}$$

T is the regulation coefficient. The number of classes to be amplified is adjusted by the size of T. Finally, this result needs to be processed by the normalization, as shown in Formula (8):

$$Y_i = S_1 + \frac{(S_2 - S_1)}{[Max(n') - Min(n')]} \times \left[n_i' - Min\left(n'\right)\right] \tag{8}$$

(S_1, S_2) is the normalized range, Y_i is the final output result of the class. n' is the set of n_i'.

4. Results

In this part, the effectiveness and generality of the MSACP algorithm are verified by experiments from multiple perspectives. First, the basic environment and dataset used in the experiment are introduced in Section 4.1. Then, the validity and generality of the MSACP algorithm are verified with the TT100K dataset in Sections 4.2 and 4.3. Finally, the results and phenomena during the experiments are analyzed in Section 4.4.

4.1. Experimental Environments

All experiments were performed using Ubuntu 20.04 (Canonical, Cape Town, South Africa) and four RTX-3090 GPUs (NVIDIA, Santa Clara, CA, USA). In terms of datasets, the MSACP algorithm is validated using the TT100K [33] dataset. Only classes with a sample size greater than 100 are used in the experiment. There are 42 classes. The number of classes in TT100K is randomly divided into training datasets, validation datasets, and test datasets at a ratio of 7:2:1. The MSACP algorithm operates on the training dataset.

4.2. Performance Testing of MSACP in Different Models

In this section, the effectiveness of MSACP is tested on Grid RCNN [15], ATSS [16], and AutoAssign [34]. It is necessary to examine the effectiveness of the MSACP algorithm by using models with different structures. The RCNN and ATSS models belong to the two-stage range. The AutoAssign model belongs to the one-stage range. The epoch of all three models is 12. Two sets of experiments were set up for comparative analysis. The first comparison method was the origin, where the model was trained on the original dataset. The other comparison method was the CPA algorithm, which is for special cases when n = 1. The model was trained with the training dataset optimized by the CPA algorithm. It can be seen from the three tables that the CPA algorithm is obviously better than the original TT100K dataset for training. The MSACP algorithm makes up for the deficiency of the CPA algorithm in expanding sample information. The algorithm can continuously modify the expanded sample size according to the training preferences of different stages.

There are two key steps to complete before formal training. First, an online class database needs to be created to simplify the process of updating the training dataset. The class information in the training dataset is cut according to the corresponding label files, and then the information is classified by category. In addition, this class information is processed by a variety of data augmentation, such as color shifts, flips, random occlusion, etc. The class information in the online library is extracted when the training dataset needs to be updated. The second step is to pre-train the model with the original datasets. The evaluation results of pre-training are extracted and the training preferences of the model in the pre-training are calculated. The calculated results are used to optimize the class information in the training dataset. This training dataset will be used for the first stage of formal model training. When n is set to two, it means that the overall training amount is divided into two training stages by the MSACP algorithm. Before starting the next stage, training preferences are calculated based on the evaluation results of the current stage. Based on this result, the MSACP algorithm can amplify the appropriate amount of information for each class, which can reduce the phenomenon of overfitting or underfitting.

As shown in Table 1, the CPA algorithm has an obvious improvement effect on Grid RCNN because the class imbalance of the TT100K dataset is serious. In the three optimization experiments of the MSACP algorithm, the overall improvement effect is best when n is equal to four. The AP is increased by 1.2% compared with the CPA algorithm. Since the epoch of Grid RCNN is 12, the model will calculate the training preferences based on the evaluation results when trained to epoch = 3, epoch = 6, and epoch = 9, respectively. The MSACP algorithm updates the training set based on the generated results. After the new training set is imported, the model begins the next stage of training. When n = 2 and n = 3, the corresponding AP increased by 0.3% and 1% compared to the CPA algorithm, respectively.

The MSACP algorithm is embedded into the ATSS, and the results of the experiment are shown in Table 2. In the three optimization experiments, the overall improvement effect is best when n is set to three. The *AP* is increased by 1.2% compared with the CPA algorithm. When n = 2 and n = 4, the *AP* increases by 0.4% and 0.3%, respectively, compared with the CPA algorithm. For Table 3, the MSACP algorithm was embedded into the AutoAssign model. In three optimization experiments, the overall improvement effect is best when n is set to two. The *AP* is improved by 3.5% compared with the CPA algorithm. When n = 3 and n = 4, the *AP* is increased by 2.8% and 2.6% compared with the CPA algorithm, respectively. The MSACP algorithm provides phased optimization for the training dataset according to the training information, which makes the information distribution actively adapt to different stages of model training.

Table 1. The training results are compared by the Grid RCNN model (epoch = 12).

Method	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
origin	43.4	53.4	51.0	8.7	58.5	81.7
CPA	49.7	62.3	58.9	10.7	66.7	84.7
n = 2	50.0	62.6	58.8	9.6	66.7	84.7
n = 3	50.7	63.4	59.5	10.5	68.4	85.7
n = 4	50.9	63.5	59.5	11.1	68.6	86.7

Table 2. The training results are compared by the ATSS model (epoch = 12).

Method	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
origin	39.3	51.1	45.7	15.5	48.8	69.4
CPA	48.4	62.1	56.4	13.7	61.2	84.2
n = 2	48.8	62.4	56.9	13.6	61.5	85.5
n = 3	49.0	62.7	57.0	13.5	61.6	85.5
n = 4	48.7	62.4	56.6	13.7	61.3	85.5

Table 3. The training results are compared by the AutoAssign model (epoch = 12).

Method	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
origin	48.0	66.2	55.5	25.8	58	75.6
CPA	49.9	66.5	58	21.4	60.3	80.7
n = 2	53.4	71.6	61.9	22.9	64.8	82.1
n = 3	52.7	70.9	60.7	23.6	63.8	82.5
n = 4	52.5	70.5	61.3	23.5	63.7	80.8

4.3. Performance Detection of MSACP under Different Training Amounts

The MSACP algorithm divides the total training amount into multiple stages on average. Before starting the next stage of training, the class information of the training dataset will be adjusted adaptively according to the training situation of the current stage. The amount of training in each stage is related to the amount of the hyper-parameters, epoch and n. In Section 3.2, the epochs of the models were all 12. It is necessary to use the MSACP algorithm to examine models with a different epoch.

In this section, three models with more epochs are selected, including SSD512 [35], NAS FGN [36], and YOLOv3 [17]. The SSD512 and YOLOv3 models belong to the one-stage range. The NAS FGN model is a classic feature pyramid network. Their experimental results are listed in Tables 4–6, respectively. For Table 4, the MSACP algorithm is tested with the model SSD512. The epoch of this model is 24. In these three experiments, the overall improvement effect of model SSD512 is the best when n = 2. The *AP* reaches 52.9%, which is improved by 0.5% over the CPA algorithm. When n = 3 and n = 4, the corresponding AP increases by 0.2% and 0.4% compared to the CPA algorithm, respectively. MSACP adaptively adjusts the class information of the training dataset according to the training condition of the model SSD512 to improve the training quality.

The effect of the MSACP algorithm was tested with the NAS FGN model. The results are shown in Table 5. When n = 4, the overall effect of model NAS FGN is the best, reaching 40.8%, which is improved by 1.2% compared with CPA. When n = 2 and n = 3, the corresponding *AP* of our proposed method increases by 0.4% and 0.5% compared to the CPA, respectively. MSACP adaptively adjusts the class information of the training dataset according to the training condition of the NAS FGN, which forces the model to pay more attention to the poorly learned classes in different training stages.

Table 4. The training results are compared by the SSD512 model (epoch = 24).

Method	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
origin	50.4	74.9	56.6	22.9	61.9	83.1
CPA	52.4	75.9	59.3	22.5	64.2	84.1
n = 2	52.9	76.4	59.6	21.7	64.6	84.9
n = 3	52.6	76.7	58.5	22.3	64.0	84.7
n = 4	52.8	76.2	59.1	23.3	64.2	85.5

Table 5. The training results are compared by the NAS FGN model (epoch = 50).

Method	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
origin	25.9	41.4	27.4	15.7	32.9	51.2
CPA	39.6	56.3	43.5	16.2	46.8	81.6
n = 2	40.0	56.8	44.4	16.0	47.5	81.0
n = 3	40.1	56.8	44.6	16.8	47.7	81.2
n = 4	40.8	57.5	45.4	16.4	49.0	82.4

Table 6. The training results are compared by the YOLOv3 model (epoch = 273).

Method	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
origin	50.3	76.8	58.3	37.8	59.3	62.1
CPA	57.9	84.9	66.3	37.2	68.3	76.7
n = 2	58.3	85.6	67.7	38.6	68.8	76.6
n = 3	58.4	85.5	68.1	39.7	68.9	75.2
n = 4	58.6	85.8	67.4	38.0	69.4	77.4
n = 5	59.0	86.2	68.2	39.1	69.6	77.3

According to Table 6, when n = 5, the AP of model YOLOv3 reached 59.0%, which is increased by 1.1% compared with CPA. When n = 2, n = 3, and n = 4, the corresponding AP increases by 0.4%, 0.5%, and 0.7% compared to the CPA algorithm, respectively. From these three experiments, MSACP can still improve the model with a large number of epochs. Each model will show different tendencies to different classes in training because of its own unique structure. MSACP can provide dynamic information compensations for model training according to this tendency, which provides a reasonable enhancement amplitude for the Copy-Paste augmentation algorithm. Therefore, in the process of model training, each class can receive reasonable attention and learning, which avoids the phenomenon of overfitting or underfitting caused by unreasonable information expansion, and improves the robustness of the model.

4.4. Discussions

To analyze the experimental phenomenon in detail, the experimental results of AutoAssign are visually analyzed. The number of expanded samples after CPA optimization and the evaluation results of each class are shown in Figure 3. The horizontal coordinate is the class of the TT100K dataset. The hyper-parameters e and T in the CPA algorithm are set to 2.5 and 0.05, respectively. It can be seen from Figure 3 that 15 classes in the TT100K dataset are not amplified by the CPA algorithm. This is due to the fact that each of these

classes has a high AP or contains a very large sample size. Other classes are amplified because they contain less information in the training dataset and their APs are not ideal.

Two sets of experimental data for the model AutoAssign are visualized, including the experiments with n = 2 and n = 3 in Table 3. The results of updating the model AutoAssign when n = 2 is shown in Figure 4. The blue bar is the number of samples in the original training dataset. The orange bar is the number of samples amplified by the MSACP algorithm. The height of the two columns is the number of samples in the updated training dataset. The broken line in the figure is the evaluation results when the model AutoAssign is trained to epoch = 6. The training preference is calculated by the MSACP algorithm to amplify the appropriate class information for each class according to the current training situation. In this data update, there are 11 categories that have not been amplified in the TT100K dataset.

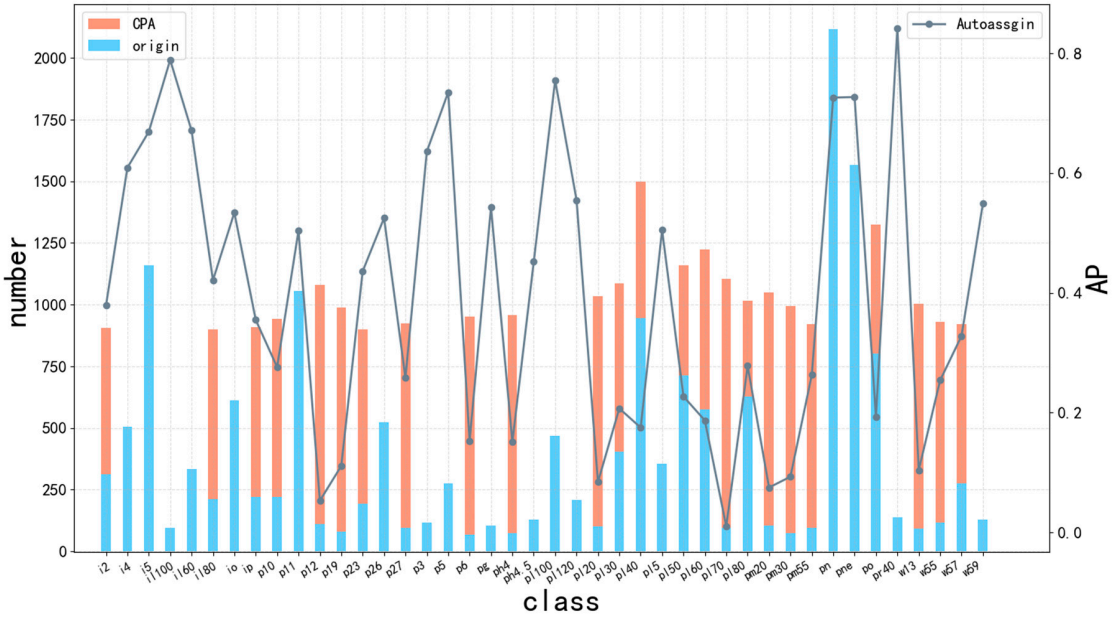


Figure 3. Updated training dataset by the CPA and evaluation results of AutoAssign.

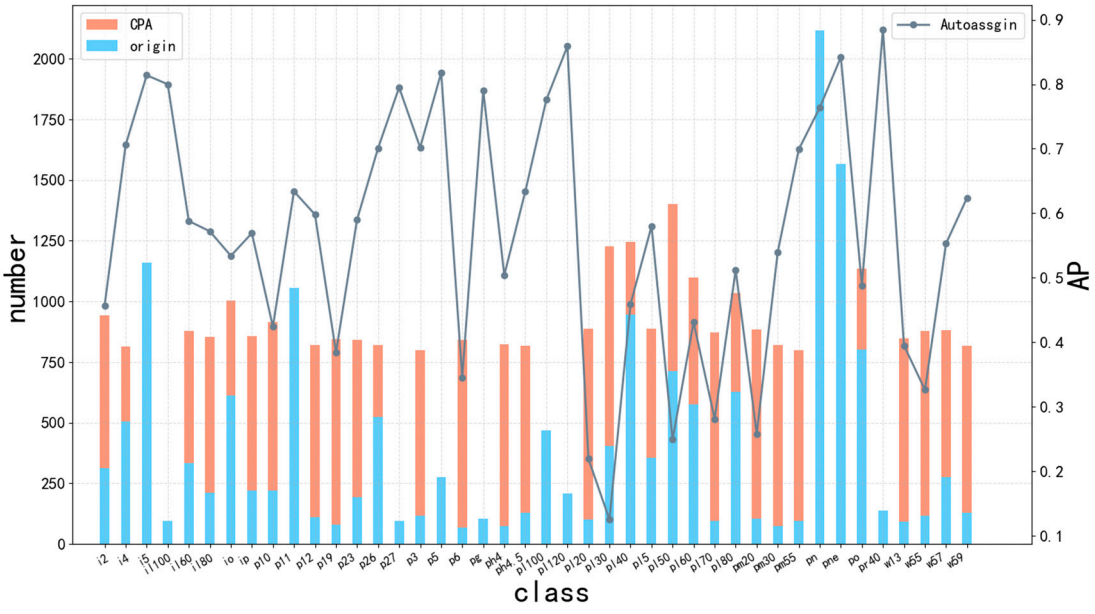


Figure 4. Updated training dataset by MSACP and corresponding evaluation results at n = 2.

Figures 5 and 6 show the data updates of the AutoAssign model when n = 3. For Figure 5, this is the evaluation when the model AutoAssign is trained to epoch = 4. There are 11 classes in the dataset that have not been expanded by MSACP. i4, i5, il100, il60, ip, p23, p27, pl100, and pr40 have not been amplified because these classes have higher AP.

Figure 5. The first updated training dataset by MSACP and corresponding evaluation results of AutoAssign at n = 3.

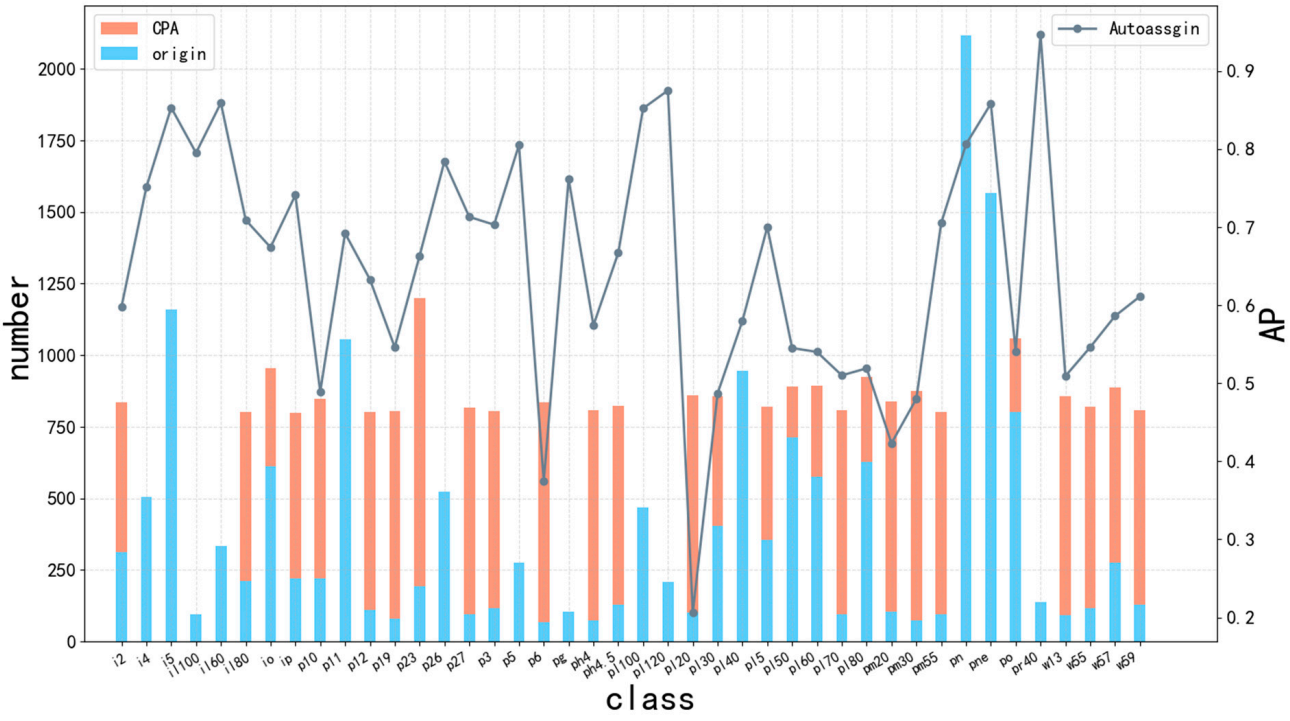


Figure 6. The second updated training dataset by MSACP and corresponding evaluation results of AutoAssign at n = 3.

Figure 6 shows the second data update of the AutoAssign model when n = 3. The result of AutoAssign is trained to epoch = 8. There are 14 classes in the figure that have not been expanded by MSACP. i4, i5, il100, il60, p26, p5, pg, pl100, p11, pl120, pl40, and pr4 do not have amplification information because these classes have higher APs. Compared with the first data update with n = 3, three more classes of information were amplified in this update, namely, p26, p5, and pl40. For example, the model does not pay much attention to category pl40 in early training, which leads to poor model detection of this class. The MSACP algorithm adjusts the model's attention to this class with information compensation. In the second data update, class pl40 does not need compensation for the information by MSACP because it has a higher AP. The class information is updated according to the different stages of training.

According to the data visualization of the MSACP experiment, the expanded sample quantity of each category is different in the three updates. By dividing the training amount into multiple stages, MSACP dynamically adjusts the class information of the training dataset and modifies the learning direction of the model at intervals according to the learning situation. This kind of dynamic adjustment at intervals not only reduces the computing power requirement of training, but also alleviates the class imbalance.

Six models are used to verify the effectiveness of the MSACP algorithm. There are three models that achieve better results when n = 4, namely, Grid RCNN, NAS FGN, and YOLOv3. The other three models achieve better results when n = 2 or n = 3. From the experimental results, it can be seen that more training stages do not necessarily make the model obtain a better improvement effect. When n is greater than four, these models are likely to have better results. The larger the n, the more time it takes to update the training dataset, which leads to a longer training time. When n = 1, the training of the model has only one stage and the training set is not updated. The entire training of the AutoAssign model takes approximately 270 min. When n is greater than one, the training dataset is updated by MSACP. For AutoAssign, it takes about 30 min to complete an update of the training dataset and start the next stage of training. When n = 4, the total running time of MSACP is about 120 min, as shown in Table 7. Although better detection performance may occur if the hyper-parameter n is set larger, there is not much room for improvement in the model. Under the comprehensive consideration of training time and performance improvement, the experiment only set four groups to test the impact of n on the MSACP algorithm.

Table 7. The time of AutoAssign model to update the training dataset by MSACP.

n	Time (min)	AP (%)
1	270	48.0
2	300	53.4
3	330	52.7
4	370	52.5

5. Embedded Application of MSACP Algorithm

Many deep learning algorithms have been applied in real life, but there are also many challenges in the practical application. How to improve the performance of the embedded platform under the limited computing power is one of the main challenges. Therefore, MSACP is applied to the autonomous driving scenario to test the effect of the algorithm in practical applications. The scenario is shown in Figure 7. It can be seen that the autonomous driving scenario has multiple speed limit signs and a traffic light. The embedded platform is in the red box, which is equipped with a CPU of 4-core ARM Cortex-57 series (ARM, Cambridge, UK) and a GPU of 128-core Maxwell architecture (NVIDIA, Santa Clara, CA, USA). The computing power is not less than 472 GFLOPs. In addition, the platform is equipped with a human–computer interaction interface, which can output the road condition information and detect the corresponding object information in real time.

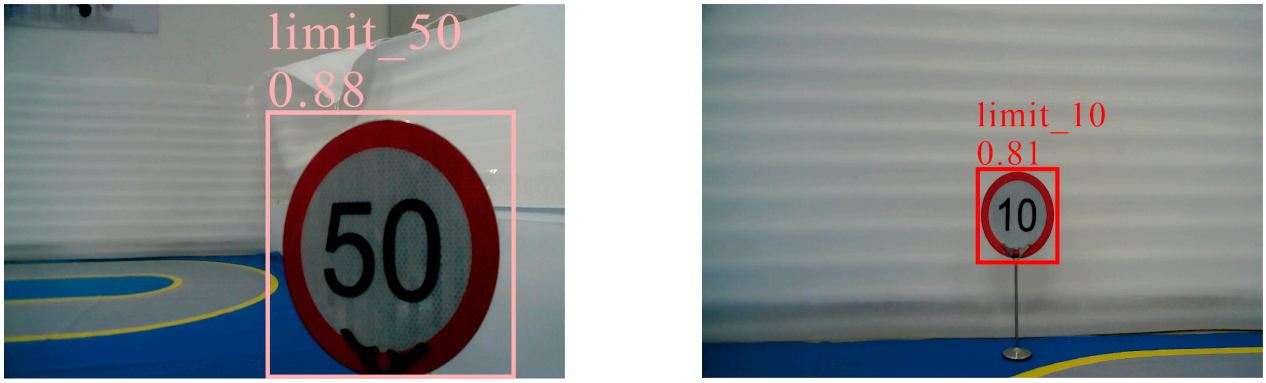
The model YOLOv3-tiny is selected for traffic sign detection in the embedded platform, which is a simplified version of YOLOv3. Yolov3-tiny mainly removes some feature layers based on YOLOv3, and at the same time changes the three independent prediction branches into two. Therefore, YOLOv3-tiny has a faster inference speed and is widely used in practical projects. The dataset used in this experiment contains 12 classes, some of which are difficult to collect in real life. There is still an imbalance of categories in the dataset. The class of people and car in the dataset has the largest number of samples, while the class with the smallest number of samples is yellow light. Because of the cost and time of production, MSACP is used to optimize the dataset, which is embedded into the YOLOv3-tiny model. The optimal weight file is obtained after the training is completed and deployed to the embedded platform for object detection. A comparative test is performed before importing the embedded platform, as shown in Table 8. The metric used in the table is AP_{50}. The YOLOv3-tiny model with the MSACP algorithm is improved by 3.3% compared to the original model. Some classes have significant improvements, such as limit_10 and limit_50.

Figure 7. The scene of simulated autonomous driving.

Table 8. The comparison of class performances based on the YOLOv3-tiny model.

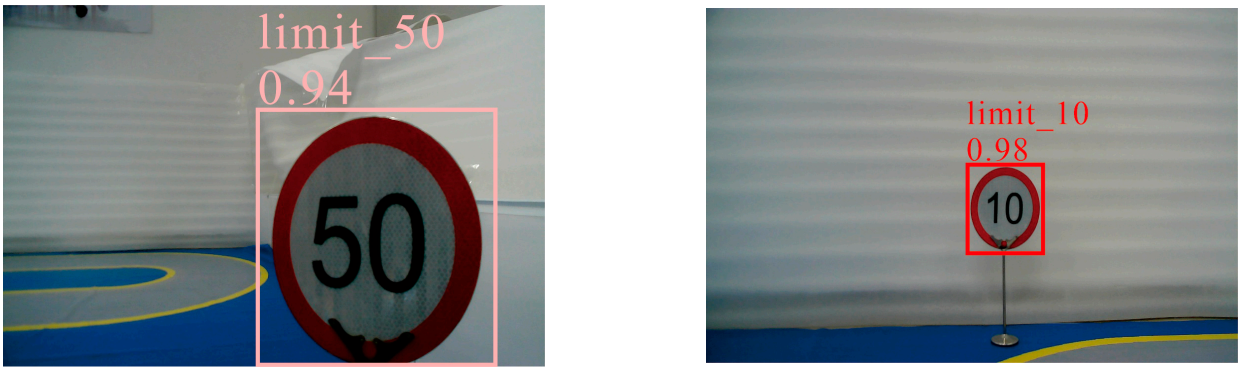
Class	YOLOv3-Tiny (%)	YOLOv3-Tiny + MSACP(%)
green_light	78.8	76.8 (−2.0)
yellow_light	97.0	98.4 (+1.4)
red_light	68.8	70.3 (+1.5)
turn_right	85.2	83.4 (−1.8)
turn_left	77.4	82.3 (+4.9)
limit_10	72.0	99.5 (+27.5)
limit_50	85.6	91.3 (+5.7)
limit_60	95.6	95.6 (+0.0)
limit_110	94.5	96.9 (+2.4)
person	99.3	99.4 (+0.1)
car	99.5	99.5 (+0.0)
stop	99.5	99.5 (+0.0)
Average	87.8	91.1 (+3.3)

In addition, a more intuitive experiment was performed. Three images with traffic signs on which the model had not been trained were randomly selected for testing. The results are shown in Figures 8 and 9. Figure 8 shows the detection results of YOLOv3-tiny. Figure 9 shows the detection results of YOLOv3-tiny + MSACP. From the visual results, the YOLOv3-tiny model with MSACP has higher confidence in detecting traffic signs. More importantly, the inference time is the same for both comparisons.



(**a**) (**b**)

Figure 8. The detection effect of YOLOv3-tiny: (**a**) The class of speed limit 50 km/h; (**b**) The class of speed limit 10 km/h.



(**a**) (**b**)

Figure 9. The detection effect of YOLOv3-tiny + MSACP: (**a**) The class of speed limit 50 km/h; (**b**) The class of speed limit 10 km/h.

6. Conclusions

In this paper, we propose MSACP, a multi-stage adaptive Copy-Paste augmentation method based on model training preference. MSACP introduces a multi-stage adjustment mechanism to provide dynamic information compensations for the training of the model. Under the premise of alleviating class imbalance, the over-fitting or under-fitting phenomenon generated in the training of the model is avoided as much as possible. Although the MSACP algorithm is effective according to the experimental results, it takes a long time to update the training dataset. When the MSACP algorithm processes complex data such as 3D images, the operation among objects easily destroys key information in the image, and even generates data that are not conducive to model training. In future work, this is one of the difficulties that we will focus on overcoming. In addition, we hope that the multi-stage adjustment mechanism in the MSACP algorithm can further be applied to different data enhancement strategies so that data augmentation can play an effective role in model training.

Author Contributions: Conceptualization, X.Y., F.L., Y.L. and A.W.; methodology, X.Y., A.W., Y.L. and F.L.; software, F.L.; validation F.L.; writing—review and editing X.Y., F.L., Y.L. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Major Science and Technology Projects of Zhongshan City in 2022 (2022A1020).

Data Availability Statement: https://cg.cs.tsinghua.edu.cn/traffic-sign/ (accessed on 1 January 2016).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Oksuz, K.; Cam, B.C.; Kalkan, S.; Akbas, E. Imbalance problems in object detection: A review. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 3388–3415. [CrossRef] [PubMed]
2. Zhang, Y.; Kang, B.; Hooi, B.; Yan, S.; Feng, J. Deep Long-Tailed Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 10795–10816. [CrossRef] [PubMed]
3. Feng, C.; Zhong, Y.; Huang, W. Exploring classification equilibrium in long-tailed object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3417–3426.
4. Wang, Y.; Gan, W.; Yang, J.; Wu, W.; Yan, J. Dynamic curriculum learning for imbalanced data classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5017–5026.
5. Shrivastava, A.; Gupta, A.; Girshick, R. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 761–769.
6. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
7. Chen, P.; Liu, S.; Zhao, H.; Jia, J. Gridmask data augmentation. *arXiv* **2020**, arXiv:2001.04086.
8. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
9. Huang, S.W.; Lin, C.T.; Chen, S.P.; Wu, Y.Y.; Hsu, P.H.; Lai, S.H. Auggan: Cross domain adaptation with gan-based data augmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 718–731.
10. Ghiasi, G.; Cui, Y.; Srinivas, A.; Qian, R.; Lin, T.-Y.; Cubuk, E.D.; Le, Q.V.; Zoph, B. Simple copy-paste is a strong data augmentation method for instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 2918–2928.
11. Dwibedi, D.; Misra, I.; Hebert, M. Cut, paste and learn: Surprisingly easy synthesis for instance detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1301–1310.
12. Cubuk, E.D.; Zoph, B.; Mane, D.; Vasudevan, V.; Le, Q.V. Autoaugment: Learning augmentation strategies from data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 113–123.
13. Lim, S.; Kim, I.; Kim, T.; Kim, C.; Kim, S. Fast AutoAugment. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 6665–6675.
14. Yu, X.; Li, F.; Bai, P.; Liu, Y.; Chen, Y. Copy-paste with self-adaptation: A self-adaptive adjustment method based on copy-paste augmentation. *IET Comput. Vis.* **2023**. [CrossRef]
15. Lu, X.; Li, B.; Yue, Y.; Li, Q.; Yan, J. Grid r-cnn. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7363–7372.
16. Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; Li, S.Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9759–9768.
17. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
18. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
19. Wang, T.; Zhu, Y.; Zhao, C.; Zeng, W.; Wang, J.; Tang, M. Adaptive class suppression loss for long-tail object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3103–3112.
20. Tan, J.; Wang, C.; Li, B.; Li, Q.; Ouyang, W.; Yin, C.; Yan, J. Equalization loss for long-tailed object recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11662–11671.
21. Menon, A.K.; Jayasumana, S.; Rawat, A.S.; Jain, H.; Veit, A.; Kumar, S. Long-tail learning via logit adjustment. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
22. Li, T.; Cao, P.; Yuan, Y.; Fan, L.; Yang, Y.; Feris, R.S.; Indyk, P.; Katabi, D. Targeted supervised contrastive learning for long-tailed recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 6918–6928.
23. Cui, Y.; Jia, M.; Lin, T.Y.; Song, Y.; Belongie, S. Class-balanced loss based on effective number of samples. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9268–9277.
24. Yun, S.; Han, D.; Oh, S.J.; Chun, S.; Choe, J.; Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6023–6032.
25. Kisantal, M.; Wojna, Z.; Murawski, J.; Naruniec, J.; Cho, K. Augmentation for small object detection. *CS IT Conf. Proc.* **2019**, *9*, 119–133.
26. Dvornik, N.; Mairal, J.; Schmid, C. Modeling visual context is key to augmenting object detection datasets. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 364–380.
27. Li, C.L.; Sohn, K.; Yoon, J.; Pfister, T. Cutpaste: Self-supervised learning for anomaly detection and localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9664–9674.

28. Georgakis, G.; Mousavian, A.; Berg, A.C.; Kosecka, J. Synthesizing training data for object detection in indoor scenes. *arXiv* **2017**, arXiv:1702.07836.
29. Yun, W.H.; Kim, T.; Lee, J.; Kim, J.; Kim, J. Cut-and-paste dataset generation for balancing domain gaps in object instance detection. *IEEE Access* **2021**, *9*, 14319–14329. [CrossRef]
30. Zhang, W.; Wang, Z.; Loy, C.C. Exploring Data Augmentation for Multi-Modality 3D Object Detection. In Proceedings of the International Conference on Learning Representations 2023 Workshop on Scene Representations for Autonomous Driving, Kigali, Rwanda, 1–5 May 2023.
31. Cubuk, E.D.; Zoph, B.; Shlens, J.; Le, Q.V. Randaugment: Practical automated data augmentation with a reduced search space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 702–703.
32. Lin, S.; Yu, T.; Feng, R.; Li, X.; Yu, X.; Xiao, L.; Chen, Z. Local patch autoaugment with multi-agent collaboration. *arXiv* **2021**, arXiv:2103.11099. [CrossRef]
33. Zhu, Z.; Liang, D.; Zhang, S.; Huang, X.; Li, B.; Hu, S. Traffic-sign detection and classification in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2110–2118.
34. Zhu, B.; Wang, J.; Jiang, Z.; Zong, F.; Liu, S.; Li, Z.; Sun, J. Autoassign: Differentiable label assignment for dense object detection. *arXiv* **2020**, arXiv:2007.03496.
35. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14. Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
36. Ghiasi, G.; Lin, T.Y.; Le, Q.V. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7036–7045.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Multi-Scale Spatial–Spectral Attention-Based Neural Architecture Search for Hyperspectral Image Classification

Yingluo Song [1], Aili Wang [1,*], Yan Zhao [2], Haibin Wu [1] and Yuji Iwahori [3]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; 2120610140@stu.hrbust.edu.cn (Y.S.); woo@hrbust.edu.cn (H.W.)
[2] Communication Construction Operation and Maintenance Center, State Grid Heilongjiang Electric Power Co., Ltd., Information and Communication Company, Harbin 150010, China; q_peach@126.com
[3] Department of Computer Science, Chubu University, Kasugai-shi 487-8501, Aichi, Japan; iwahori@isc.chubu.ac.jp
* Correspondence: aili925@hrbust.edu.cn

Abstract: Convolutional neural networks (CNNs) are indeed commonly employed for hyperspectral image classification. However, the architecture of cellular neural networks typically requires manual design and fine-tuning, which can be quite laborious. Fortunately, there have been recent advancements in the field of Neural Architecture Search (NAS) that enable the automatic design of networks. These NAS techniques have significantly improved the accuracy of HSI classification, pushing it to new levels. This article proposes a Multi-Scale Spatial–Spectral Attention-based NAS, MS^{3}ANAS) framework for HSI classification to automatically design a neural network structure for HSI classifiers. First, this paper constructs a multi-scale attention mechanism extended search space, which considers multi-scale filters to reduce parameters while maintaining large-scale receptive field and enhanced multi-scale spectral–spatial feature extraction to increase network sensitivity towards hyperspectral information. Then, we combined the slow–fast learning architecture update paradigm to optimize and iteratively update the architecture vector and effectively improve the model's generalization ability. Finally, we introduced the Lion optimizer to track only momentum and use symbol operations to calculate updates, thereby reducing memory overhead and effectively reducing training time. The proposed NAS method demonstrates impressive classification performance and effectively improves accuracy across three HSI datasets (University of Pavia, Xuzhou, and WHU-Hi-Hanchuan).

Keywords: hyperspectral image (HSI) classification; neural architecture search; differentiable architecture search (DARTS); multi-scale attention mechanism

1. Introduction

Hyperspectral remote sensing images (HSIs) capture abundant spatial–spectral information across numerous spectral bands, enabling effective differentiation of surface cover. Therefore, hyperspectral images are widely used in environmental science [1], mineral exploration [2], plant detection [3], military reconnaissance [4], and so on.

HSIs can extract feature vectors containing thousands of bands from each spatial pixel position. In practice, there are two phenomena in HSIs: (1) Different objects have similar spectral characteristics; (2) The same object at different positions has different spectral characteristics [5]. This is due to the influence of the imaging factors of atmosphere and light, which leads to the obvious spectral shift between different scenes.

HSI classification is based on pixel level. In the past decade, various traditional machine learning methods, including the K-nearest neighbor (KNN) algorithm, Support Vector Machine (SVM) [6], and others, have been widely employed for pixel-level HSI classification [7].

Compared with traditional classification algorithms, deep learning-based HSI classification methods have demonstrated their effectiveness in extracting robust features from HSIs, and they have better classification performance. HSI classification methods utilizing CNN can be classified into three categories based on the feature extraction approach employed: spectral CNN [8–10], spatial CNN [11,12], and spectral–spatial CNN [13–17]. However, many excellent deep classification network architectures need to be designed manually, such as ResNet [16], DenseNet [18], VGG [19], and GoogLeNet [20].

Designing an efficient network architecture for HSI classification is a challenging task that involves significant time, energy, and a large number of verification experiments, making it difficult to achieve manually. In addition, HSI data has obvious differences in the number of frequency bands, spectral range, and spatial resolution, so the network architectures applied to different HSI data classifications are also different [21]. It is necessary to design different network architectures for HSI classification, which requires a large amount of work to adjust parameters, resulting in a large consumption of time and resource costs. Therefore, a natural idea is to automate the neural network design process and minimize human assistance.

The need to enhance the efficiency of automatic neural network design technology is motivated by both the expense of computing resources and the burden of parameter adjustment [22]. NAS has been applied to many tasks and has achieved remarkable results, such as speech recognition [23], computer vision [24], and so on. The objective of Neural Architecture Search (NAS) is to automate the construction of high-performance neural network structures by selecting and combining various neural operations from predefined search spaces. Previous approaches to NAS have employed various methods, including reinforcement learning algorithm (RL) [25], evolutionary algorithm (EA) [26], and gradient-based methods, to perform architecture searches. The reinforcement learning algorithm regards NAS search as an agent behavior, and network construction is realized through different behaviors and rewards, which can evaluate the performance of the acquired network structure. Representation and optimization of agent strategy are two key contents of network structure search using reinforcement learning [25]. The evolutionary algorithm uses a genetic algorithm to realize selection, crossover, and compilation to initialize the search of individuals and cyberspace [27,28].

Among various NAS methods, DARTS and population-based NAS are among the most popular ones because they have unique advantages in dealing with many challenging tasks. DARTS mainly benefits from the advantage of relaxing the search space into a continuous and high search efficiency. The population-based NAS mainly benefits from the advantages of diverse candidate structures within the population and involves genetic operators to drive the search process. As the term itself indicates, the population-based NAS represents a candidate architecture for each individual. Collaboration between candidate architectures can eliminate poor and well-preserved competition and push the overall optimization. At present, most population-based NAS methods use genetic algorithms or genetic programming [25,29] to simulate natural evolution processes, which requires careful design of random crossover mutation operators. For example, the population-based NAS method [30] has designed 11 mutation operators to modify the attributes of the network, including changing the learning rate, resetting weights, inserting convolutions, etc. This study shows that designing different genetic operators has good feasibility, and neural networks can be developed and changed from an architectural perspective. Although the population-based NAS has good performance, many current methods mainly have the drawback of low computational efficiency.

Whether based on reinforcement learning or population algorithm, NAS realizes automation through resource-consuming search. To minimize resource usage, one-shot NAS techniques using hypernetworks have been devised [31,32]. DARTS, a one-shot NAS approach employing a discernible search strategy [32], incorporates weight sharing to integrate hypernetwork training and the search for the optimal candidate architecture. This integration effectively curtails computing resource wastage. At the same time, the gradient-

based method is used to optimize the over-parameterized hypernetwork. However, the greater number of weight parameters in DARTS compared to architecture parameters imposes constraints on architecture optimization, leading to performance crashes [31]. Essentially, this is because DARTS' candidate results lack diversity in gradient optimization.

However, these NAS methods have their own characteristics that DARTS jointly trains the hypernetwork and only searches for the optimal solution through the gradient, so they have defects in flexibility and stability [33]. Many population-based NASs mainly rely on random crossover/mutation search, which usually requires a lot of computational cost to evaluate the performance. Therefore, this work adopts the slow–fast learning paradigm architecture update process in DARTS, which integrates the idea of population-based NAS. In addition, it can benefit from the advantages of differentiated NAS while overcoming the shortcomings of the population-based NAS [34].

NAS technology can efficiently automate this process, not only to find the weight of specific image classification tasks but also to obtain the best network architecture. The incorporation of NAS offers an excellent solution for HSI data classification, relieving individuals from the burdensome task of network architecture design. Chen et al. pioneered the integration of the DARTS method into the HSI classification task, presenting their proposed approach as 3D-Auto-CNN with Cutout (CNAS) [35] and leveraging point-by-point convolution to reduce the spectral dimensions of HSI to several dozens. Subsequently, DARTS is utilized to search for a neural network architecture that is well-suited for the HSI dataset. This approach aims to minimize redundancy and repetition.

In recent years, the rise of attention mechanisms has brought new research directions to deep learning. Many researchers use different scales of attention mechanisms to extract the effective features of different objects in HSI data and classify them effectively. To reduce repetition, there have been several approaches proposed in the literature for HSI classification. Wang introduced the Squeeze and Excitation (SE) [36] module, which adaptively learns the weights of different spectral bands and adjacent pixels in HSI. This module helps capture relevant information for classification. Roy et al. [37] introduced a novel method named A2S2K-ResNet (Attention-based Adaptive Spectral Spatial Kernel Improved Residual Network). This approach integrates spectral attention to effectively capture discriminative features for HSI classification. These methods contribute to reducing redundancy and improving the efficiency of HSI classification tasks. However, in the field of HSI classification for NAS applications, there is limited search space utilizing multi-scale attention mechanisms. In addition, various samples in the HSI dataset exhibit long- tailed distributions, resulting in imbalanced HSI classification results. In light of the aforementioned approaches, we propose a novel method that utilizes a multi-scale search space combined with multi-scale spatial–spectral attention. This approach aims to enhance the extraction of significant spectral–spatial information while suppressing redundant information and noise. By incorporating this multi-scale attention mechanism, our method effectively improves the accuracy of classification while reducing computational complexity. Furthermore, it helps to minimize repetition and redundancy in the HSI classification process.

Building upon the aforementioned discussion, we present a novel NAS method that encompasses the automatic design of a search space for multi-scale attention mechanisms and a search strategy based on the slow–fast paradigm. The key contributions of this research can be summarized as follows:

1. To address the issue of redundancy, we have proposed a highly effective NAS classification framework called Multi-Scale Spatial–Spectral Attention-based NAS (MS^3ANAS). By carefully analyzing the characteristics of HSI, we have designed a multi-scale search space that incorporates rich spatial–spectral attention mechanisms. This search space consists of seven convolutional operators, each equipped with attention mechanisms of different scales. The search process can automatically learn to add attention modules to appropriate locations in the architecture to fully explore the spectral and spatial information of HSIs for classification.

2. The slow–fast learning paradigm was quoted for NAS, which further optimizes the overall architecture vector through pseudo gradient iteration, stabilizing the optimal search process, which improves the update and rate of convergence speed of the architecture for HSI classification.
3. To better train the HSI classification model, the Lion optimizer has been introduced, which not only effectively improves the accuracy of HSI data classification but also reduces memory overhead and training time because the algorithm only tracks momentum and uses symbol operations to calculate updates.

2. Materials and Methods

2.1. Search Framework

Figure 1 illustrates the proposed NAS framework for HSI classification in this paper. Initially, the hyperspectral image is divided into patches using a sliding window approach. These patches are then fed into the automatic architecture search model, which consists of both normal cells and reduction cells. Secondly, the NAS network includes supernet architecture search and final network optimization. We designed a multi-scale attention mechanism search operator for the search space (The different colored arrows represent different candidate operations), enhancing the ability to extract multi-scale spectral–spatial features from HSI data, thereby improving classification performance. Then, we utilized the slow–fast learning paradigm to effectively update the overall architecture vector, improving the efficiency of constructing hypergrid architecture units. Finally, we introduced the Lion optimizer to accelerate model convergence and alleviate model memory overhead.

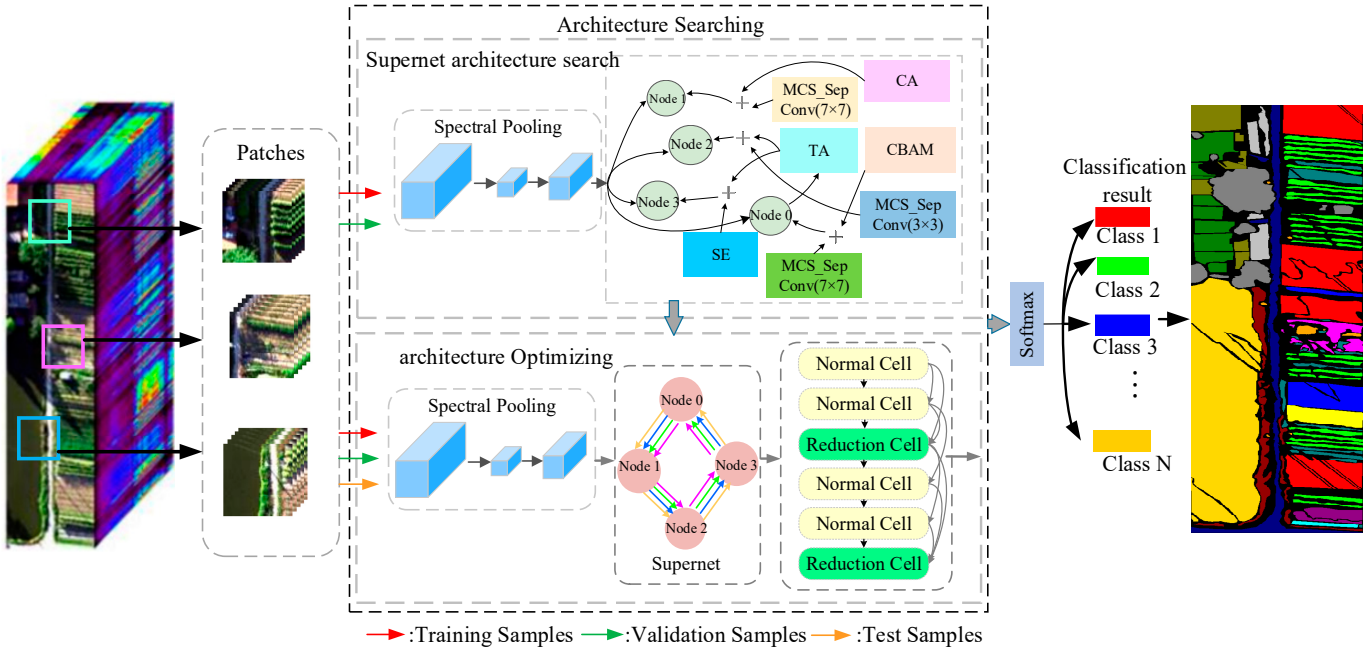


Figure 1. The overall framework of the proposed MS^{3}ANAS model.

2.2. Neural Architecture Search Algorithm

MS^{3}ANAS consists of three architecture search components: the search space of the multi-scale expanding attention mechanism, the search strategy of the slow–fast learning paradigm, and the Lion optimizer. Here is a detailed explanation of each component.

2.2.1. Multi-Scale Attention Mechanism Expanded Search Space

1. Multiple attention mechanism guided search space

Applying different attention mechanisms to different datasets may face varying challenges. For example, in some HIS datasets, it may be more challenging to use spatial

attention mechanisms due to strong correlations between data. In contrast, it may be more difficult to use channel attention mechanisms due to highly variable features. Moreover, the performance of attention mechanisms can also be affected by factors such as dataset size and quality, model architecture, and hyperparameters. Therefore, it is important to choose appropriate attention mechanisms based on different HSI datasets and tasks to achieve optimal performance and effectiveness. In addition, various samples in the HSI dataset exhibit long-tailed distributions, resulting in imbalanced HSI classification results. We propose a multiple search space with rich attention for HSI classification, which can effectively improve classification accuracy and reduce computational complexity. We select four types of attention mechanisms to form it. They are the convolutional block attention module (CBAM) [38], squeeze-and-excitation (SE) module [36], triplet attention (TA) module [39], and coordinate attention mechanism (CA) [40], as shown in Figure 2.

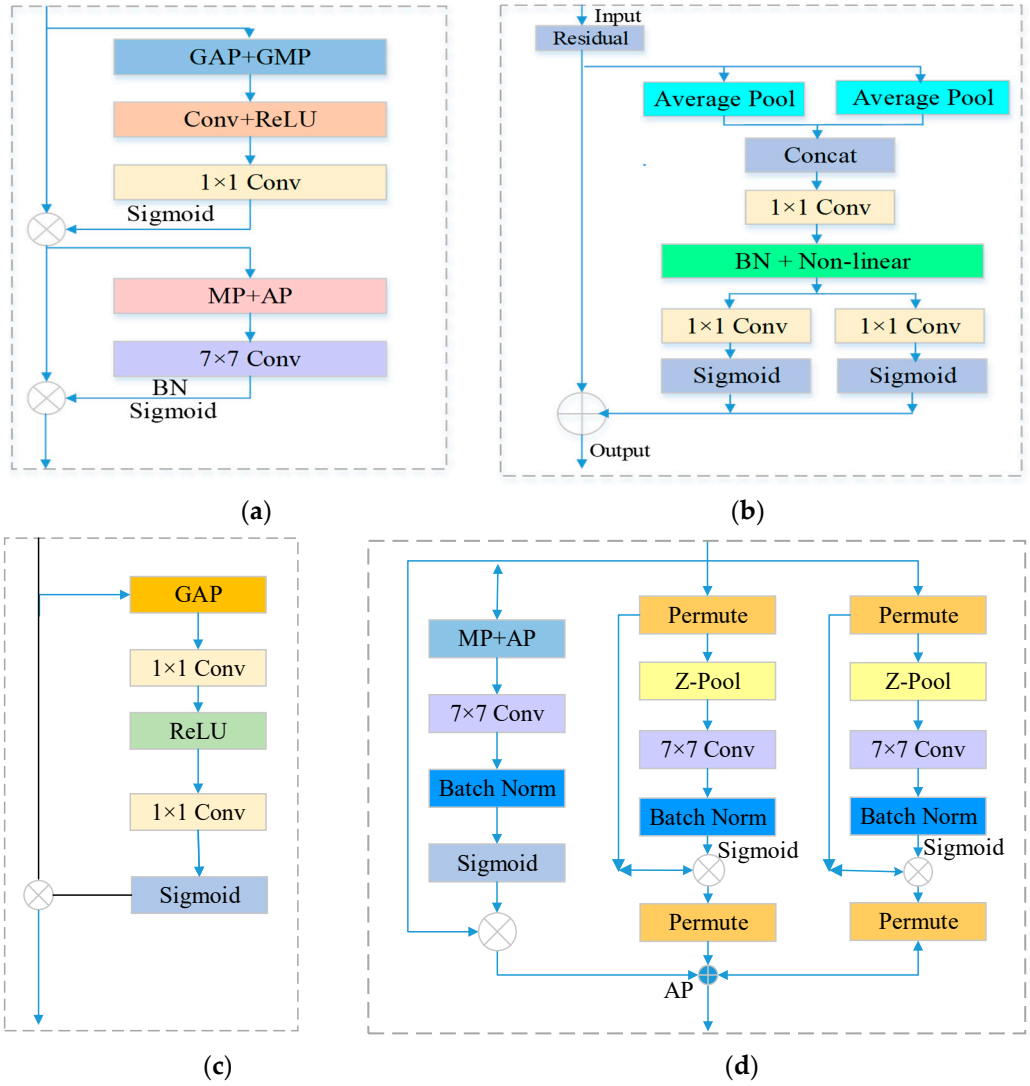


Figure 2. Multiple attention mechanism search space (**a**) CBAM; (**d**) CA; (**b**) SE; (**c**) TA.

To reduce repetition, various attention mechanisms have been proposed in the literature. One such mechanism is the CBAM module, which combines channel attention and spatial attention. It aggregates spatial features by performing max-pooling on top of global average pooling. This allows the network to focus on important spatial information. Another attention mechanism is the SE module, which enhances the network's sensitivity to informative features. It recalibrates filter responses through squeeze-and-excitation operations, improving the learning ability of convolutional layers. The TA module is

composed of three parallel branches, each serving a distinct purpose. The first branch focuses on establishing spatial attention, while the other two branches aim to capture cross-dimensional interactions between the channel and spatial dimensions. The final output is obtained by averaging the outputs from these three branches. The CA mechanism learns weights by combining features at each position with their corresponding coordinate information. This mechanism effectively captures spatial correlations, leading to improved model performance. By incorporating these attention mechanisms, models can effectively reduce redundancy and improve the performance of HSI classification tasks.

As NAS-based HSI classification requires the determination of the search space, we propose search space O. Formally, let O denote a sequence of candidate operations (multiple attention guided operations), where each operation represents a function $p(\cdot)$ to be applied to s_i. For each cell q, we configure an architecture parameter $\alpha_{p_i}^q$ for operation p_i. To ensure continuity of the search space, we relax the search space to allow it to be optimized via gradient descent. Specifically, we relax the architecture parameter $\alpha_{p_i}^q$ to be continuous, and then compute the operational probabilities for different operations by applying softmax over all $\alpha_{p_i}^q$.

$$R_{p_i}^q = \frac{\exp(\alpha_{p_i}^q)}{\sum_{j=1}^{n} \exp(\alpha_{p_j}^q)} \tag{1}$$

Here, n represents the number of candidate operations available. The larger value of $R_{p_i}^q$ indicates a higher likelihood of selecting the representative operation. The output of the cell is obtained by taking the weighted sum of all possible operations.

$$\overline{p}(y_i) = \sum_{p \in P} R_{p_i}^q p_i(y_i) \tag{2}$$

The notation $p_i(y_i)$ signifies the application of operation p_i on input y_i. Consequently, the search process is transformed into a learning process of a set of architectural parameters $\{\alpha_{p_i}^q\}$. Furthermore, as the network weights w also need to be learned, we are required to solve the following bi-level optimization problem.

$$\min_{\alpha} \mathcal{L}_{val}(w*, \alpha) \tag{3}$$

$$s.t.\ \mathrm{w}* = \mathrm{argmin}\mathcal{L}_{train}(w, \alpha) \tag{4}$$

The purpose of the above formulae is to search for the architectural parameters that minimize the validation loss $\mathcal{L}_{val}(w*, \alpha)$, and the network weights $w*$ are obtained by minimizing the training loss $\mathcal{L}_{train}(w, \alpha)$. It is important to note that the training loss and the validation loss are identical.

The objective of the aforementioned formulae is to search for architectural parameters that minimize the validation loss $\mathcal{L}_{val}(w*, \alpha)$, while the network weights $w*$ are obtained by minimizing the training loss $\mathcal{L}_{train}(w, \alpha)$. It is crucial to emphasize that the training loss and the validation loss are indeed the same.

2. Multi-scale attention mechanism search space

Different types of convolutions have varying computational requirements and parameter counts. Building deeper models can be challenging due to the expensive parameters and time required for high-dimensional convolutions. This can limit the efficiency and feasibility of constructing deeper models. Therefore, high-dimensional convolutions are often replaced with lower-dimensional separable convolutions to alleviate this issue. The novel operation MCS_sepConv_(b $\times$ b) (b = 3, 5, 7) in the search space combines spectral–spatial CBAM with spatial separable convolution [41], which helps to extract deeper spectral–spatial features, and it can enhance the spectral–spatial adaptive learning ability of the I data. In addition, we use small filters to reduce parameters while maintaining a large-scale receptive field. We have strengthened the ability to extract scale spectral–spatial features from data, thereby

improving classification performance. MCS_sepConv(b = 3, 5, 7) with different scales are as shown in Figure 3.

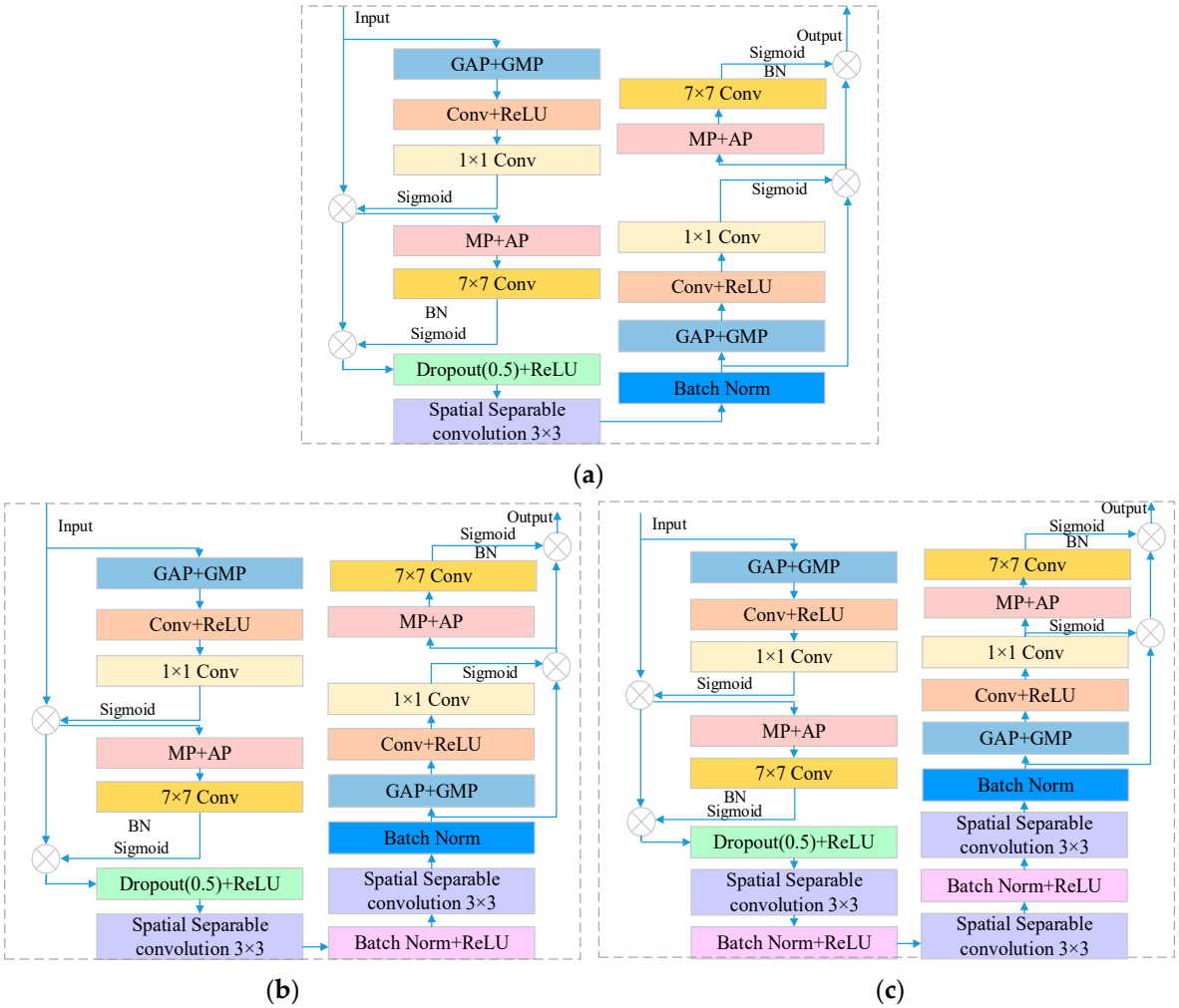


Figure 3. Multi-scale attention mechanism guided search space (**a**) MCS_sepConv(3 × 3); (**b**) MCS_sepConv(5 × 5); (**c**) MCS_sepConv(7 × 7).

MCS_sepConv_(b × b) refers to a set of convolutional operators that are commonly used in remote sensing and other image-processing tasks to extract deeper spectral–spatial features. These operators consist of two separate convolutional layers, one for processing the spectral dimension and the other for processing the spatial dimension, and they are applied in a cascaded manner. By using MCS_sepConv_(b × b), the model can capture more complex and abstract features by combining spectral and spatial information.

Convolution is a widely used operation in NAS-based methodIor HSI classification. Previous research has primarily concentrated on traditional convolution techniques, depthwise separable convolution, and dilated convolution in order to reduce redundancy. However, the high-dimensional convolutional operations can lead to a large number of parameters and time consumption, making it very difficult to construct an optimal architecture. For example, depthwise separable convolution can be represented as Sep-Conv(b × b) = Conv (1 × 1)(Conv(b × b)(x)), and the parameter and Flops(floating point operations per second) calculations are:

$$P_1 = C_{in} \times b \times b + 1 \times C_{in} \times C_{out} \tag{5}$$

$$F_1 = C_{in} \times b \times b \times H \times W + 1 \times 1 \times C_{in} \times C_{out} \times H \times W \tag{6}$$

where $C_{in} \times H \times W$ is the size of the input x, and $C_{out} \times H \times W$ corresponds to the output features, where C_{in} and C_{out} represent the input and output spectral bands, respectively. Additionally, P_1 and F_1 represent the number of parameters and Flops. Incorporating a pointwise convolution following a depthwise convolution in the CNN can effectively extract both spatial and spectral features in a sequential manner. Assuming $C_{in} = C_{out} = C$, the number of parameters and Flops for separable convolutions is only $(1/b^2 + 1/C)$ of regular convolutions. If we extend the concept of separable convolution to the spatial dimensions, we can define spatial separable convolution as Spatial_SepConv(b $\times$ b) = Conv ($1 \times$ 1Conv($1 \times$ b)(Conv(b $\times$ 1)(x)). The number of parameters and Flops for a spatial separable convolution is only $(2b + C)/(C + b^2)$ of separable convolution:

$$P_4 = 2 \times (C_{in} \times b \times 1) + 1 \times 1 \times C_{in} \times C_{out} \tag{7}$$

$$\begin{aligned} F_4 = {} & 2 \times (C_{in} \times b \times 1 \times H \times W) \\ & +1 \times 1 \times C_{in} \times C_{out} \times H \times W \end{aligned} \tag{8}$$

To reduce redundancy in a convolutional neural network (CNN), one approach is to use small filters instead of large-scale filters. This helps to decrease the number of parameters while still maintaining a large receptive field. This is because, in the CNN, the number of parameters and computations is directly related to the size of the input data and the convolution kernel. By using smaller filters, the number of parameters and computations can be significantly reduced. By using small filters, we can reduce the number of parameters while maintaining a large receptive field, which is important for capturing multi-scale features in HSI classification. This can be achieved by using a combination of small filters with different kernel sizes, which allows us to capture multi-scale features while reducing the number of parameters and computations.

To enhance the adaptive feature extraction capability of the designed convolution operation, a lightweight attention module called CBAM (Multi-scale Channel–Spatial Attention, MCS) is incorporated to enhance the spectral and spatial adaptive learning ability of the data cube. MCS is a combination of the Multi-scale Channel Attention (MS) mechanism and the Multi-scale Spatial Attention (MS) mechanism as follows:

$$\begin{aligned} Mc(F) &= \sigma(MLP(AvgPool(F) + MLP(MaxPool(F)) \\ &= (\mathrm{W}_1(\mathrm{W}_0(\mathrm{F}^C_{avg})) + \mathrm{W}_1(\mathrm{W}_0(\mathrm{F}^C_{max}\Big))) \\ F' &= M_c(F) \otimes F \end{aligned} \tag{9}$$

$$\begin{aligned} Ms(F) &= \sigma(f^{7\times7}([AvgPool(F'); MaxPool(F')])) \\ &= (f^{7\times7}[F\prime^S_{avg}; F\prime^S_{max}]) \\ F'' &= M_s(F') \otimes F' \end{aligned} \tag{10}$$

where $M_{cs}(F) = M_s(M_c(F))$ represents the spectral–spatial CBAM, which involves element-wise multiplication $\otimes$, $\mathrm{W}_0 \in \mathbb{R}^{C_{in}/\mathrm{r}\times C_{in}}$,$\mathrm{W}_1 \in \mathbb{R}^{C_{in}\times C_{in}/r}$, sigmoid activation σ, and the 7×7 convolutional $f^{7\times7}$.

As shown in Table 1, Conv(5×5) and Conv(7×7) can be achieved by repeating the Conv(3×3) operations for substitution. In addition, Conv(3×3) can be replaced with Spatial sepconv(3×3) = (Conv(1×3)Conv(3×1)). Assuming $C_{in} = C_{out} = C$, when using Spatial sepconv(3×3) for equivalent replacement of Conv(5×5), the parameter count is reduced by about 25%. After replacing Conv(7×7), the parameters were reduced by about 33%. The amount of Flops also showed a significant decrease. It can be explained that we use small filters instead of large ones to effectively reduce parameters while maintaining a large receptive field, which has obvious effectiveness.

Table 1. Comparison of convolution parameters and Flops at different scales.

Operation	Convolution	Repeat	Parameters	Flops
Conv(3 × 3)	Conv(3 × 3)	-	$9 \times C^2$	$9 \times C^2 \times H \times W$
Conv(5 × 5)	Conv(3 × 3)	2	$18 \times C^2$	$18 \times C^2 \times H \times W$
Conv(7 × 7)	Conv(3 × 3)	3	$27 \times C^2$	$27 \times C^2 \times H \times W$
Spatial_sepconv(3 × 3)	Conv(1 × 3)(Conv(3 × 1))	-	$6 \times C^2$	$6 \times C \times H \times W$
Spatial_sepconv(5 × 5)	Conv(1 × 3)(Conv(3 × 1))	2	$12 \times C^2$	$12 \times C \times H \times W$
Spatial_sepconv(7 × 7)	Conv(1 × 3)(Conv(3 × 1))	3	$18 \times C^2$	$18 \times C \times H \times W$

2.2.2. Search Strategy Based on the Slow–Fast Learning Paradigm

In order to further utilize the search space guided by multiple attention mechanisms, we apply the slow–fast learning paradigm to optimize and iteratively update the architecture vector. In addition, the HSI classification involves multiple categories, and the sample size in each category presents the long-tailed distribution, resulting in imbalanced classification results for certain categories in classification tasks. By using the slow–fast paradigm to update the architecture vector, the overall architecture vector can be updated from the perspective of pseudo gradients [42]. Different hyperspectral imbalanced data can be updated during the construction of the architecture unit to obtain the best results and effectively enhance the generalization of the model. This design is essentially aimed at designing an effective NAS method that benefits from the high efficiency of differentiable NAS and overcomes the drawbacks of high search costs in population-based NAS.

This section optimizes from the perspective of NAS search strategy and effectively updates the population of architecture vectors through pseudo gradient iteration to achieve optimal results. The objective of NAS is to initially search for architecture vectors $\alpha* \in O$. Then, the architecture parameter $C_{\alpha*}$ is used to minimize the validation loss $\mathcal{L}_{val}(C_{\alpha*}, w_{\alpha*})$, and finally, the weight $w*$, which is associated with the architecture, is obtained by minimizing the training loss $\mathcal{L}_{train}(C_{\alpha*}, w_{\alpha*})$. Therefore, the slow–fast learning is essentially the optimizer for α parameters in our NAS. Specifically, we give the architecture vector α^t obtained by slow–fast learning in the t^{th} generation, and its iterative update method is shown in Figure 4.

$$\alpha^{t+1} = \alpha^t + \Delta\alpha^t \tag{11}$$

where the above equation satisfies $\mathcal{L}_{val}(C_{\alpha^{g+1}}, w^*_{\alpha^{g+1}}) < \mathcal{L}_{val}(C_{\alpha^g}, w^*_{\alpha^g})$.

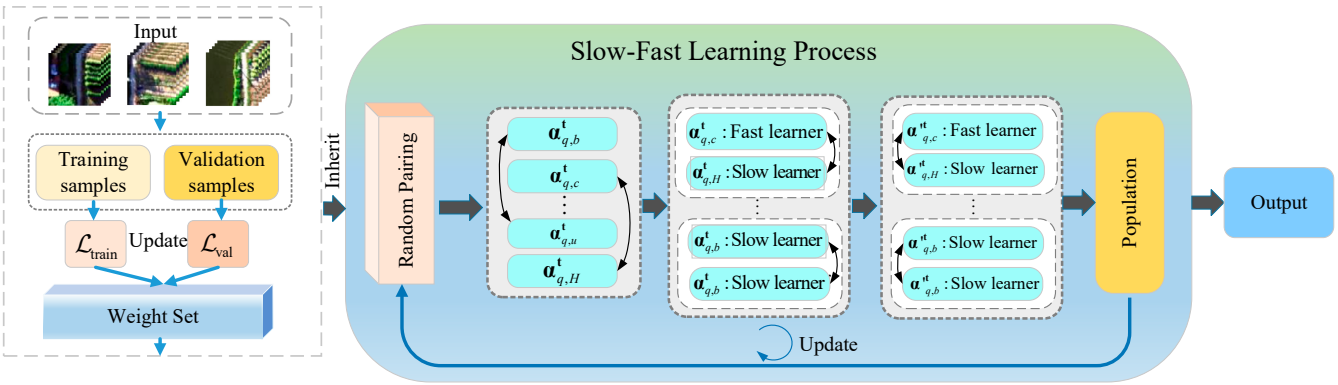


Figure 4. The slow–fast learning process.

To generate pseudo gradient $\Delta\alpha^t$ effectively, this approach suggests utilizing a population of H architecture vectors $\{\alpha\}_{h=1}^{H}$. Each generation randomly splits the population into H/2 pairs. Next, for each pair of population q, the order of validation loss values is employed to identify the fast learner $\alpha^t_{q,f}$ and slow learner $\alpha^t_{q,s}$, with smaller loss values

indicating slow learner and larger loss values indicating fast learner [$\mathcal{L}_{val}(C_{\alpha^t_{q,f}}, w^*_{\alpha^t_{q,f}}) < \mathcal{L}_{val}(C_{\alpha^t_{q,s}}, w^*_{\alpha^t_{q,s}})$]. Then, $\alpha^t_{q,s}$ learns from $\alpha^t_{q,f}$ and updates it.

$$\Delta\alpha^t_{q,s} = \eta_1(\alpha^t_{q,f} - \alpha^t_{q,s}) + \eta_2\Delta\alpha^{t-1}_{q,s} \tag{12}$$

Here, $\eta_1, \eta_2 \in [0, 1]$ represents a randomly generated value obtained from a uniform distribution. Specifically, η_1 determines the step size for $\alpha^t_{q,s}$ to learn from $\alpha^t_{q,f}$, and η_2 determines the influence of momentum $\Delta\alpha^{t-1}_{q,s}$. Finally, all fast–slow learners are aggregated to form a new group of the t + 1th generation. The above pseudo gradient update method is derived from the second derivative of gradient descent in back propagation, so each architecture vector in the search space of the multi-attention mechanism designed in this work will move towards the optimal update direction from the vector that converges faster than them.

After the architecture update process, it is necessary to evaluate the performance of candidate architectures for decoding architecture vectors, so that for each pair of architecture vectors, fast learners can learn and update slow learners by verifying the loss. Subsequently, the candidate architecture $C_{\alpha g}$ is assessed by solving the following optimization problems:

$$w^*_{\alpha g} = \text{optimize}(w_{\alpha g}) = step(step(\ldots step(w_{\alpha g}|C_{\alpha g})\ldots|C_{\alpha g})|C_{\alpha g}) \tag{13}$$

In the given equation, w^*_{ag} represents the optimal weight of the candidate architecture, while $step(\cdot)$ denotes the iterative optimization process utilized for updating the weights of the neural network.

2.2.3. Lion Optimizer

The optimizer is utilized to update and compute the network parameters that influence model training and output, with the goal of approaching or achieving the optimal value and thereby minimizing (or maximizing) the loss function. This study utilizes the Lion optimizer, known for its simplicity, efficiency, and speed. Reducing redundancy is an important aspect of this work. At the same time, this optimizer is suitable for large-scale optimization problems involving big datasets or high-dimensional parameter spaces. The use of the Lion optimizer can not only effectively improve the accuracy of HSI data classification but also reduce memory overhead and effectively reduce training time because the algorithm only tracks momentum and uses symbolic operation to calculate updates. The Lion optimizer formula is written as follows [43]:

$$\theta_{t+1} = \omega_t\theta_t \tag{14}$$

$$g_t = \nabla_\theta f(\theta_{t-1}) \tag{15}$$

where ω_t is the decay rate, θ_t is the weights, and $g_t = \nabla_\theta f(\theta_{t-1})$ is the gradient at θ_{t-1}.

To calculate the average attenuation of the current and past square gradient, the mathematical relationship can be written as follows:

$$c_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t \tag{16}$$

where c_t is the 1st moment estimate, and m_{t-1} represents the momentum vector from the previous iteration. β_1 is the decay rate of the 1st moment. The weight reduction process for decoupling is as follows:

$$\theta_t \leftarrow \theta_{t-1} - \eta_t(sign(ct) + \lambda\theta_{t-1}) \tag{17}$$

where η_t is the step size. To counteract biases, the bias-corrected first and second moments are computed in the following manner:

$$m_t \leftarrow \beta_2 m_{t-1} + (1 - \beta_2) g_t \tag{18}$$

The algorithm differs from various adaptive algorithms in that it only tracks momentum and uses symbolic operations to compute updates, leading to lower memory overhead and achieving a unified update magnitude across all dimensions.

3. Results

3.1. Hyperspectral Image Datasets

To assess the efficacy of the proposed NAS method, this research conducts classification experiments on three datasets: one standard hyperspectral dataset (Pavia University) and two practical datasets (Xuzhou and WHU-Hi-Hanchuan). Tables 2–4 show the number of pixels in each category, the false color composite image, and the ground truth map of the three datasets, respectively. The aim is to reduce redundancy in the experiments and ensure a comprehensive evaluation. The Pavia University dataset was gathered using AROSIS sensors that flew over Pavia in northern Italy. It consists of 103 spectral bands and has a dataset size of 610 × 340 pixels. It consists of nine categories. The Xuzhou dataset was collected in November 2014 using the HySpex SWIR-384 and HySpex VNIR-1600 imaging spectrometers. The dataset consists of 436 bands and covers a specific pixel size in the mining area. A field survey was conducted, and a total of nine categories were calibrated. In total, 68,877 marker samples were used.

Table 2. Pavia University Dataset Labeled Sample Counts.

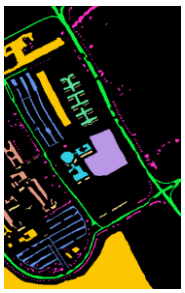
No.	Class	Color	Sample Numbers	False Color Map	Ground Truth Map
1	Asphalt		6631		
2	Meadows		18,649		
3	Gravel		2099		
4	Trees		3064		
5	Painted metal sheets		1345		
6	Bare Soil		5029		
7	Bitumen		1330		
8	Self-Blocking Bricks		3682		
9	Shadows		947		
	Total		42,776		

Table 3. Xuzhou Dataset Labeled Sample Counts.

No.	Class	Color	Sample Numbers	False Color Map	Ground Truth Map
1	Bareland-1		26,396		
2	Lakes		4027		
3	Coals		2783		
4	Cement		5214		
5	Crops-1		13,184		
6	Trees		2436		
7	Bareland-2		6990		
8	Crops-2		4777		
9	Red-tiles		3070		
	Total		68,877		

Table 4. WHU-Hi-Hanchuan Dataset Labeled Sample Counts.

No.	Class	Color	Sample Numbers	False Color Map	Ground Truth Map
1	Strawberry		44,735		
2	Cowpea		22,753		
3	Soybean		10,287		
4	Sorghum		5353		
5	Water spinach		1200		
6	Watermelon		4533		
7	Greens		5903		
8	Trees		17,978		
9	Grass		44,735		
10	Red roof		9469		
11	Gray roof		10,516		
12	Plastic		16,911		
13	Bare soil		3679		
14	Road		9116		
15	Bright object		18,560		
16	Water		1136		
	Total		0		

The WHU-Hi-Hanchuan dataset was collected in Hanchuan, Hubei Province, using the 17 mm focus Headwall Nano Hyperspec sensor installed on the Leica Airbot X6 UAV V1 platform. The size of the dataset is 1217 × 303 pixels, with 16 categories. The survey collected data from 68,877 marker samples.

3.2. Implementation Details

Our experiments are conducted at Intel (R) Xeon (R) 4208 CPU@2.10GHz Processor and Nvidia GeForce RTX 2080Ti graphics card. We conducted 10 experiments to take the average value to obtain the overall accuracy (OA), average accuracy (AA), and Kappa coefficient (K) of the experiment.

The hyperspectral image samples are extracted using the sliding window strategy with a window size and overlap rate set at 50%. For training purposes, 30 samples are randomly selected as the training dataset, while 20 samples are used for validation. The training dataset is utilized to train the weights and biases of each neuron in the model, whereas the architecture variables are optimized based on the validation dataset. Once the optimal architecture is obtained, the remaining samples are employed as the test dataset to evaluate the performance of the optimized network architecture and derive the final classification results.

To evaluate the discovered architectures, we conducted evaluations on the discovered architectures with two cells and sixteen initial channels. Each cell consisted of seven nodes, including two input nodes, four intermediate nodes, and one output node. The population size (N) and generation number were set to 1, 20, 35, and 50, respectively. The batch size for each search stage was 32, and we used Stochastic Gradient Descent (SGD) to optimize the weight parameters with a learning rate of 0.025. The cosine power annealing strategy was employed to reduce the learning rate, with a maximum learning rate of 0.025, a minimum learning rate of 0.0001, and a power curve parameter of 2. To prevent overfitting and ensure the robustness of the final model, we also used label smoothing regularization with a smoothing factor of 0.1.

3.3. Classification Comparison with State-of-the-Art Methods

In this section, we evaluate the classification performance of MS^{3}ANAS by comparing it with several advanced methods. These methods include Extended Morphological Profile combined with Support Vector Machine (EMP-SVM) [44], 2D-CNN [11], 3D-CNN [14], Spectral–Spatial Residual Network (SSRN) [16], Residual Network (ResNet) [45], Multi-Layer Perceptron Mixer (MLP Mixer) [46], CNN model designed using NAS (CNAS) [34], and Efficient Convolutional Neural Architecture Search for LiDAR DSM Classification (AN-AS-CPA-LS) [47]. To ensure the rigor of the experiment, we randomly selected three training samples for classes with fewer than three samples. The training times for manually designed CNN models were set to 200. For CNAS and ANAS-CPA-LS, they searched for the optimal architecture based on the DARTS strategy. Therefore, their configuration and superparameter settings were the same as those of the MS^{3}ANAS model. All experimental results are shown in Tables 5–7, and they represent the average values of ten runs with different random initializations.

Table 5. Classification results of all methods on the Pavia University dataset.

Methods	EMP-SVM	2D-CNN	3D-CNN	SSRN	MLP-Mixer	CNAS	ANAS-CPA-LS	MS^{3}ANAS
1	60.77 ± 9.11	76.52 ± 7.32	86.51 ± 3.50	94.01 ± 0.72	96.17 ± 1.07	96.45 ± 0.13	97.70 ± 0.70	99.01 ± 0.35
2	95.68 ± 1.10	94.47 ± 3.35	94.32 ± 3.69	99.40 ± 0.22	97.80 ± 0.81	98.96 ± 0.02	99.46 ± 0.42	99.79 ± 0.14
3	88.06 ± 6.64	93.23 ± 0.13	69.58 ± 5.82	98.26 ± 3.28	94.43 ± 3.16	98.17 ± 0.11	91.13 ± 6.27	97.09 ± 0.11
4	98.38 ± 1.55	99.52 ± 0.15	95.97 ± 3.02	98.73 ± 1.32	99.31 ± 1.00	99.70 ± 0.04	99.66 ± 0.16	99.89 ± 0.01
5	99.46 ± 0.36	99.36 ± 0.63	99.58 ± 0.11	99.64 ± 0.69	99.78 ± 0.33	99.42 ± 0.04	100 ± 0.00	99.87 ± 0.12
6	92.70 ± 1.16	81.95 ± 12.16	94.80 ± 1.00	95.28 ± 0.19	99.27 ± 4.13	99.00 ± 0.21	99.80 ± 0.11	99.86 ± 0.07
7	99.77 ± 0.22	42.66 ± 9.32	95.07 ± 1.29	96.35 ± 0.35	98.47 ± 0.53	99.41 ± 0.13	99.16 ± 0.83	99.38 ± 0.36
8	65.37 ± 6.31	71.28 ± 5.13	84.52 ± 6.71	84.58 ± 0.03	81.86 ± 2.72	86.12 ± 0.25	95.13 ± 0.68	95.33 ± 1.33
9	49.30 ± 8.33	96.79 ± 2.22	99.77 ± 0.10	99.74 ± 0.16	99.93 ± 1.46	97.47 ± 1.33	99.65 ± 0.34	100 ± 0.00
OA/%	83.39 ± 0.85	87.36 ± 1.67	91.04 ± 1.70	94.68 ± 1.32	96.16 ± 0.71	97.42 ± 0.29	98.44 ± 0.57	99.16 ± 0.03
AA/%	83.27 ± 3.86	89.14 ± 7.20	91.12 ± 1.11	96.22 ± 1.58	96.32 ± 0.90	97.18 ± 0.25	97.97 ± 0.72	98.91 ± 1.14
100 K	81.70 ± 1.44	82.99 ± 2.46	87.99 ± 2.35	92.94 ± 0.41	94.89 ± 0.09	96.76 ± 0.18	97.93 ± 0.76	98.89 ± 0.09
Params/M	0.0032	0.1439	0.0934	0.0894	0.4305	0.2164	0.2432	0.1376
Flops/G	0.0008	0.1274	0.4682	0.4843	0.0248	2.8302	2.7284	0.0194

Table 6. Classification results of all methods on the Xuzhou dataset.

Methods	EMP-SVM	2D-CNN	3D-CNN	SSRN	MLP-Mixer	CNAS	ANAS-CPA-LS	MS^{3}ANAS
1	95.60 ± 1.46	85.83 ± 1.83	94.71 ± 0.80	95.55 ± 1.92	96.33 ± 0.06	98.86 ± 0.66	99.15 ± 0.65	99.71 ± 0.13
2	92.32 ± 0.04	90.94 ± 4.72	85.10 ± 1.34	94.87 ± 4.97	99.94 ± 0.61	99.06 ± 0.79	99.50 ± 0.49	99.94 ± 0.03
3	85.76 ± 0.82	82.38 ± 1.81	92.21 ± 0.06	89.04 ± 6.99	93.98 ± 0.44	95.09 ± 2.71	99.40 ± 0.03	99.99 ± 0.01
4	98.45 ± 1.57	93.67 ± 4.81	96.86 ± 0.37	95.68 ± 0.28	95.02 ± 3.02	92.82 ± 5.03	99.10 ± 0.16	99.91 ± 0.04
5	88.00 ± 0.09	97.30 ± 0.82	93.33 ± 1.74	92.88 ± 0.37	96.20 ± 0.93	98.57 ± 0.15	95.95 ± 0.01	94.21 ± 0.62
6	82.41 ± 0.24	83.66 ± 2.10	86.41 ± 1.36	95.28 ± 1.26	79.38 ± 6.11	91.75 ± 4.89	97.56 ± 0.24	97.55 ± 0.45
7	72.46 ± 3.52	90.16 ± 3.11	86.47 ± 2.43	90.08 ± 1.23	92.75 ± 3.05	96.40 ± 1.41	94.83 ± 0.15	99.86 ± 0.98
8	59.38 ± 2.77	96.93 ± 1.80	98.81 ± 1.04	95.95 ± 0.45	89.20 ± 8.11	99.60 ± 0.24	97.07 ± 2.06	99.72 ± 0.07
9	92.16 ± 1.26	89.71 ± 0.32	89.72 ± 0.35	89.77 ± 2.49	98.27 ± 0.32	98.51 ± 0.58	96.92 ± 0.07	99.92 ± 0.01
OA/%	84.50 ± 0.81	88.32 ± 0.27	91.71 ± 0.93	92.41 ± 1.34	94.54 ± 2.32	97.07 ± 0.06	97.72 ± 0.19	98.44 ± 1.67
AA/%	85.17 ± 1.30	87.84 ± 0.72	91.51 ± 1.84	92.90 ± 2.21	93.45 ± 2.51	96.74 ± 1.82	97.67 ± 0.49	98.98 ± 0.25
100 K	80.38 ± 0.77	84.87 ± 0.43	89.46 ± 1.31	90.37 ± 1.66	93.14 ± 2.89	97.04 ± 1.01	97.11 ± 0.23	98.04 ± 2.10
Params/M	0.0193	0.0627	0.1094	0.2761	0.3976	0.3844	0.2452	0.0598
Flops/G	0.0059	0.1354	0.4717	0.4489	0.0643	3.0262	2.6810	0.0198

Table 7. Classification results of all methods on the WHU-Hi-Hanchuan dataset.

Methods	EMP-SVM	2D-CNN	3D-CNN	SSRN	MLP-Mixer	CNAS	ANAS-CPA-LS	MS^{3}ANAS
1	85.45 ± 2.47	88.15 ± 0.32	87.16 ± 0.19	90.11 ± 1.52	91.62 ± 2.63	93.71 ± 0.13	95.93 ± 0.68	97.99 ± 0.31
2	92.05 ± 2.01	87.90 ± 3.17	93.69 ± 0.01	94.31 ± 2.52	95.14 ± 0.01	96.32 ± 0.25	98.23 ± 0.23	98.68 ± 0.14
3	80.76 ± 3.53	82.11 ± 0.05	87.66 ± 1.09	90.52 ± 1.77	90.47 ± 0.32	94.41 ± 1.22	96.09 ± 2.92	98.80 ± 0.14
4	93.03 ± 2.77	86.44 ± 1.56	92.53 ± 1.23	97.82 ± 0.16	97.47 ± 0.70	97.97 ± 0.49	99.44 ± 0.04	99.32 ± 0.12
5	77.85 ± 16.59	52.13 ± 3.19	61.44 ± 4.24	81.64 ± 5.60	78.18 ± 3.02	88.30 ± 4.50	89.75 ± 2.39	96.02 ± 1.44
6	64.54 ± 7.93	65.71 ± 1.61	75.41 ± 1.09	74.34 ± 1.91	84.51 ± 0.05	84.48 ± 1.65	91.86 ± 0.68	94.00 ± 0.01
7	65.95 ± 1.24	69.80 ± 6.17	74.36 ± 1.63	82.13 ± 0.78	82.35 ± 3.44	84.44 ± 5.14	92.51 ± 2.64	93.79 ± 2.94
8	88.62 ± 0.49	83.48 ± 0.82	90.43 ± 2.19	92.07 ± 1.12	91.39 ± 1.05	95.86 ± 0.71	98.35 ± 0.02	98.15 ± 0.13
9	75.43 ± 5.81	78.55 ± 1.34	87.84 ± 1.25	87.51 ± 2.47	88.69 ± 3.05	90.85 ± 2.13	94.88 ± 0.87	97.35 ± 0.38
10	90.48 ± 3.12	93.35 ± 1.26	95.38 ± 1.69	95.04 ± 0.44	95.66 ± 2.72	97.38 ± 1.22	99.22 ± 0.04	99.73 ± 0.19
11	85.60 ± 3.01	82.62 ± 0.85	89.71 ± 1.26	86.97 ± 2.67	91.47 ± 1.37	91.39 ± 0.02	96.60 ± 0.01	97.53 ± 0.07
12	73.08 ± 2.90	74.78 ± 1.89	75.03 ± 4.91	80.84 ± 0.69	86.70 ± 0.89	89.59 ± 3.92	93.69 ± 0.19	94.75 ± 0.33
13	67.30 ± 1.71	67.97 ± 2.68	81.14 ± 1.10	88.13 ± 2.92	85.28 ± 1.08	87.45 ± 5.35	94.50 ± 1.87	95.55 ± 0.26
14	85.91 ± 0.73	84.59 ± 2.78	89.09 ± 0.13	92.72 ± 1.08	89.45 ± 0.85	94.22 ± 2.02	97.32 ± 0.23	97.47 ± 0.93
15	88.87 ± 6.20	87.60 ± 2.66	89.78 ± 4.21	84.40 ± 11.6	90.09 ± 7.58	92.99 ± 0.15	97.08 ± 0.56	95.82 ± 0.71
16	97.72 ± 3.02	97.48 ± 0.71	97.79 ± 0.26	99.15 ± 0.16	99.11 ± 0.01	99.49 ± 1.37	99.66 ± 0.14	99.77 ± 0.06
OA/%	87.96 ± 0.30	89.07 ± 0.31	91.01 ± 0.34	92.78 ± 0.09	93.95 ± 0.02	96.15 ± 0.03	97.44 ± 0.03	98.30 ± 0.07
AA/%	82.04 ± 3.97	83.04 ± 2.45	85.53 ± 0.44	88.61 ± 1.24	89.66 ± 0.98	94.43 ± 1.07	95.50 ± 0.44	97.17 ± 0.25
100 K	85.85 ± 0.37	87.85 ± 0.36	89.43 ± 0.40	91.53 ± 0.11	91.73 ± 0.03	95.32 ± 0.02	96.97 ± 0.08	98.01 ± 0.08
Params/M	0.0170	0.0677	0.9028	1.0653	0.6211	0.3807	0.3672	0.1564
Flops/G	0.0009	0.1466	0.4719	0.4780	0.1035	3.9640	4.0821	0.0742

As shown in Tables 5–7, our proposed MS^{3}ANAS typically outperforms other methods in terms of OA, AA, and Kappa on three datasets. The EMP-SVM has the lowest classification accuracy, and the classification accuracy of 2D-CNN and 3D-CNN is also poor, while SSRN uses skip connections to extract depth feature information, resulting in higher classification accuracy than 2D-CNN or 3D-CNN. This is due to the lightweight spatial–spectral attention operations found in multi-scale attention search spaces that can better extract joint spectral–spatial features of hyperspectral images. Taking the Pavia dataset as an example, compared to CNAS, the MS^{3}ANAS model proposed in this paper improves OA from 97.42% to 99.16%. Both ANAS-CPA-LS and MS^{3}ANAS have built the search space of attention mechanism, but the search space built by our method contains attention mechanisms of multiple scales. At the same time, we use small filters to reduce parameters while maintaining a large range of receptive fields. Therefore, the ability to extract different scales of spectral–spatial features from data is enhanced, thereby improving classification performance. However, in the case of limited training samples, the ability of attention mechanisms to fit spatial structural information is not easily utilized. Compared with ANAS-CPA-LS, MS^{3}ANAS has better classification performance.

As shown in Table 6, compared with EMP-SVM, 2D-CNN, 3D-CNN, SSRN, ResNet, CNAS, and ANAS-CPA-LS, OA obtained by our proposed method increased by 13.94%, 10.12%, 6.73%, 6.03%, 3.90%, 1.37%, and 0.72%, respectively, on the Xuzhou dataset. Taking the WHU-Hi-Hanchuan dataset as an example, OA reached 98.30%, and increased by 0.86%, 2.15%, 4.35%, 5.52%, 7.29%, 9.23%, and 10.34%, respectively, compared with ANAS-CPA-LS, CNAS, ResNet, SSRN, 3D-CNN, 2D-CNN, and EMP-SVM. Except for EMP-SVM, 2D-CNN is the second worst in performance because it only uses spatial features. Meanwhile, for other methods, MS^{3}ANAS achieved significant improvements in both AA and Kappa results. From the classification results of the three datasets, it can be seen that the overall performance of the NAS-based method is higher than that of the handmade methods. At the same time, the search space using a multi-scale attention mechanism can enhance the model's ability to consider spectral and spatial information, effectively improving classification performance.

Due to the design of a lightweight NAS model in this article, we compared its complexity with existing technologies in terms of required model parameters (Params) and Flops. We can see from Tables 5–7 that different search space operations and architecture update methods can lead to differences in the amount of network parameters. Flops are determined by the input data size and parameters; therefore, we will conduct an overall analysis based

on the complexity and classification performance of the network model. Compared with handmade hyperspectral image classification models such as 2D-CNN and 3D-CNN, as well as NAS methods such as CNAS and ANAS-CPA-LS, the proposed method has lower occupancy rates in Params and Flops but achieves the highest OA while achieving the best classification performance on three datasets. The results further indicate that the proposed method can solve classification tasks with low cost but high performance. This mainly benefits from the effectiveness of our special multi-scale search space and the applicability of the slow–fast learning architecture update paradigm to different hyperspectral datasets.

The classification diagrams of the other seven methods on the three HSI datasets are displayed in Figures 5–7. It is evident that the proposed algorithm demonstrates superior performance in general. Compared with ANAS-CPA-LS, the ANAS-CPA-LS incorrectly classified Bare Soil (Class 6) as Meadows (Class 2) in Figure 5h of the Pavia dataset classification results.

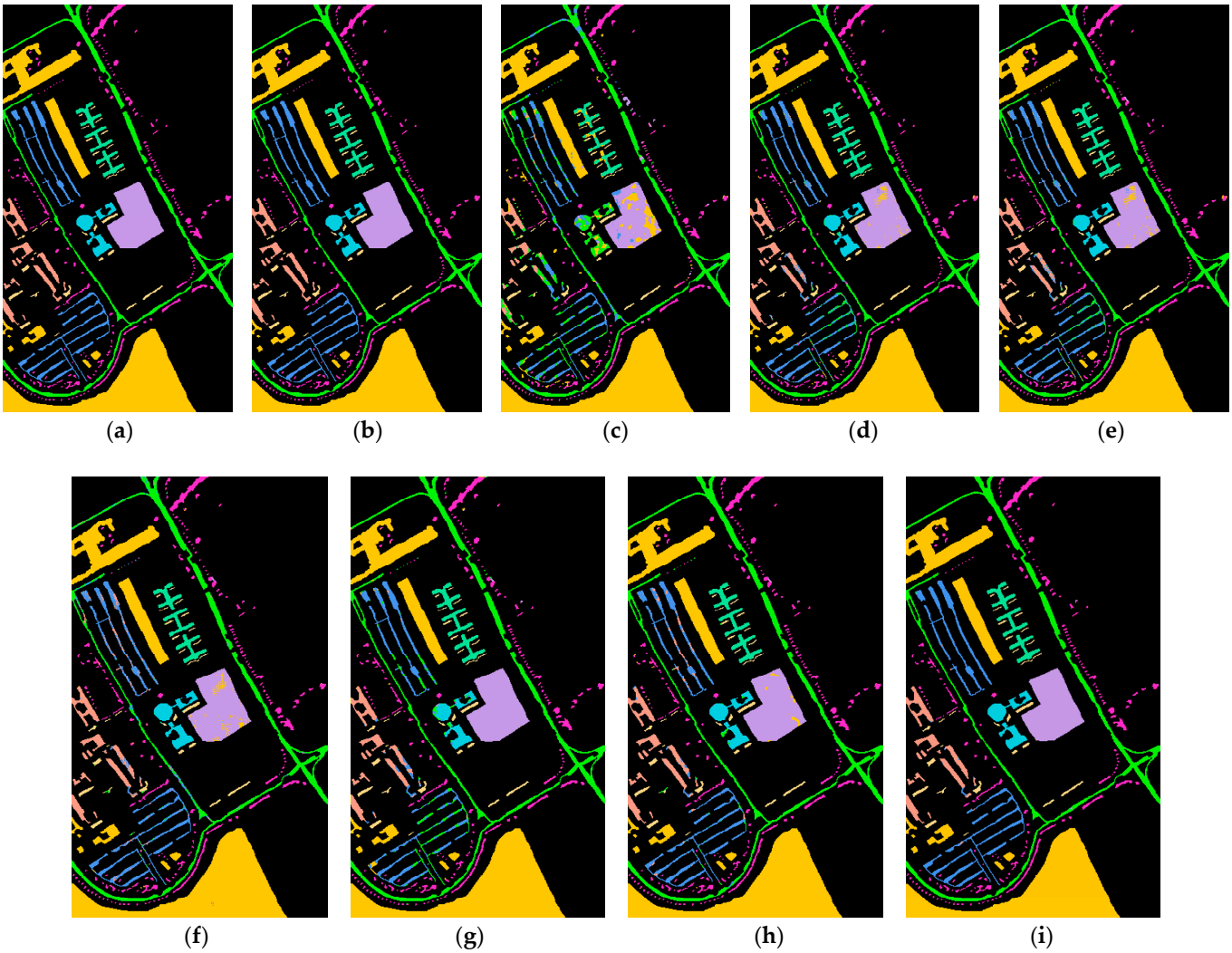


Figure 5. The classification results of the Pavia University dataset. (**a**) Ground truth map; (**b**) EMP-SVM; (**c**) 2D-CNN; (**d**)3D-CNN; (**e**) SSRN; (**f**) MLP-Mixer; (**g**) CNAS; (**h**) ANAS-CPA-LS; and (**i**) MS^{3}ANAS.

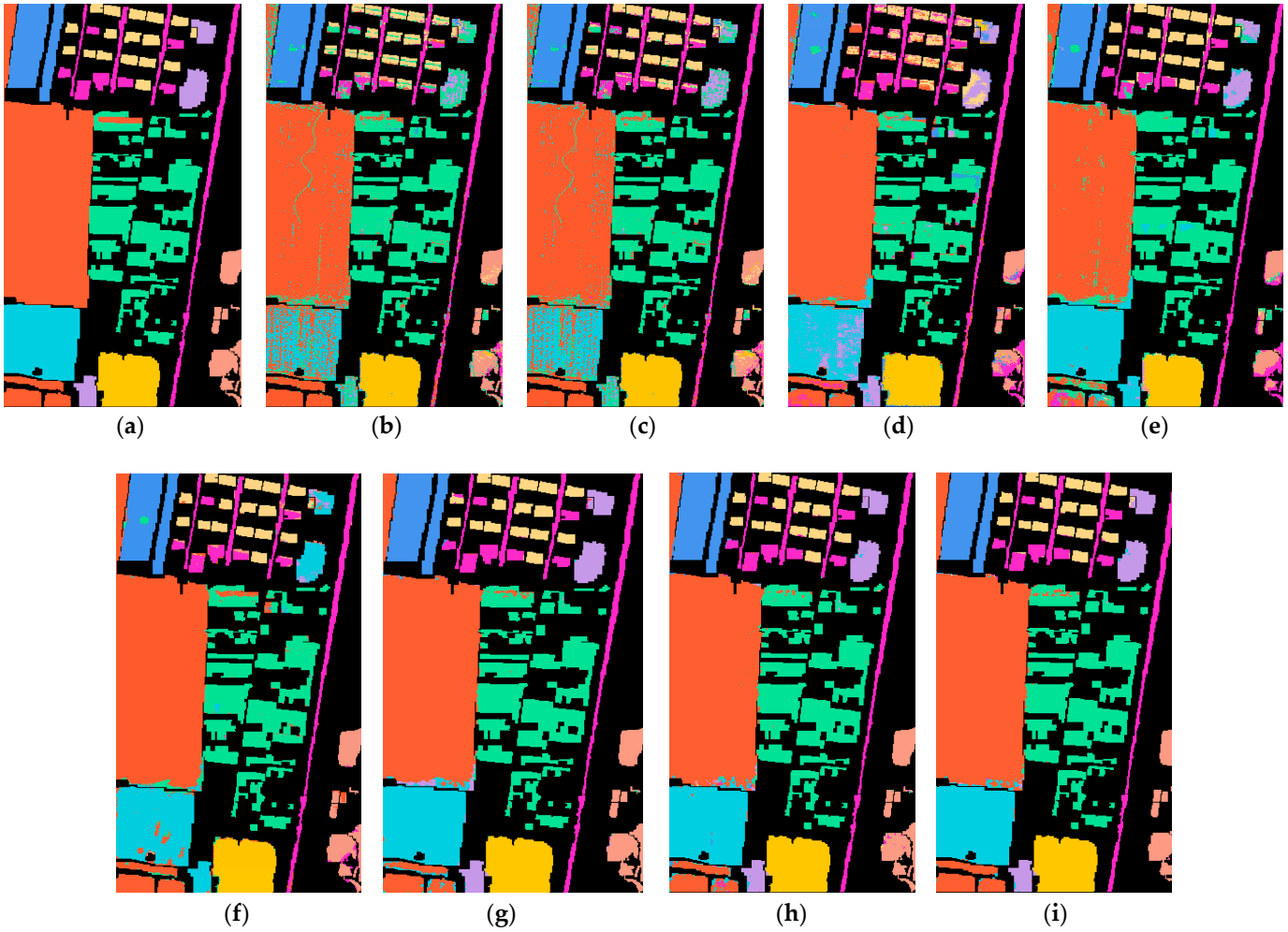


Figure 6. The classification results of the Xuzhou dataset. (**a**) Ground truth map; (**b**) EMP-SVM; (**c**) 2D-CNN; (**d**) 3D-CNN; (**e**) SSRN; (**f**) MLP-Mixer; (**g**) CNAS; (**h**) ANAS-CPA-LS; and (**i**) MS^{3}ANAS.

In Figure 6h of the experimental results on the Xuzhou dataset, some Coals (level 2) were mistakenly classified as Cement (level 7). Through a comparison of the classification maps obtained, it can be concluded that our method achieves more accurate classification results with reduced salt-and-pepper noise. Because our method uses the combination of multi-scale attention mechanism and common attention mechanism as the search operator, it can more effectively extract deeper spatial features and combine spectral and spatial information to capture more complex and abstract features.

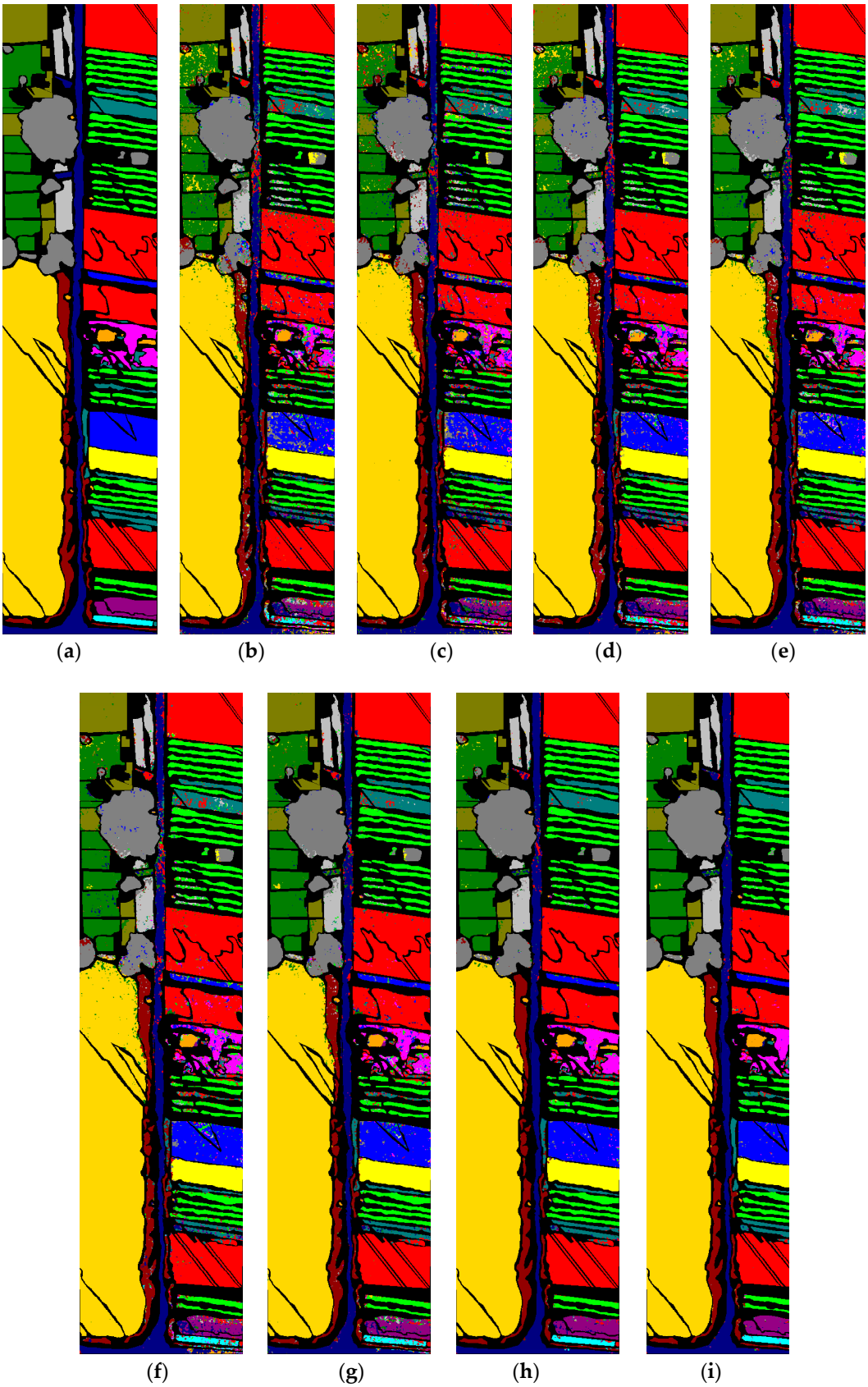
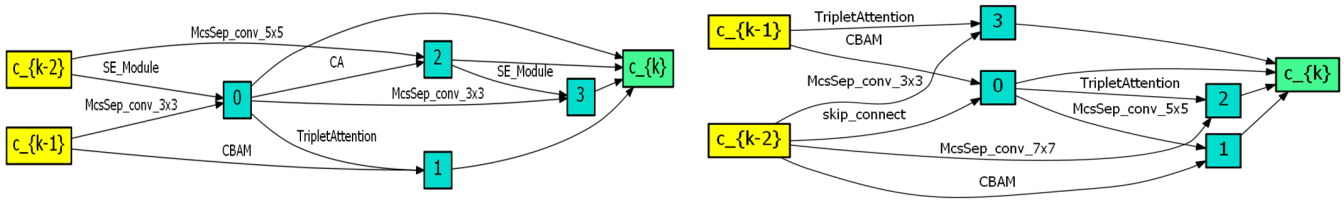


Figure 7. The classification results of the WHU-Hi-Hanchuan dataset. (**a**) Ground truth map; (**b**) EMP-SVM; (**c**) 2D-CNN; (**d**) 3D-CNN; (**e**) SSRN; (**f**) MLP-Mixer; (**g**) CNAS; (**h**) ANAS-CPA-LS; and (**i**) MS^{3}ANAS.

4. Discussion

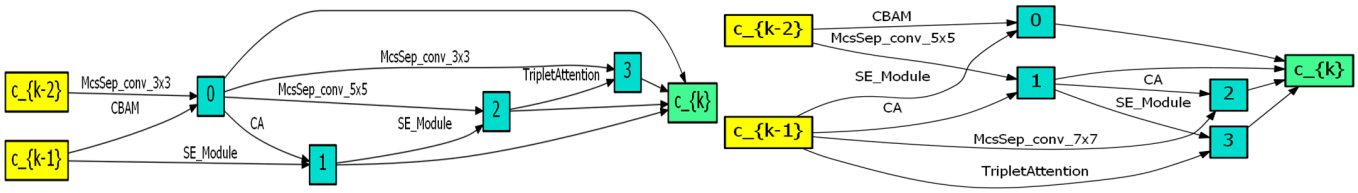
4.1. Optimal Architecture Analysis

Figures 8–10 illustrate the cells with the optimal architecture on the three datasets, which are obtained through the proposed method. These cells consist of both normal cells and reduction cells.



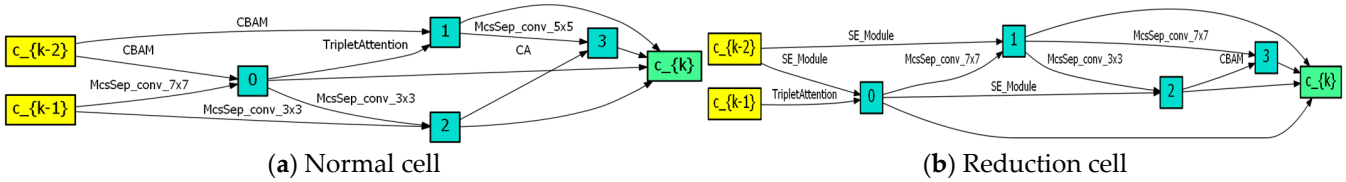
(**a**) Normal cell (**b**) Reduction cell

Figure 8. The searched cell architectures of the Pavia University dataset.



(**a**) Normal cell (**b**) Reduction cell

Figure 9. The searched cell architectures of the Xuzhou dataset.



(**a**) Normal cell (**b**) Reduction cell

Figure 10. The searched cell architectures of the WHU-Hi-Hanchuan dataset.

In our experiment, we employed the slow–fast learning paradigms and the multi-scale attention mechanism search space to search for the optimal network architecture for each HSI dataset. We can observe that the network architecture retains more multi-scale search operators (such as MCS_ sepConv (3×3), MCS_ SepConv (5×5), and MCS_ SepConv (7×7)). As a result, our model comprises a higher number of learnable parameters, allowing it to maintain a wide range of receptive fields. This effectively addresses the issues of gradient disappearance and network degradation. Consequently, our method can extract scaled spectral–spatial features from the data more effectively, leading to an improvement in classification performance.

4.2. The Analysis of Slow and Fast Learning Using Optimal Combination Actions

In order to empirically analyze the slow–fast learning process, it is necessary to select different parameters for experiments. In this section, the generation is combined with different search spaces to conduct experiments on three datasets. In the experiment, the control variable method is used for each dataset. The number of experiments, training samples, validation samples, and test samples is consistent. The conclusion drawn is that the (a) Pavia University, (b) Xuzhou, and (c) WHU-Hi-Hanchuan datasets display the complete presentation of each combination, and all datasets display MA + MS and generation 50 as the best performance combination (Figure 11). Therefore, it can be proven that slow–fast learning can perform pseudo gradient architecture updates on different hyperspectral imbalanced data during the construction process of building units, effectively improving classification performance.

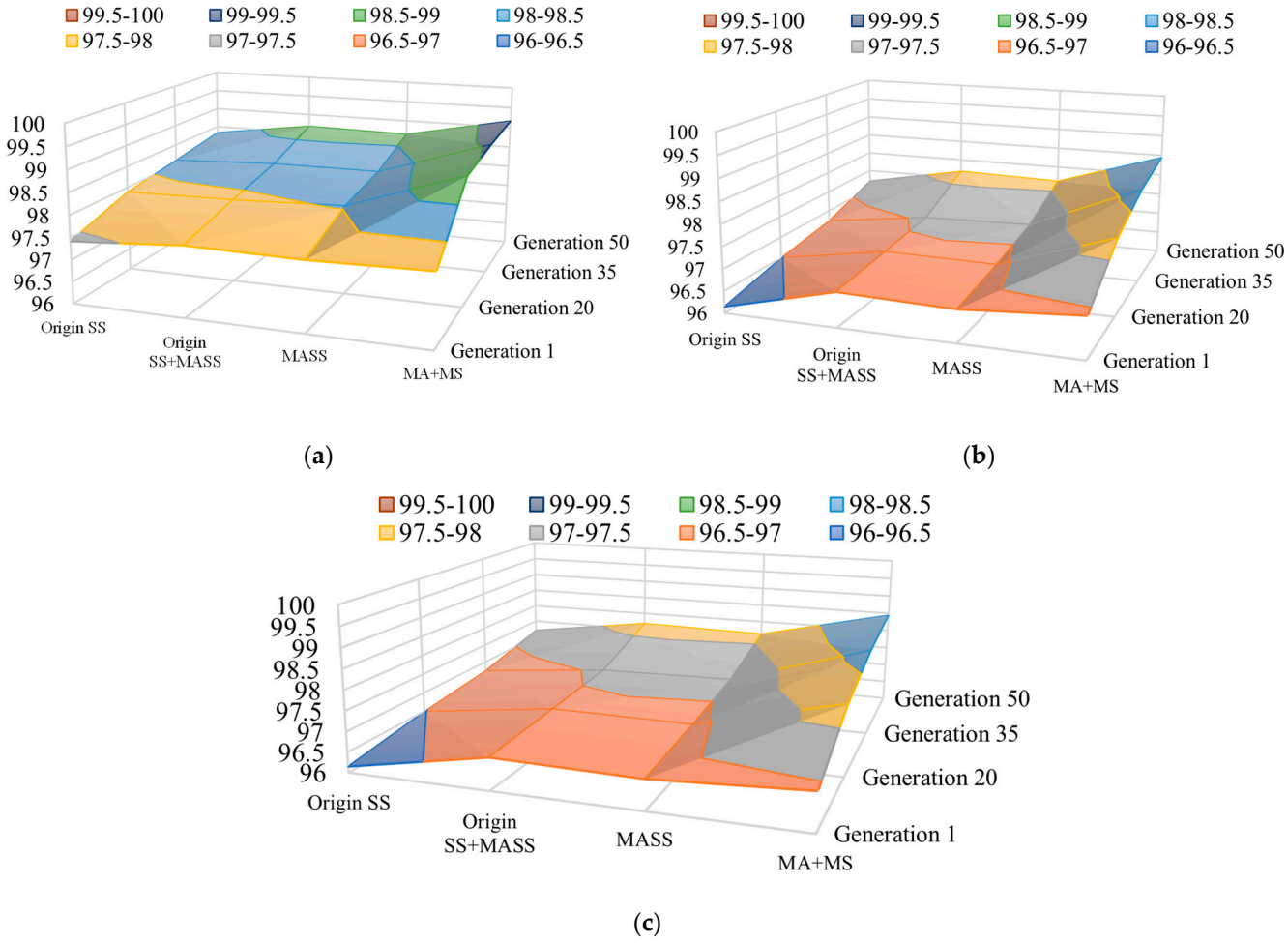


Figure 11. The optimal combined action of the search space and the number of generations for each dataset. (**a**) Pavia University; (**b**) Xuzhou; (**c**) WHU-Hi-Hanchuan.

4.3. Ablation Experiments Analysis

To validate the effectiveness of our method in HSI data classification, we conducted numerous ablation experiments, as presented in Table 8. Specifically, when we solely employ the CNAS model for classifying the Pavia dataset with only 30 training samples, OA reached 97.42%, 97.07%, and 96.15%, respectively. When CNAS was combined with multi-scale attention expansion mechanism search space (MS), OA increased by 0.48%, 0.77%, and 0.89%, respectively. This result demonstrates that the incorporation of a multi-scale attention mechanism to expand the search space enhances the network's sensitivity to informative features, resulting in a slight improvement in the classification performance of the model. In comparison to CNAS, our method achieves better results. the search time of MS + NAS on the three datasets does not change significantly, but it significantly reduces the parameters and achieves better classification accuracy. Obviously, when NAS was combined with the slow–fast learning paradigm search strategy, the OA of the three datasets increased by 1.02%, 0.39%, and 0.89%, respectively. This is due to the application of slow and fast learning paradigms to optimize and interactively update the architecture vector. It can update the architecture of different hyperspectral imbalanced data and obtain classification models with stronger generalization ability. At the same time, it can be observed that the search time for the three datasets was reduced by 0.249 h, 0.111 h, and 0.087 h after SF + MSNAS referenced the Lion optimizer. This is because this optimizer can effectively reduce the time consumption of NAS architecture search. In addition, the algorithm only tracks momentum, effectively reducing memory overhead.

Table 8. The classification results of all methods on each dataset.

Dataset	Index	CNAS	MS + NAS	SF + MSNAS	SF + MSNAS + Lion
Pavia University	OA (%)	97.42 ± 0.29	97.90 ± 0.37	98.92 ± 0.20	99.16 ± 0.03
	AA (%)	97.18 ± 0.25	97.84 ± 1.36	98.07 ± 1.83	98.91 ± 1.14
	100 K	96.76 ± 0.18	97.95 ± 0.79	98.45 ± 2.51	98.89 ± 0.09
	Search Cost (hours)	3.054	3.013	2.982	2.733
	Params (M)	0.1692	0.1574	0.1579	0.1564
Xuzhou	OA (%)	97.07 ± 0.06	97.84 ± 0.94	98.23 ± 0.09	98.44 ± 1.67
	AA (%)	96.74 ± 1.82	97.94 ± 0.81	98.70 ± 0.59	98.98 ± 0.25
	100 K	97.04 ± 1.01	97.18 ± 0.78	97.84 ± 0.42	98.04 ± 2.10
	Search Cost (hours)	3.312	3.295	3.205	3.094
	Params (M)	0.0658	0.0570	0.0578	0.059
WHU-Hi-Hanchuan	OA (%)	96.15 ± 0.03	97.04 ± 0.94	97.93 ± 0.09	98.30 ± 0.07
	AA (%)	94.43 ± 1.07	96.52 ± 0.81	97.01 ± 0.59	97.17 ± 0.25
	100 K	95.32 ± 0.02	96.11 ± 0.78	97.74 ± 0.42	98.01 ± 0.08
	Search Cost (hours)	4.201	4.178	4.113	4.026
	Params (M)	0.292	0.174	0.176	0.156

5. Conclusions

In this paper, we introduce a neural network architecture search algorithm called MS^{3}ANAS. This approach aims to address the limitations of conventional differentiable NAS methods in three key areas: search space, search strategy, and architecture resource optimization. Firstly, we introduce an extended search space with a multi-scale attention mechanism, which can not only enhance the robustness and receptive field capture ability of the model but also better realize path optimization. Secondly, we quoted the slow–fast architecture learning paradigm, which not only optimizes the iterative updating of architecture vectors but also effectively improves the model's generalization ability. Finally, the Lion optimizer was introduced to improve the accuracy of HSI data classification, reduce memory overhead, and reduce computational complexity. We conducted experiments on three HSI datasets and compared MS^{3}ANAS with seven methods. The experimental results show that our method is more competitive. Through the datasets of Xuzhou, Pavia University, and WHU-Hi-Hanchuan, the OA of our method reached 97.82%, 98.46%, and 98.41%, respectively.

In future work, we will further optimize the NAS structure continuously to make it more lightweight and easier to adapt to more complex remote sensing image classification tasks. Furthermore, it is important to consider the practical applicability of our proposed method. Meanwhile, we plan to apply our approach to real-world hyperspectral image classification tasks, such as environmental monitoring, agricultural assessment, and urban planning. By validating the performance of our method on these practical applications, we can demonstrate its effectiveness and potential impact in various domains. Moreover, the potential applications of our method extend beyond hyperspectral image classification. By exploring these new application areas, we can further broaden the scope of our research and contribute to advancements in related fields. We are excited about the opportunities that lie ahead and look forward to witnessing the practical implementation and impact of our work in the field of remote sensing and computer vision.

Author Contributions: Conceptualization, Y.S., Y.Z., Y.I., H.W. and A.W.; methodology, Y.S. and Y.Z.; software, Y.S.; validation, Y.S.; writing—review and editing, Y.Z., Y.I., H.W. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the High-end Foreign Experts Introduction Program (G2022012010L), Heilongjiang Natural Science Foundation Project (LH2023F034), and Reserved Leaders of Heilongjiang Provincial Leading Talent Echelon (2021).

Data Availability Statement: https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes (accessed on 20 May 2011); http://rsidea.whu.edu.cn/resource_WHUHi_sharing.html (accessed on 21 September 2021); https://github.com/szubing/ED-DMM-UDA (accessed on 5 October 2019).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Transon, J.; d'Andrimont, R.; Maugnard, A.; Defourny, P. Survey ofhyperspectral earth observation applications from space in the sentinel-2 context. *Remote Sens.* **2018**, *10*, 157. [CrossRef]
2. Carrino, T.A.; Crósta, A.P.; Toledo, C.L.B.; Silva, A.M. Hyper-spectral remote sensing applied to mineral exploration in southern peru:A multiple data integration approach in the chapi chiara gold prospect. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *64*, 287–300.
3. Behmann, J.; Steinrücken, J.; Plümer, L. Detection of early plant stress responses in hyperspectral images. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 98–111. [CrossRef]
4. Shimoni, M.; Haelterman, R.; Perneel, C. Hypersectral imaging for military and security applications: Combining myriad processing and sensing techniques. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 101–117. [CrossRef]
5. Sun, W.; Peng, J.; Yang, G.; Du, Q. Fast and latent low-ranksubspace clustering for hyperspectral band selection. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 3906–3915. [CrossRef]
6. Samadzadegan, F.; Hasani, H.; Schenk, T. Simultaneous feature selection and SVM parameter determination in classification of hyperspectral imagery using Ant Colony Optimization. *Can. J. Remote Sens.* **2012**, *38*, 139–156. [CrossRef]
7. Chandra, B.; Sharma, R.K. On improving recurrent neural network for image classification. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 1904–1907. [CrossRef]
8. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep convolutionalneural networks for hyperspectral image classification. *J. Sens.* **2015**, *2015*, 258619. [CrossRef]
9. Makantasis, K.; Karantzalos, K.; Doulamis, A.; Doulamis, N. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 4959–4962.
10. Li, W.; Wu, G.; Zhang, F.; Du, Q. Hyperspectral Image Classification Using Deep Pixel-Pair Features. *IEEE Trans. Geosci. Remote Sens.* **2016**, *55*, 844–853. [CrossRef]
11. Yue, J.; Zhao, W.; Mao, S.; Liu, H. Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sens. Lett.* **2015**, *6*, 468–477. [CrossRef]
12. Zhang, H.; Li, Y.; Zhang, Y.; Shen, Q. Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network. *Remote Sens. Lett.* **2017**, *8*, 438–447. [CrossRef]
13. Wang, B.; Shao, Q.; Song, D.; Li, Z.; Tang, Y.; Yang, C.; Wang, M. A Spectral-Spatial Features Integrated Network for Hyperspectral Detection of Marine Oil Spill. *Remote Sens.* **2021**, *13*, 1568. [CrossRef]
14. Li, Y.; Zhang, H.; Shen, Q. Spectral–spatial classification of hyper-spectral imagery with 3d convolutional neural network. *Remote Sens.* **2017**, *9*, 67. [CrossRef]
15. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [CrossRef]
16. Zhong, Z.; Li, J.; Luo, Z.; Chapman, M. Spectral–spatial residual network for hyperspectral image classification: A 3-d deep learning framework. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 847–858. [CrossRef]
17. Zhang, H.; Li, Y.; Jiang, Y.; Wang, P.; Shen, Q.; Shen, C. Hyperspectral classification based on lightweight 3-d-cnn with transfer learning. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 5813–5828. [CrossRef]
18. Yang, G.; Gewali, U.B.; Ientilucci, E.; Gartley, M.; Monteiro, S.T. Dual-Channel Densenet for Hyperspectral Image Classification. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 2595–2598.
19. Bagaskara, A.; Suryanegara, M. Evaluation of VGG-16 and VGG-19 Deep Learning Architecture for Classifying Dementia People. In Proceedings of the 2021 4th International Conference of Computer and Informatics Engineering (IC2IE), Depok, Indonesia, 14–15 September 2021; pp. 1–4.
20. Aswathy, P.; Siddhartha; Mishra, D. Deep GoogLeNet Features for Visual Object Tracking. In Proceedings of the 2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS), Rupnagar, India, 1–2 December 2018; pp. 60–66.
21. Xia, M.; Yuan, G.; Yang, L.; Xia, K.; Ren, Y.; Shi, Z.; Zhou, H. Few-Shot Hyperspectral Image Classification Based on Convolutional Residuals and SAM Siamese Networks. *Electronics* **2023**, *12*, 3415. [CrossRef]
22. Jiang, Y.; Yu, S.; Wang, T.; Sun, Z.; Wang, S. Skeleton-Based Human Action Recognition Based on Single Path One-Shot Neural Architecture Search. *Electronics* **2023**, *12*, 3156. [CrossRef]
23. Xu, H.; Yao, L.; Li, Z.; Liang, X.; Zhang, W. Auto-FPN: Auto-matic network architecture adaptation for object detection beyond classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6648–6657.

24. Zoph, B.; Le, Q.V. Neural architecture search with reinforcement earning. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017; pp. 1–16.
25. Real, E.; Aggarwal, A.; Huang, Y.; Le, Q.V. Regularized evolution for image classifier architecture search. In Proceedings of the Association for the Advancement of Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 4780–4789.
26. Tan, M.; Chen, B.; Pang, R.; Vasudevan, V.; Sandler, M.; Howard, A.; Le, Q.V. MnasNet: Platform-Aware neural architecture search for mobile. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 2820–2828.
27. Jin, J.; Zhang, Q.; He, J.; Yu, H. Quantum Dynamic Optimization Algorithm for Neural Architecture Search on Image Classification. *Electronics* **2022**, *11*, 3969. [CrossRef]
28. Liu, H.; Simonyan, K.; Yang, Y. DARTS: Differentiable architecture search. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 30 April–30 May 2019; pp. 1–13.
29. Stanley, K.O.; Clune, J.; Lehman, J.; Miikkulainen, R. Designing neural networks through neuroevolution. *Nat. Mach. Intell.* **2019**, *1*, 24–35. [CrossRef]
30. Real, E.; Moore, S.; Selle, A.; Saxena, S.; Suematsu, Y.L.; Tan, J.; Le, Q.V.; Kurakin, A. Large-scale evolution of image classifiifiers. In Proceedings of the 34th International Conference on Machine Learning, PMLR, Sydney, NSW, Australia, 6–11 August 2017; pp. 2902–2911.
31. Liang, H.; Zhang, S.; Sun, J.; He, X.; Huang, W.; Zhuang, K.; Li, Z. DARTS+: Improved differentiable architecture search with early stopping. *arXiv* **2020**, arXiv:1909.06035.
32. Guo, Z.; Zhang, X.; Mu, H.; Heng, W.; Liu, Z.; Wei, Y.; Sun, J. Single path one-shot neural architecture search with uniform sampling. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Volume 12361, pp. 544–560. [CrossRef]
33. Park, G.; Yi, Y. CondNAS: Neural Architecture Search for Conditional CNNs. *Electronics* **2022**, *11*, 1101. [CrossRef]
34. Stanley, K.O.; Miikkulainen, R. Evolving neural networks through augmenting topologies. *Evol. Comput.* **2002**, *10*, 99–127. [CrossRef] [PubMed]
35. Chen, Y.; Zhu, K.; Zhu, L.; He, X.; Ghamisi, P.; Benediktsson, J.A. Automatic design of convolutional neural network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7048–7066. [CrossRef]
36. Hu, J.; Shen, L.; Sun, G. Squeeze–and–excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
37. Roy, S.K.; Manna, S.; Song, T.; Bruzzone, L. Attention-Based Adaptive Spectral–Spatial Kernel ResNet for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 7831–7843. [CrossRef]
38. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
39. Misra, D.; Nalamada, T.; Arasanipalai, A.U.; Hou, Q. Rotate to Attend: Convolutional Triplet Attention Module. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 3138–3147.
40. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the Computer Vision and Pattern Recognition, Beijing, China, 29 October–1 November 2021.
41. Cao, C.; Xiang, H.; Song, W.; Yi, H.; Xiao, F.; Gao, X. Lightweight Multiscale Neural Architecture Search with Spectral–Spatial Attention for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–15. [CrossRef]
42. Tan, H.; Cheng, R.; Huang, S.; He, C.; Qiu, C.; Yang, F.; Luo, P. RelativeNAS: Relative Neural Architecture Search via Slow-Fast Learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *34*, 475–489. [CrossRef]
43. Chen, X.; Liang, C.; Huang, D.; Real, E.; Wang, K.; Liu, Y.; Pham, H.; Dong, X.; Luong, T.; Hsieh, C.-J.; et al. Symbolic Discovery of Optimization Algorithms. *arXiv* **2023**, arXiv:2302.06675.
44. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [CrossRef]
45. Liu, X.; Meng, Y.; Fu, M. Classification Research Based on Residual Network for Hyperspectral Image. In Proceedings of the 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP), Wuxi, China, 19–21 July 2019; pp. 911–915.
46. He, X.; Chen, Y. Modifications of the Multi-Layer Perceptron for Hyperspectral Image Classification. *Remote Sens.* **2021**, *13*, 3547. [CrossRef]
47. Wang, A.; Xue, D.; Wu, H.; Gu, Y. Efficient Convolutional Neural Architecture Search for LiDAR DSM Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–17. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Gaze Estimation Method Combining Facial Feature Extractor with Pyramid Squeeze Attention Mechanism

Jingfang Wei [1], Haibin Wu [1,*], Qing Wu [1], Yuji Iwahori [2], Xiaoyu Yu [3] and Aili Wang [1,*]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; 2220610105@stu.hrbust.edu.cn (J.W.); wuqing@hrbust.edu.cn (Q.W.)
[2] Department of Computer Science, Chubu University, Kasugai 487-8501, Japan; iwahori@isc.chubu.ac.jp
[3] College of Electron and Information, University of Electronic Science and Technology of China, Zhongshan Institute, Zhongshan 528402, China; yuxy@zsc.edu.cn
* Correspondence: woo@hrbust.edu.cn (H.W.); aili925@hrbust.edu.cn (A.W.)

Abstract: To address the issue of reduced gaze estimation accuracy caused by individual differences in different environments, this study proposes a novel gaze estimation algorithm based on attention mechanisms. Firstly, by constructing a facial feature extractor (FFE), the method obtains facial feature information about the eyes and locates the feature areas of the left and right eyes. Then, the L2CSNet ($\updownarrow_2$ loss + cross-entropy loss + softmax layer network), which integrates the PSA (pyramid squeeze attention), is designed to increase the correlation weights related to gaze estimation in the feature areas, suppress other irrelevant weights, and extract more fine-grained feature information to obtain gaze direction features. Finally, by integrating L2CSNet with FFE and PSA, FPSA_L2CSNet was proposed, which is fully tested on four representative publicly available datasets and a real-world dataset comprising individuals of different backgrounds, lighting conditions, nationalities, skin tones, ages, genders, and partial occlusions. The experimental results indicate that the accuracy of the gaze estimation model proposed in this paper has been improved by 13.88%, 11.43%, and 7.34%, compared with L2CSNet, FSE_L2CSNet, and FCBA_L2CSNet, respectively. This model not only improves the robustness of gaze estimation but also provides more accurate estimation results than the original model.

Keywords: gaze estimation algorithm; face feature extractor; feature information; PSA attention mechanism

1. Introduction

With the rapid development of technology and society, human–computer interaction [1,2] has improved people's work efficiency and quality of life, which plays a crucial role in many applications such as virtual reality [3] and medical research [4,5]. Eye detection and tracking is a challenging task in the field of computer vision. It provides useful information for human–computer interaction. The researchers have utilized gaze estimation to study behavior in many domains. It aims to infer people's focus of attention and behavioral intention by analyzing and understanding the fixation point and direction of the human eye. Sight is the nonverbal information in human communication, which can reveal people's interests, emotions, and cognitive process. Therefore, gaze estimation has broad application value in fields such as human–computer interaction, intelligent monitoring, medical diagnosis, and intelligent driving.

In human–computer interaction, gaze estimation can be used as an "eye mouse" to control the operation of the computer interface by capturing the fixation point of the human eye, achieving a more natural and intuitive interaction experience. This can be used to improve user interface design. By analyzing the user's gaze patterns and eye movements, it is possible to understand the user's attention allocation and interaction behavior toward

interface elements. This is very helpful for optimizing interface layout, enhancing the visibility of key features, and providing personalized interactive experiences. For example, in web design, gaze estimation can determine the user's fixation point and reading path when browsing a webpage, thereby adjusting the layout of content and the position of key information, making it easier for users to find the required information.

In terms of intelligent driving, firstly, gaze estimation can be used for driver fatigue and attention monitoring. By analyzing the driver's gaze patterns and eye movements, it is possible to detect whether they experience fatigue, distraction, or lack of concentration. Once the driver's attention drops, the system can issue a warning in a timely manner, reminding the driver to take necessary rest or attention adjustments to avoid traffic accidents. Secondly, gaze estimation can be used for driver behavior analysis. By analyzing drivers' gaze points and gaze directions during driving, we can understand their attention distribution on roads, traffic signs, and other vehicles. This can provide valuable data support for the development of driver behavior models and the optimization of driving decision systems. In addition, gaze estimation can also be used for monitoring the driver's emotional and cognitive state. By observing the driver's gaze patterns and eye movements, one can infer their emotional state, such as anxiety, fatigue, or distraction. This is very important for intelligent driving systems, as it can adjust driving strategies and provide personalized driving experiences based on the driver's emotional and cognitive state.

In terms of intelligent monitoring, gaze estimation can be used for behavior analysis and anomaly detection. By analyzing people's gaze patterns and eye movements, we can understand their points of interest and areas of interest in monitoring scenarios. This is very helpful for detecting suspicious behavior, abnormal activities, or potential threats. For example, in a shopping mall monitoring system, if someone's gaze frequently shifts to a specific area, it may indicate that they are engaging in theft or other improper behavior. Gaze estimation can help intelligent monitoring systems detect and alarm in a timely manner.

In medical diagnosis, the use of eye-tracking technology enables real-time monitoring of patients' fixation points and directions, providing important information about their attention allocation and cognitive processes. Eye line estimation can be used for ophthalmic diagnosis and treatment. By analyzing the patient's gaze patterns and eye movements, their eye coordination, eye movement function, and abnormal eye movements can be evaluated. This is very helpful for early detection and treatment of eye diseases. For example, in the diagnosis of strabismus, by observing the patient's fixation point and eye movements, the type and degree of strabismus can be determined, and an appropriate treatment plan can be developed. In addition, it can be used for neuroscience research and brain function localization. By analyzing patients' gaze patterns and eye movements, we can understand their attention allocation and cognitive processing processes toward stimuli. This is very helpful for studying the cognitive function of the brain and neurological diseases. For example, in the study of cognitive process, eyesight estimation can determine the cognitive load and attention distribution of patients under different tasks and stimuli, thus revealing the neural mechanism of a cognitive process.

2. Related Work

At first, human gaze estimation information is mainly obtained through mechanical, electrical signals, or traditional algorithms. T. Eggert et al. used a recognition algorithm for electrical signals to achieve human gaze estimation technology. This method mainly uses intelligent sensors to sample electrical signals from skin electrodes and represents the subject's eye movement through changes in this electrical signal [6]. This method requires connecting the device to the head position of the subject, which is relatively simple to implement. However, traditional devices are bulky, inconvenient for the subject to move, and have low accuracy. Later, with the continuous development of technology, many researchers began to use image processing algorithms to achieve gaze estimation. Zhang et al. proposed a nonlinear unscented Kalman filter for gaze estimation [7], which

can overcome the difficulties of nonlinear gaze estimation and improve the accuracy of gaze estimation. Jiashu Zhang et al. proposed to use several groups of points to match the posterior probability density function of eye movement [8], which is more accurate than the estimation effect of the traditional Kalman filter. Although nonlinear Kalman filtering algorithms can improve the accuracy of gaze estimation, they are numerically unstable in practical applications and require more computational time. The process of optimizing the structure is complex and has poor real-time performance. Amudha J et al. used deformable template matching to achieve various challenges in gaze estimation, which achieved a tracking accuracy of 98% [9]. L. Yu et al. proposed an eye gaze estimation method based on particle swarm optimization [10], which reduces the restrictive requirements for hardware and improves the practicability of the system. Although the gaze estimation technology based on genetic algorithm and particle swarm optimization algorithm can achieve accurate gaze estimation, the algorithm design is difficult, the parameter adjustment range is wide, and it is time-consuming and laborious.

With the continuous innovation of technology, deep learning methods have gradually attracted more and more attention from researchers. Compared to the Kalman filter algorithm, genetic algorithm, ant colony algorithm, and so on, deep learning advocates direct end-to-end solutions to the problem of gaze estimation, which can be classified into two types:

1. The model-based gaze estimation methods

The model-based gaze estimation is a state-of-the-art computer vision technique that harnesses the power of machine learning algorithms to simulate the intricate movement of human eyes and accurately predict gaze points. B. Yan et al. presented an innovative technique for estimating gaze direction using differential residual networks [11], which can measure the disparity in gaze between two eyes with higher precision. H. Zhang et al. developed a multitask network model to estimate eye gaze and blinks concurrently [12]. S. H. Kim et al. proposed a continuous engagement evaluation framework utilizing estimated gaze direction and facial expression data [13], substantiated by creating a database of gaze estimates. This method effectively expresses fine-grained state information. Liu et al. proposed a 3D gaze estimation model integrating an automatic calibration mechanism [14], which utilizes eye images to construct a 3D eye model, creating a point cloud with an RGBD camera. This method yields an average precision of 3.7° but requires expensive hardware. J. Ma et al. [15] proposed an eye-specific offset predictor for humans that enables predicting gaze estimation for multiple target figures. J. Zhuang et al. proposed adding a squeeze-and-excitation (SE) attention mechanism to ResNet networks to predict gaze points of flight simulator operators on the screen [16], which fills a void in research on multiscreen and multicamera systems. B. Saha et al. developed a real-time gaze estimation interface based on a subjective appearance method to achieve gaze estimation in a free environment using algorithms like decision trees and random forests [17], which has an accuracy of approximately 98%. However, the error varies in distinct environments and lighting conditions, and the system's stability is suboptimal. Chang C C et al. proposed a simple facial tagging function based on the YOLO model to locate two facial markers and six facial orientations [18]. It achieves an average precision of 99% while detecting facial orientations and facial features. However, the accuracy of locational facial marker points is questionable when subjects move significantly. Z. Wan et al. proposed a fractional perceptual gaze estimation method utilizing the actual pupil axis, which regresses the pupil's spherical coordinates normal to the gaze points [19]. This technique effectively solves the issue of pupil cornea refraction. However, the method did not detect the pupil contour, producing a less accurate model. H. Huang et al. presented a new framework: GazeAttentionNet [20], utilizing global and local attention modules to achieve gaze features. The method obtained high accuracy on a dataset with real-time validation. Nevertheless, the method's accuracy is low, with the average error reaching approximately 2 cm for mobile devices.

2. The appearance-based gaze estimation methods

The appearance-based gaze estimation is a cutting-edge computer vision technique employed to predict a person's gaze direction accurately. This advanced method relies on the analysis of appearance-based features surrounding the human eye to achieve gaze estimation. Zhao et al. proposed a monocular gaze estimation network utilizing mixed attention to predict gaze points from monocular features and their location information [21]. Wan et al. introduced a technique for estimating gaze direction utilizing one mapping surface [22], which accomplished eye-center calibration through the mapping surface, simplifying the model by making assumptions. Zhuang et al. developed a simplified network model for gaze estimation based on the LeNet neural network, called SLeNet [23], utilizing depth-separable convolution to decrease the number of convolution parameters and improve the speed of the model. Nonetheless, the model design of this technique is too simple, and the accuracy needs to be enhanced.

Zhang et al. presented a gaze data processing model based on artificial neural networks and face recognition techniques [24], which analyzed gaze estimation from the perspective of data processing, lessened the false recognition rate, and experimentally verified the effectiveness of the method. Murthy et al. designed an end-to-end gaze estimation system to anticipate gaze points employing infrared eyeglass images captured by wearable vision-estimating eyes [25], which realized comprehensive gaze estimation. Liu et al. developed a technique to predict the disparity in gaze between two eye input images of the same subject by directly training a differential convolutional neural network and predicting the direction of eye gaze using the estimated difference [26]. This method achieves high accuracy but involves a more complicated process of adjusting the parameters of the differential network when the subject's eyelids are closed or when the pupils are affected. Suzuki et al. proposed a sight-tracking dataset-based technique for estimating candidate regions for superimposed information in football videos [27]. Nevertheless, the accuracy of line-of-sight tracking decreases with an increase in superimposed information. Chang et al. proposed a gaze estimation technique based on YOLO and deep learning for localizing and detecting facial orientation by integrating appearance and geometric features [28], which achieved a robust accuracy of 88% without calibration. However, it suffered from slow real-time tracking due to its inability to locate facial feature coordinates. Senarath et al. proposed a three-stage, three-attention deep convolutional neural network for retail remote gaze estimation using image data [29]. Luo et al. proposed a collaborative network-based gaze estimation model with an attention mechanism that assigns appropriate weights between eye and facial features and achieved more accurate gaze estimation [30]. Han et al. proposed a pupil shape-based gaze estimation method, which used a deep network to learn gaze points by extracting various features from the image [31]. X. Song et al. proposed an end-to-end network using U-Net with residual blocks to retain eye features in high-resolution feature maps for efficient gaze estimation [32]. Zhao et al. proposed a unified network for simultaneous head detection and gaze estimation, where the two aspects share the same set of features to promote each other and enhance detection accuracy [33]. However, this method still had limitations in terms of accuracy, with an error of 19.62° at 23 fps.

In summary, the aforementioned algorithms faced two primary challenges. (1) The gaze estimation accuracy is greatly reduced when the subject's face is obscured or in a different environment with varying lighting conditions. (2) The designed network models or algorithms are simple in structure, leading to decreased gaze estimation accuracy.

To address the above problems, this paper proposes a new attention-based mechanism for the gaze estimation algorithm FPSA_L2CSNet, which integrates L2CSNet with facial feature extractor (FFE) and pyramid squeeze attention (PSA), which has the following contribution:

1. A facial feature extractor is integrated with L2CSNet, enabling retrieval of facial details, location of eye features, extraction of key eye points, and efficient narrowing of eye feature extraction range to enhance gaze estimation accuracy.

2. L2CSNet augmented with PSA incorporates multiscale spatial information and cross-channel attention into the model, selectively highlighting feature regions relevant to vision estimation and suppressing irrelevant weights. Further, it facilitates granular level feature extraction, gaze direction feature extraction, and accurate vision estimation.

3. The proposed model is tested on a real-life dataset and four representative public datasets: the MPIIGaze dataset, the Gaze360 dataset, the ETH-XGaze dataset, and the GazeCapture dataset. The robustness and accuracy of the algorithm are verified by testing on individuals of varying nationalities, skin colors, ages, partial occlusion, genders and complex backgrounds, and lighting conditions.

3. Materials and Methods

First, this paper constructed a feature extractor to obtain the facial features, thereby narrowing down the range of eye feature extraction and accurately locating the key points of the eyes. Then, a high-precision gaze estimation algorithm [34] is proposed in this paper, which uses ResNet50 as the backbone network, uses a separate loss function to calculate the error of each gaze angle, and regresses the gaze result separately. Furthermore, to further improve the overall performance of the model and reduce the error of gaze results, this paper also introduces the PSA attention module, which combines multiscale spatial information and cross-channel attention into the model and selectively highlights the feature areas related to gaze estimation. Finally, the accuracy and robustness of the algorithm are verified in different environments. The algorithm flowchart of FPSA_L2CSNet is illustrated in Figure 1.

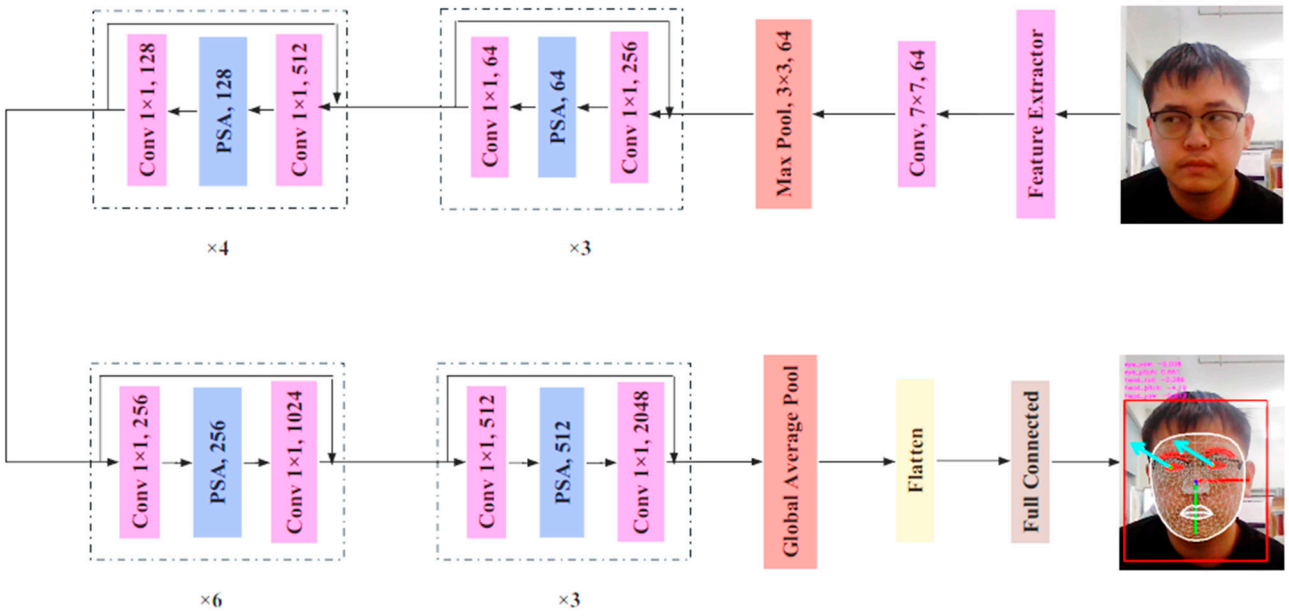


Figure 1. The flowchart of FPSA_L2CSNet model.

Therefore, this paper presents a novel attention-based mechanism, FPSA_L2CSNet, for gaze estimation. This approach leverages attention mechanisms to achieve more precise and efficient gaze estimation. The FPSA_L2CSNet algorithm is trained on different datasets, including MPIIGaze, Gaze360, ETH-XGaze, and GazeCapture, and its performance is thoroughly evaluated via comparison with other models. The results confirm the effectiveness and superiority of FPSA_L2CSNet in achieving more accurate and faster gaze estimation.

3.1. Acquisition of Facial Feature Points

As depicted in Figure 2, the FFE (facial feature extractor) in this study utilizes an RGB input image of 128 × 128 pixels to extract facial features. Due to the inherent variation in pixel composition within the dataset, we shall employ the transformative power of the

resize () function to preprocess the images. By skillfully applying bilinear interpolation, we shall artfully scale the image, ultimately attaining a harmonious size of 128 × 128 pixels. The results are obtained after passing through 5 single BlazeBlocks and 6 double BlazeBlocks. The main route of the single BlazeBlock consists of a 5 × 5 deep convolution and a 1 × 1 convolution, with the depth-separable convolution layer being the core component. In contrast to traditional convolution layers that perform convolution operations in both the spatial and channel directions, the depthwise separable convolutional layer focuses on these two directions separately to reduce the parameter amount. Specifically, the depthwise separable convolution uses the same convolution kernel to perform convolution on each input channel, resulting in a set of individual output channels. In this way, each convolution kernel is reused, enabling the reduction in a significant number of parameters. The element-wise convolution is the second part of the depthwise separable convolution, using a 1 × 1 convolution kernel to transform each individual output channel into the desired shape. In this way, the depthwise separable convolutional layer establishes the connection between depth and width. The side route of the single BlazeBlock consists of max pooling and channel pad, aiming to increase the convolution kernel size and cover the entire receptive field with fewer convolution layers. The skip connections in the single BlazeBlock allow the model to learn higher-level features that match the input better. Typically, cross-layer connections that span multiple layers help capture globally or partially informative features. Additionally, through skip connections, the single BlazeBlock can integrate with deeper grid structures to enhance the model's accuracy.

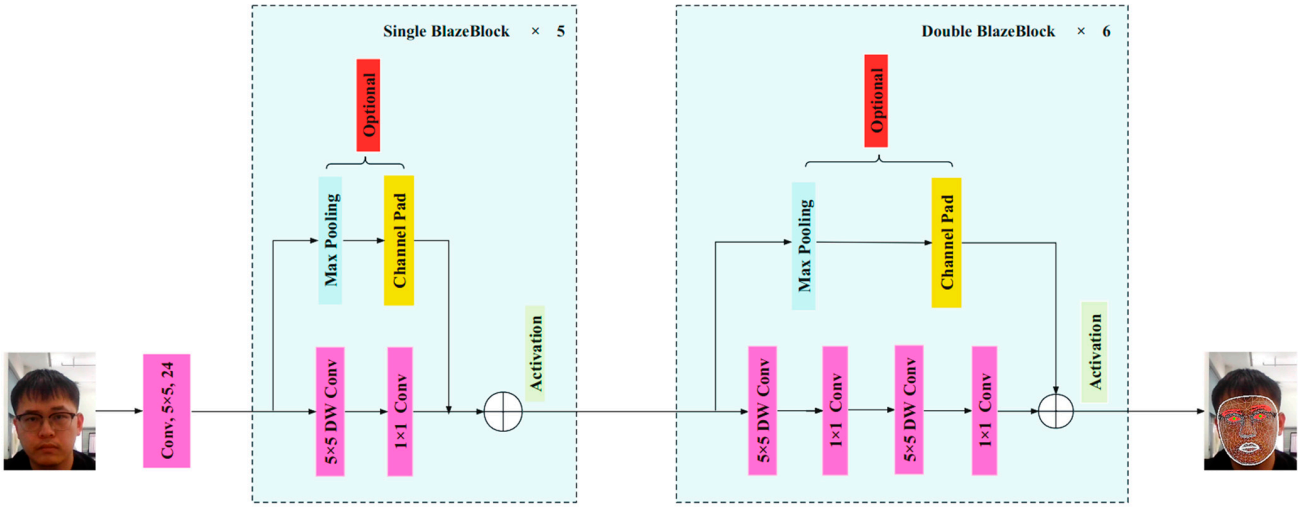


Figure 2. The structure of FFE.

As the network deepens and the features extracted become more advanced, the double BlazeBlock was designed on top of the single BlazeBlock. Compared to the Single BlazeBlock, it simply adds another 5 × 5 deep convolution and a 1 × 1 convolution to the main path. In the double BlazeBlock, the two single BlazeBlock modules are similar in composition to single BlazeBlock in that they both consist of a depthwise separable convolution layer and a skip connection. These modules perform convolutional operations on the input feature map to extract high-level features by using a depthwise separable convolutional layer. The skip connection, on the other hand, ensures that the model learns feature information at a shallower level and allows for a fine-grained combination of feature spectra. The two single BlazeBlock modules in double BlazeBlock are similar in composition to single BlazeBlock in that they both consist of a depth-separable convolutional layer and a skip connection. These modules perform convolutional operations on the input feature map to extract high-level features by using a deeply separable convolutional layer. The skip connection, on the other hand, ensures that the model learns feature information at a shallower level and allows for a fine-grained combination of feature spectra. In double

BlazeBlock, after processing by two single BlazeBlock modules, the feature map size and computational burden are then further reduced using the downsampling module. This allows the model to handle large-scale objects better, thus improving the accuracy of the network.

Therefore, the single BlazeBlock can be used for the shallow depth of the network, while the double BlazeBlock can be used for the deeper depth of the network. Through the aforementioned steps, facial feature information, including the feature areas for the left and right eyes, can be obtained.

3.2. ResNet50

Although designing deeper neural networks can lead to better recognition results in the process of deep network design, experiments indicate that as networks become increasingly deep, models will actually perform worse. While overfitting disturbance is eliminated, the real reason behind this problem comes from "vanishing gradients". Vanishing gradients are inherent defects in the backpropagation training algorithm. As the error is returned, gradients from earlier layers of the network will become increasingly smaller.

Equation (1) presents the loss function of the network, where X represents the input of the network, and W represents the weight parameters. The corresponding gradient values from backpropagation are given by Equation (2).

$$Loss = F(X, W) \tag{1}$$

$$\frac{\partial Loss}{\partial X} = \frac{\partial F(X, W)}{\partial X} \tag{2}$$

Furthermore, extending to multilayer networks, the loss function is given by Equation (3), where n denotes the number of layers in the network. According to the chain rule, the gradient of layer i can be derived, as shown in Equation (4). It can be observed that as the error is backpropagated, gradients from earlier layers of the network become progressively smaller.

$$Loss = F_n(X_n, W_n), L_n = F_{n-1}(X_{n-1}, W_{n-1}), \cdots, L_2 = F_1(X_1, W_1) \tag{3}$$

$$\frac{\partial Loss}{\partial X_i} = \frac{\partial F_n(X_n, W_n)}{\partial X_n} * \cdots * \frac{\partial F(X_{i+1}, W_{i+1})}{\partial X_i} \tag{4}$$

To address this issue, ResNet ingeniously introduced residual structures, as shown in Figure 3.

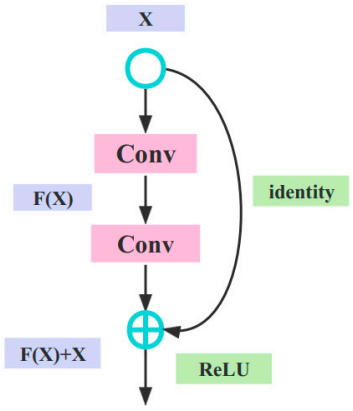


Figure 3. The residual block structure of ResNet.

This means that the output layer is $y = x + F(x)$, which implies that, in the process from Equation (4) to Equation (5), the gradient will not vanish even if the network becomes deeper.

$$\frac{\partial X_{i+1}}{\partial X_i} = \frac{\partial X_i + \partial F(X_i, W_i)}{\partial X_i} = 1 + \frac{\partial F(X_{i+1}, W_{i+1})}{\partial X_i} \tag{5}$$

The introduction of residual structures in ResNet allows for training neural networks that are deeper and more efficient, reducing the risk of overfitting while improving model performance. Specifically, there are several advantages to using residual structures: mitigate the problem of gradient disappearance, solve the problem of degradation, and avoid the problem of overfitting.

ResNet [35] has five different structures, namely ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152, representing different model depths and widths. The main differences between them are the depth and number of parameters of the models. ResNet-18 and ResNet-34 have the same basic structure and are relatively shallow networks. The basic structure of the last three models, ResNet-50, ResNet-101, and ResNet-152, is different from ResNet-18 and ResNet-34, representing deeper networks. Overall, the ResNet network structure has a large amount of computation and is complex, although the use of 1×1 convolution reduces the number of parameters. The depth of the network determines that the number of parameters is large. Therefore, choosing a network with too many layers may lead to overfitting and difficulty in convergence, while choosing a network with too few layers may result in poor training outcomes and low accuracy. Considering the above, in order to improve the accuracy of the network and prevent overfitting, ResNet50 is chosen as the backbone network in this paper. Its residual structure is shown in Figure 4.

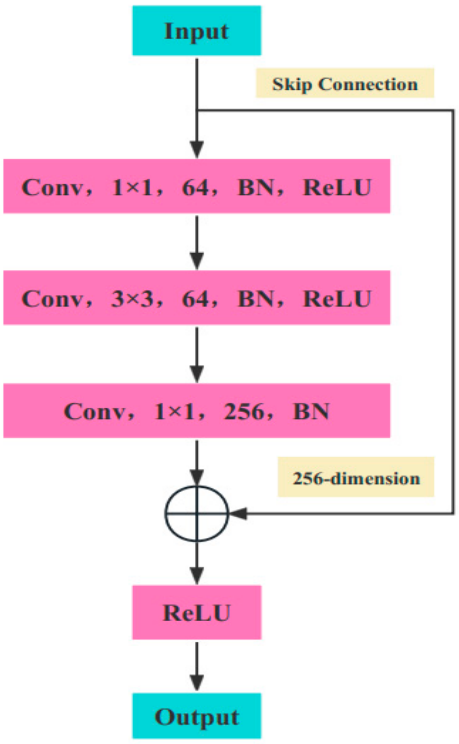


Figure 4. The residual structure of ResNet50.

This model comprises a main branch from top to bottom and skips connections on its right side. Its core idea is to transmit input information directly to the output via skip connections and to learn only the differences between the output and input through the main branch, thereby simplifying the learning process. In the main branch, 1×1 convolution is used to reduce the number of dimensions and generate 64 feature maps. Then, 3×3 convolution is performed to extract the main features. Subsequently, 1×1 convolu-

tion is executed to increase dimensionality and generate 256 feature maps. Additionally, batch normalization (BN) is utilized in ResNet to improve the model's generalization ability by subtracting the mean and dividing by the variance of the same batch of data. Furthermore, rectified linear units (ReLU), a nonlinear activation function, are employed to enhance the model's nonlinear adaptability.

Figure 5 shows the overall architecture of ResNet50, which adds branches between every two layers of the network and can sample input images with a convolutional sequence of 2. It is evident from the structure that there are four layers in the ResNet50 structure, and each layer is comprised of 3, 4, 6, and 3 residual blocks, respectively. The "3 4 6 3" in the basic block structure refers to the number of residual units in each layer, and these residual units are the basic subelements with interlayer connections. A large number of convolutional layers in ResNet are composed of these basic subelements arranged according to certain rules. Therefore, these numbers actually reflect the overall network depth of ResNet. The advantage of this design is that it enables ResNet to use the depth of the network more effectively and avoids the performance degradation problem that occurs when the number of layers increases in traditional deep networks.

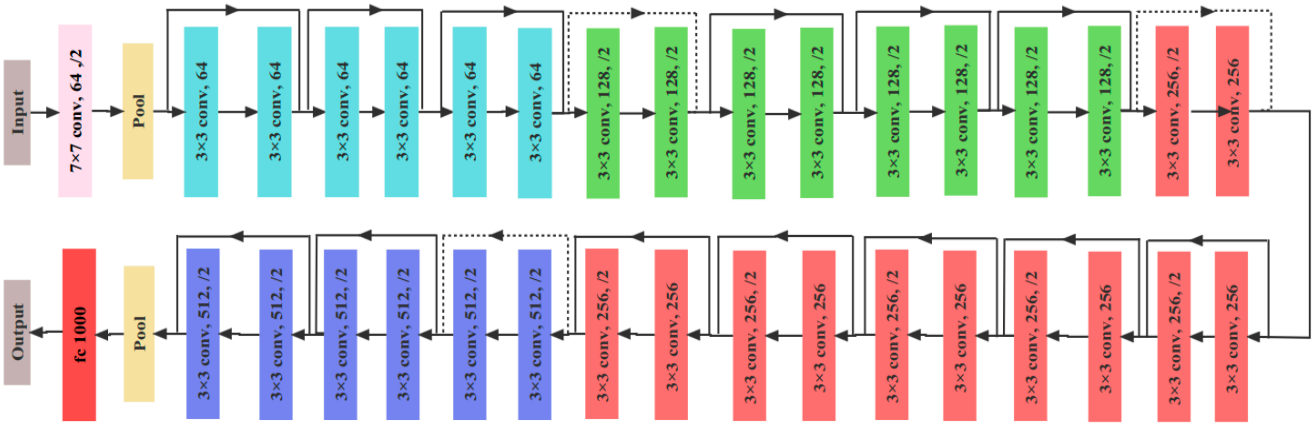


Figure 5. The overall architecture of ResNet50.

3.3. Attention Mechanism

As a data processing method in machine learning, the attention mechanism mainly focuses the network's learning on important areas. Studies have shown that embedding attention modules into convolutional neural networks can greatly improve network performance. In recent years, the method has been widely applied in computer vision, such as image classification, object detection, and semantic segmentation. Overall, attention mechanisms are mainly divided into channel attention and spatial attention; the most commonly used methods are the SE attention mechanism module [36] and the convolutional block attention module (CBAM) [37], respectively. The former has a simpler structure, consisting of two parts: squeeze and excitation. The main purpose of the squeeze part is to represent the importance of each channel feature, while the excitation part multiplies the feature map channels with the weighted channel importance obtained by the squeeze. Thus, it can make the model aware of the importance of each channel. However, the global average pooling of the squeeze module in the SE attention module is too simple to capture complex global information, and the fully connected layer of the excitation module increases model complexity, ignoring spatial information and leading to long calculation time and overfitting. The characteristic of the CBAM is that it focuses on the densest positions of effective information in an image, concatenates the feature maps generated by max pooling and average pooling, and applies convolutional and activation functions to form spatial attention maps. However, the CBAM has not captured spatial information at different scales to enrich the feature space, and its spatial attention only considers local information, failing to establish long-distance dependencies.

The PSA module not only has the ability to process tensors at multiple scales but also extracts spatial information at different scales by compressing the channel dimension of the input tensor. The PSA attention module [38] consists of four parts, as shown in Figure 6. First, the SPC module is used to obtain multiscale feature maps with channel diversity. Then, the SE weight module is used to obtain channel attention across multiple-scale feature maps. Next, the softmax function is used to adjust channel attention again and obtain the weight of multiple-scale channels. Finally, element-wise multiplication is applied to the recalibrated weights and corresponding feature maps. Based on the above four steps, multiscale feature information can be outputted. In this structure, the spatial information of the input feature maps is obtained by using a multibranch method, which can obtain rich position information and conduct parallel processing for multiple scales. Moreover, the pyramid structure generates different spatial resolutions and depths of multiscale convolution kernels, effectively extracting spatial information at different scales on each channel feature map. Therefore, by integrating multiscale spatial information and cross-channel attention into the original model, multiscale spatial information can be extracted at a more granular level.

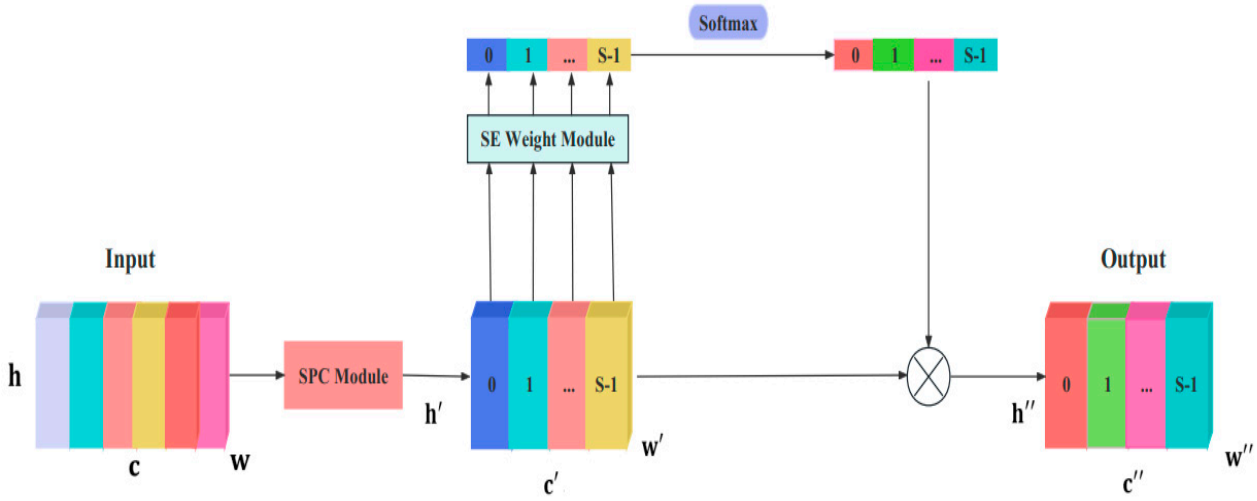


Figure 6. Structure of PSA module.

First, the SPC module is used to obtain feature maps with multiple channel scales. Although each input image in this structure has feature maps with different scales $\boldsymbol{F_i}$, they have the same channel dimension C^i. Their relationship is expressed as follows:

$$C^i = \frac{C}{S} \quad i = 0, 1, \ldots, m-1 \tag{6}$$

Each branch independently learns multiscale spatial information, allowing them to establish cross-channel interactions in a local manner. The number of parameters changes with the increase in kernel size. In order to handle input tensors with different kernel scales, group convolution is used and applied to the convolution kernel. The quantity n is the kernel size, and Z is the group size. The relationship between them is expressed as follows:

$$Z = 2^{\frac{n-1}{2}} \tag{7}$$

As a result, the generation function for multiscale feature map is expressed as follows:

$$H_i = Conv(n_i \times n_i, Z_i)(X) \quad i = 0, 1, 2 \cdots, m-1 \tag{8}$$

The preprocessed multiscale feature map is expressed as follows:

$$H = Cat([H_0, H_1, \cdots, H_{m-1}]) \tag{9}$$

Next, the SE weight module is used to obtain attention across multiple-scale feature maps in order to obtain channel attention. The weight information and weight vector for channel attention are obtained from the preprocessed multiscale feature maps. The attention weight vector is expressed as follows:

$$\boldsymbol{Q_i} = \boldsymbol{SEWeight}(\boldsymbol{H_i}),\ \boldsymbol{i} = 1, 2, 3, \cdots \boldsymbol{m} - 1 \tag{10}$$

The SEWeight module is used to obtain weights from input feature maps at different scales and then to fuse feature information of different scales. In addition, to achieve the interaction of feature information, the vectors across dimensions are fused. Therefore, the multiscale channel attention vector is obtained in the following way:

$$\boldsymbol{Q} = Q_0 \oplus Q_1 \oplus \cdots \oplus Q_{m-1} \tag{11}$$

Here, Q_i refers to the attention value of $\boldsymbol{H_i}$, and $\boldsymbol{Q}$ is the multiscale attention weight vector. Through the above steps, the interaction of local and global channel attention is achieved. Next, the channel attention vector can be obtained by fusing and concatenating channel attention. Its expression is written as follows:

$$Attention = Attention_0 \oplus Attention_1 \oplus \cdots \oplus Attention_{m-1} \tag{12}$$

Here, attention refers to the multiscale channel attention weight obtained after the interaction. Next, the weight of multiscale channel attention is multiplied by the feature map, expressed as follows:

$$T_i = F_i \odot Attention_i,\ i = 1, 2, 3, \cdots m - 1 \tag{13}$$

The above equation can effectively preserve the original feature map information. After simplification, the final input process is expressed as follows:

$$Out\ = Cat([T_0, T_1, \cdots, T_{m-1}]) \tag{14}$$

The PSA module is a novel attention mechanism that integrates multiscale spatial information and cross-channel attention into each block of feature groups. Thus, the PSA module can achieve better information interaction between local and global channel attention. Its advantages mainly include the following:

1. PSA has more refined weighting, which uses a more detailed monotonic regression algorithm to obtain the relative relationship between features, allowing it to more finely weight feature maps.

2. The PSA module can obtain the feature dependencies of each position in the feature map at different scales, thereby capturing the global and long-range features of the object better.

3. It can adapt to different resolutions: The PSA module adapts to different image inputs of varying resolutions by pyramid pooling, encoding different resolution feature maps through different sizes of pyramid layers, and then weighting them through corresponding convolution operations. This design improves the robustness of the model to different input resolutions.

The introduction of the PSA module can significantly improve the performance of the model in tasks such as object detection and image classification, even outperforming other powerful attention mechanisms. In Figure 7, the bottleneck module of ResNet consists of two 3×3 convolutions and one 1×1 convolution. The 1×1 convolution is mainly used for dimension reduction or expansion, and the 3×3 convolution is mainly used for feature extraction. The purpose of adding the attention module is to improve the feature extraction ability further, and thus, by simply replacing the 3×3 convolution of the bottleneck module with the PSA attention module, the EPSANet Bottleneck is obtained.

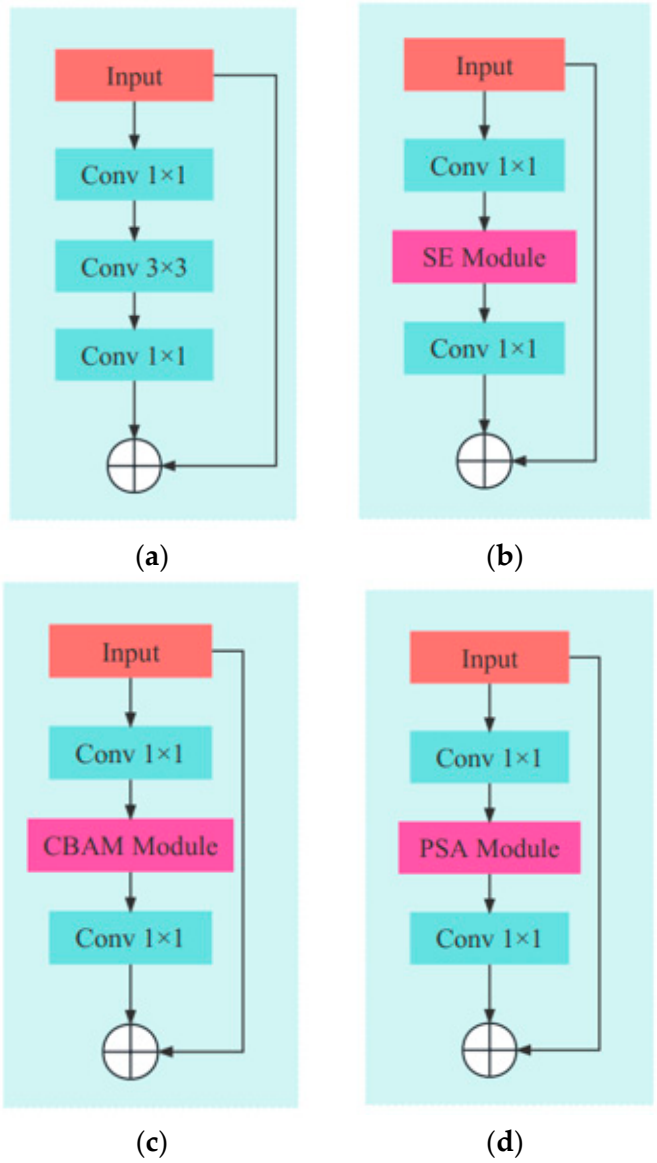


Figure 7. Comparison of bottleneck modules in different networks: (**a**) ResNet Bottleneck, (**b**) SENet Bottleneck, (**c**) CBAMNet Bottleneck, and (**d**) EPSANet Bottleneck.

Next, we adopt two identical loss functions for each gaze angle in this study. Each loss function is a linear combination of cross-entropy loss and mean squared error. The cross-entropy loss is defined as follows:

$$H(y,p) = -\sum_i \ y_i log p_i \tag{15}$$

The cross-entropy loss is defined as follows:

$$\boldsymbol{MSE}(\boldsymbol{y},\boldsymbol{p}) = \frac{1}{\boldsymbol{N}}\sum_0^N \ (\boldsymbol{y}-\boldsymbol{p})^2 \tag{16}$$

The proposed loss function for each gaze angle is a linear combination of the mean squared error and cross-entropy loss, defined as follows:

$$CLS(y,p) = H(y,p) + \alpha \cdot MSE(y,p) \tag{17}$$

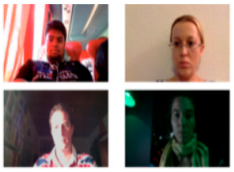
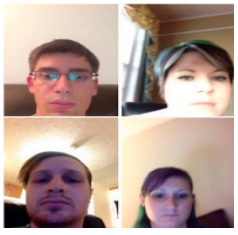
where CLS is the overall loss, p is the predicted value, y is the true value, and α is the regression coefficient.

4. Results

4.1. Selection of Datasets

The MPIIGaze [39] dataset contains 213,695 images of 15 subjects, each subject's facial images were collected at different times of the day with different mean gray intensity percentages of the facial regions. Forty-one dimensions were used to represent the subject gaze information in the dataset. The Gaze360 [40] dataset is the only dataset to obtain 3D gaze symbols of subjects in different indoor and outdoor environments, different lighting, different ages, different races, etc. The ETH-XGaze [41] dataset contains images of 110 subjects, consisting of over one million images of different gaze and extreme head postures. Eight cameras were used to capture the subjects' gaze postures, with an image resolution of up to 6000 × 4000 pixels, which is the highest image resolution dataset to date. The GazeCapture [42] dataset has 2,445,504 color images and contains 1450 2D gaze directions of subjects, which is the largest dataset to date. Table 1 gives the summary of the public datasets for gaze estimation.

Table 1. Summary of the dataset for gaze estimation.

Dataset	Source	Class	Number	Training Data	Test Data	Resolution	Samples
MPII Gaze	Max Planck Institute for Informatics	15	213,695	149,586	64,109	224 × 224	
Gaze360	Massachusetts Institute of Technology	238	172,000	120,400	51,600	3382 × 4096	
ETH-XGaze	ETH Zurich	110	1,532,658	1,072,861	459,797	6000 × 4000	
Gaze Capture	University of Georgia	1450	2,445,504	1,711,853	733,651	224 × 224	

4.2. Experimental Results and Analysis

The FPSA_L2CSNet, our novel network architecture, was implemented and evaluated on a cutting-edge hardware and software setup. The computational environment consisted of an i5-8300H CPU, 4 NVIDIA GeForce RTX2080Ti GPUs, 16 GB RAM, and leveraged CUDA version 10.2 for accelerated processing. The entire development process was conducted using the Python programming language. The network is trained using the Adam optimizer in the PyTorch framework on Ubuntu 20 with a learning rate of 0.00001. We trained our proposed network for 50 epochs using 64 batch sizes. Table 2 shows the comparison results of different methods on four datasets. It is found that the FPSA_L2CSNet proposed in this paper, which incorporates the PSA module, achieves better information interaction between local and global channel attention, and overcomes the lack of spatial information in the SE attention module and the inability of the CBAM to

establish long-range dependencies caused by only considering local information. Therefore, the final accuracy of FPSA_L2CSNet is significantly improved by 13.88%, 11.43%, and 7.34% compared to L2CSNet, FSE_L2CSNet, and FCBAM_L2CSNet, respectively.

Table 2. Comparison of gaze estimation results on four datasets.

Methods	MPIIGaze	Gaze360	ETH-XGaze	GazeCapture	Average Error
ITracker	5.57°	4.89°	10.56°	9.56°	7.65°
FullFace [43]	4.82°	4.32°	9.56°	9.32°	7.01°
DialatedNet [44]	4.68°	5.21°	10.43°	8.47°	7.19°
RTGene [45]	4.78°	5.39°	10.23°	8.98°	7.35°
FARNet [46]	4.29°	4.68°	9.87°	9.54°	7.09°
CANet [47]	4.09°	3.93°	8.97°	9.32°	6.58°
FAZE [48]	3.14°	4.54°	5.26°	3.01°	3.99°
L2CSNet	3.92°	3.89°	8.02°	6.67°	5.63°
FSE_L2CSNet	3.85°	3.52°	6.04°	6.83°	5.06°
FCBAM_L2CSNet	3.68°	3.45°	4.79°	4.79°	4.18°
FPSA_L2CSNet	3.41°	3.35°	3.73°	3.74°	3.56°

In particular, we conducted a comprehensive comparison with the state-of-the-art gaze estimators currently available. Our results revealed that on the MPIIGaze and GazeCapture datasets, the employment of meta-learning trained gaze estimation methods in FAZE led to a slightly lower error compared to the model proposed in this paper. However, upon evaluating FAZE on the Gaze360 and ETH-XGaze datasets, we observed a significantly higher error as opposed to our proposed model.

This disparity in performance can be attributed to the diversity of participants in the Gaze360 and ETH-XGaze datasets, hailing from various regions worldwide, including the United States, the Middle East, Europe, Africa, and Asia. In contrast, the MPIIGaze and GazeCapture datasets predominantly consist of participants from the United States and Europe. Consequently, the FAZE model's limited generalization ability and reliance on a relatively singular dataset structure hinder its adaptability in diverse regions.

The resulting error in the Gaze360 dataset was measured at 4.54°, while our model's error, as proposed in this article, stood at 3.35°. Similarly, the ETH-XGaze dataset yielded an error of 5.26° for FAZE, whereas our model demonstrated an error of 3.73°. Overall, across all four datasets, FAZE exhibited an average error of 3.99°, while our model showcased an average error of 3.56°. Evidently, the method presented in this paper exhibits superior adaptability to public test datasets, showcasing robustness, universality, and stability.

To obtain satisfactory results, this study compares the FPSA_L2CSNet model with L2CSNet, FSE_L2CSNet, and FCBAM_L2CSNet models. The results obtained by testing on the MPIIGaze, Gaze360, ETH-Xgaze, and GazeCapture datasets are shown in Figure 8. In Figure 8a, it can be seen that on the MPIIGaze dataset, due to the addition of the PSA module in the FPSA_L2CSNet network, the input image receives better information interaction between local and global channel attention and overcomes the lack of spatial information of the SE attention module and the inability of the CBAM to establish long-range dependencies caused by only considering local information. Therefore, the network is in a convergent state at the 10th epoch with the lowest average error.

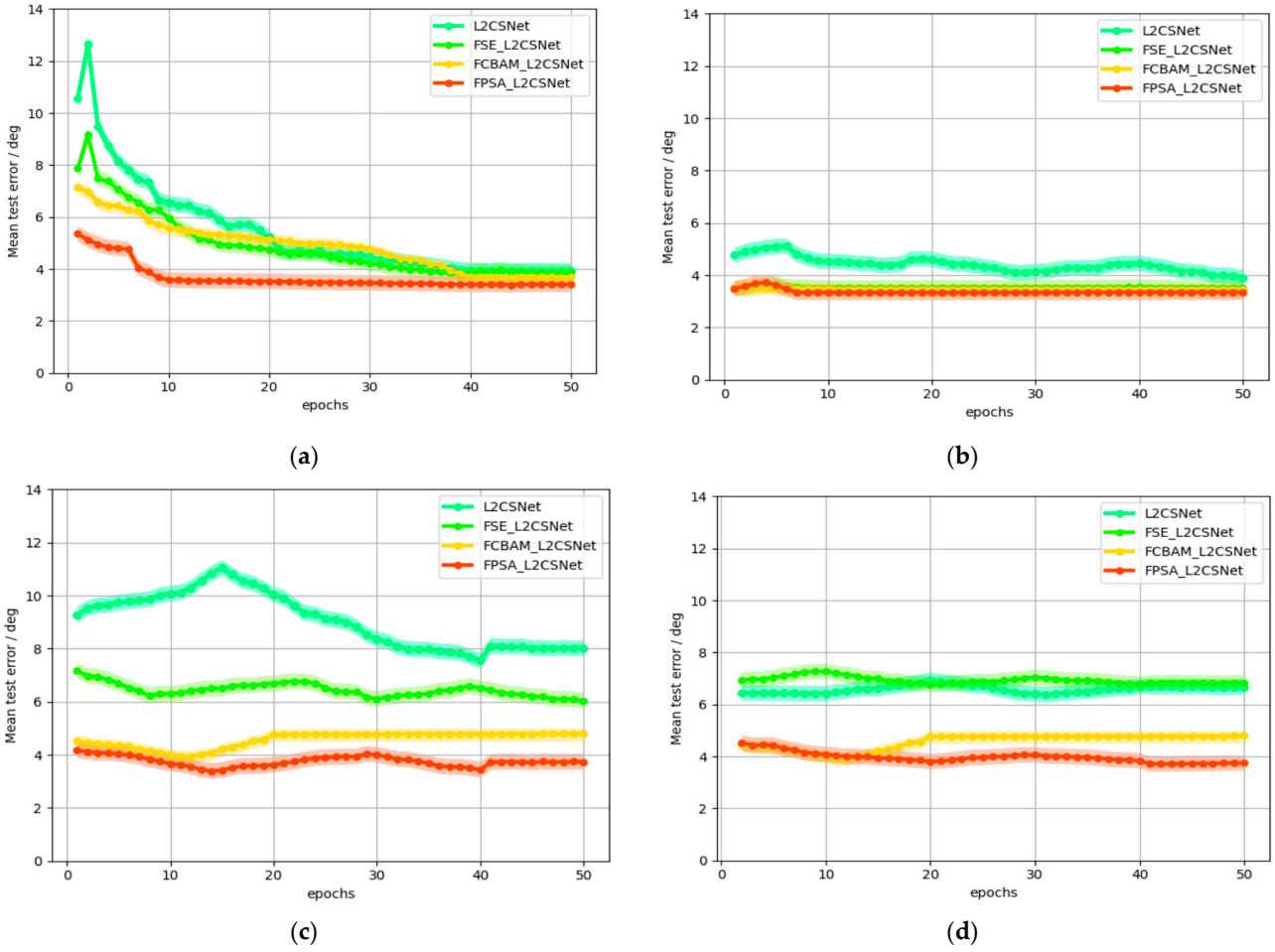


Figure 8. The comparison of gaze estimation accuracy on four datasets: (**a**) MPIIGaze, (**b**) Gaze360, (**c**) ETH-XGaze, and (**d**) GazeCapture.

In Figure 8b, on the Gaze360 dataset, since the FPSA_L2CSNet, FSE_L2CSNet, and FCBAM_L2CSNet added attention modules to the original network, the convergence of the three models is achieved at the 8th epoch, while L2CSNet gradually converges at the 45th epoch. The convergence speed and error of the former three are better than those of L2CSNet. In Figure 8c, on the ETH-XGaze dataset, because there is no added attention module, the oscillations and fluctuations of L2CSNet are the largest, and its error is also the largest. The FSE_L2CSNet ignores spatial information, leading to a large error, while the error of FCBAM_L2CSNet increases after decreasing because the accuracy decreases after saturation as the network depth increases. The architecture of FPSA_L2CSNet is better than the other three, so it has the lowest error. In Figure 8d, on the GazeCapture dataset, the error of FSE_L2CSNet is basically the same as that of L2CSNet because the feature information extraction is insufficient after spatial information is ignored. FCBAMNet has the problem of having too deep a network, leading to a rise in error after decreasing, while FPSA_L2CSNet has the lowest error.

In the testing phase, this article will test the above four methods on the four public datasets. The performance of the four methods varies on different datasets. In this paper, a thermodynamic diagram is used to analyze the error, and the results of the MPIIGaze dataset are shown in Figure 9. The horizontal axis represents the yaw angle, while the vertical axis represents the pitch angle. As the color changes from cyan to red, the error gradually increases. Therefore, the error size of this method is measured by calculating the proportion of cyan areas in the total area. The more cyan areas, the smaller the error, and vice versa. The specific results are shown in Table 3. From the table, it can be seen

that in the four datasets, FPSA_ L2CSNet has the highest proportion of cyan areas and the smallest error.

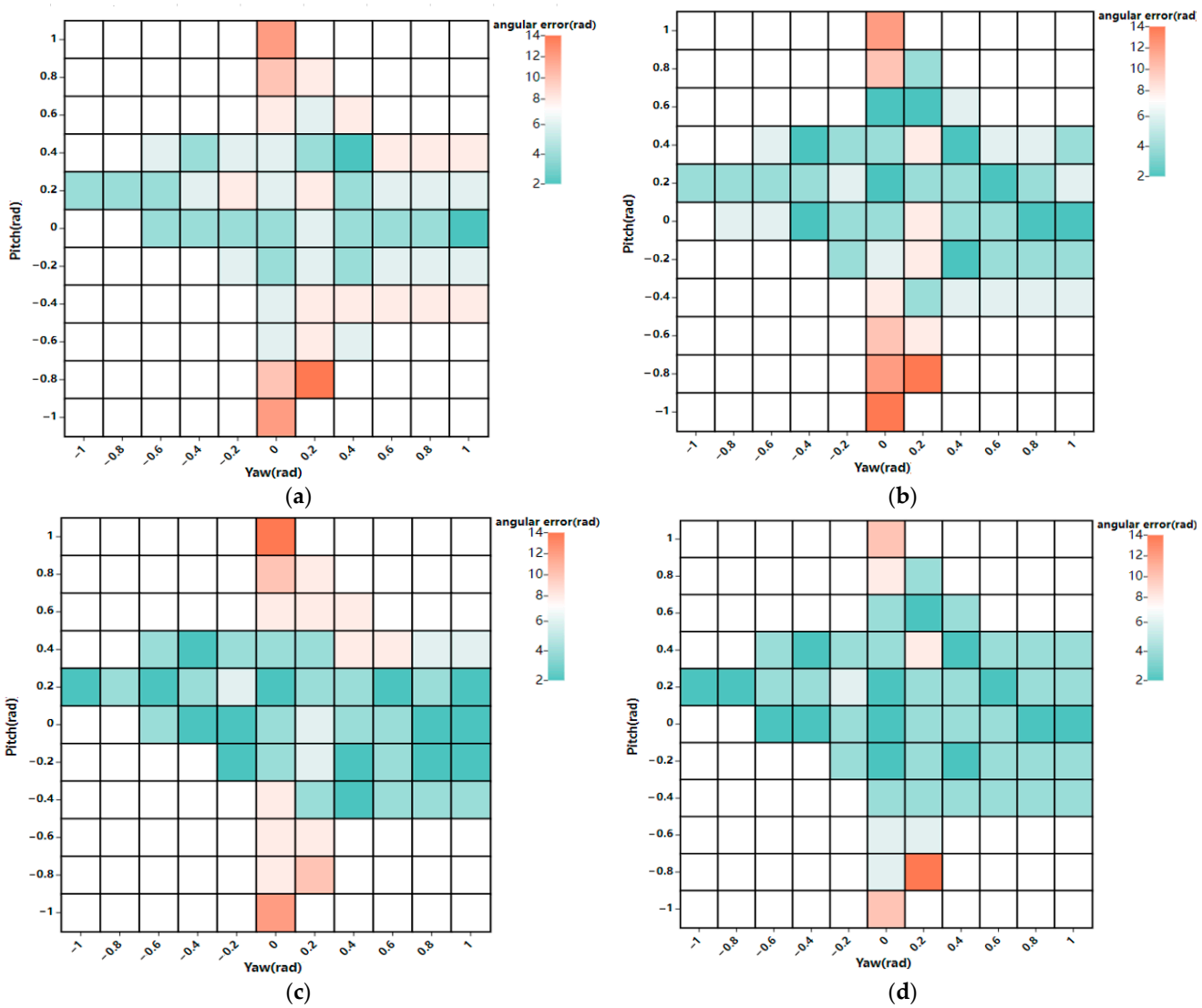


Figure 9. The comparison of gaze estimation error on four datasets: (**a**) L2CSNet, (**b**) FSE_L2CSNet, (**c**) FCBAM_L2CSNet, and (**d**) FPSA_L2CSNet.

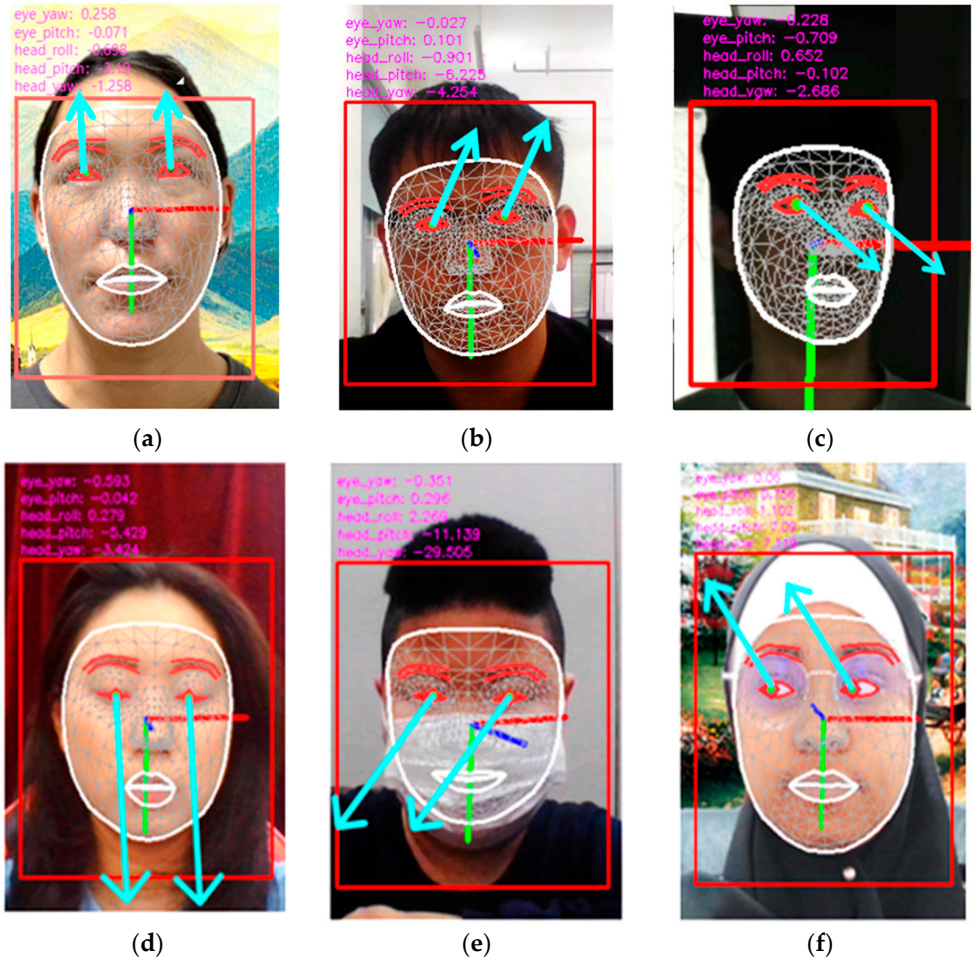
Table 3. The accuracy comparison results of different gaze estimation methods.

Methods	MPIIGaze	Gaze360	ETH-XGaze	GazeCapture
L2CSNet	59.65%	63.98%	60.49%	53.62%
FSE_L2CSNet	65.35%	68.26%	66.25%	65.12%
FCBAM_L2CSNet	76.56%	73.45%	69.23%	70.25%
FPSA_L2CSNet	87.28%	83.64%	74.36%	75.95%

Table 3 gives the accuracy comparison results of different gaze estimation methods. It can be seen from the results that the SE attention mechanism may excessively pursue the weighting of useful channels and ignore the information of low-level channels, resulting in overfitting of the model, loss of generalization ability, and decline in accuracy. The CBAM mechanism requires adjustment factors for learning channel attention and spatial attention, leading to an increase in the number of parameters and a decrease in the learning rate, which invisibly affects the accuracy of the model. The method proposed in this article comprehensively measures local and global attention mechanisms, which can help the

model better understand input data while paying attention to features at different scales and synthesizing their information to generate total attention, reducing unnecessary errors caused by fluctuations in information at a single scale, improving the robustness of the model, and using features at different scales to help the model better understand input data. In summary, FPSA_L2CSNet has the highest gaze accuracy and the strongest performance on all four datasets, with the smallest error on the four public datasets.

Finally, this proposed gaze estimation model selects real individuals of different backgrounds, lighting, nationalities, skin tones, ages, partial occlusions, and genders to gaze in different directions in the lab to prove progressiveness and effectiveness. Among them, in complex background conditions, the model can accurately obtain the gaze direction when the testers with obvious differences in skin tone and from different countries in images. For example, the girl in Figure 10a comes from Mongolia with yellow skin, and Figure 10f comes from Madagascar with black skin. For testers under bright lighting conditions and gazing in different directions, the model can accurately estimate the gaze direction as seen in Figure 10b,d. Even when the tester in image Figure 10c is under the dimmest lighting conditions compared to other testers, the model still accurately estimates the gaze direction. Moreover, even when the facial features of the tester in image Figure 10e are partially occluded and the gaze is to the lower left, the accuracy of the estimated gaze direction does not exhibit significant deviation due to the precise extraction of facial feature points. In summary, the robustness, universality, and accuracy of this model have been fully validated.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 10. The results of gaze estimation on real datasets: (**a**) upper; (**b**) upper right; (**c**) lower right; (**d**) lower; (**e**) lower left; (**f**) upper left.

5. Conclusions

To address the challenge of low gaze estimation accuracy among individuals in different environments, this study proposes a gaze estimation model based on attention mechanisms: FPSA_L2CSNet. The results obtained from testing on the public dataset show that this model exhibits the lowest error and highest accuracy on the MPIIGaze dataset, with errors of 3.41°, 3.35°, 3.73°, and 3.74°, respectively. Compared with L2CSNet, FSE_L2CSNet, and FCBAM_L2CSNet, the accuracy is improved by 13.88%, 11.43%, and 7.34%, respectively. The results obtained from testing on real-world data show that this proposed model achieves precise gaze estimation under different lighting, background, and partial occlusion conditions for individuals of different nationalities, skin tones, ages, and genders, thus verifying the robustness and universality of the model.

In the future, PSA's attention mechanism could be further enhanced to amalgamate diverse levels of gaze estimation features, which would enable network models to concentrate on line-of-sight information, thereby improving the resilience and precision of the model.

Author Contributions: Conceptualization, J.W., X.Y., Q.W., Y.I., H.W. and A.W.; methodology, J.W. and X.Y.; software, J.W.; validation J.W.; writing—review and editing X.Y., Q.W., Y.I., H.W. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the high-end foreign experts' introduction program (G2022012010L) and Major Science and Technology Projects of Zhongshan City in 2022 (2022A1020).

Data Availability Statement: MPII Gaze: http://datasets.d2.mpi-inf.mpg.de/MPIIGaze/MPIIGaze.tar.gz; Gaze360: http://gaze360.csail.mit.edu/download.php; ETH-XGaze: https://ait.ethz.ch/xgaze?query=eth; GazeCapture: https://gazecapture.csail.mit.edu/download.php.

Acknowledgments: Qing Wu acknowledges the National Natural Science Foundation of China (Grant No. 62205091), the China Postdoctoral Science Foundation Funded Project (Grant No. 2022M710983), and HeiLongJiang Postdoctoral Foundation (Grant No. LBH-Z22201). The study was supported by the Fundamental Research Foundation for Universities of Heilongjiang Province (2022-KYYWF-0121).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bulling, A.; Gellersen, H. Toward Mobile Eye-Based Human-Computer Interaction. *IEEE Pervasive Comput.* **2010**, *9*, 8–12. [CrossRef]
2. Drewes, H. Eye gaze Tracking for Human Computer Interaction. Ph.D. Thesis, LMU, Munich, Germany, 2010; pp. 9–16.
3. Patney, A.; Kim, J.; Salvi, M. Perceptually-based foveated virtual reality. In Proceedings of the ACM SIGGRAPH 2016 Emerging Technologies, Anaheim, CA, USA, 24–28 July 2016; pp. 1–2.
4. Novak, D.; Riener, R. Enhancing patient freedom in rehabilitation robotics using gaze-based intention detection. In Proceedings of the IEEE International Conference on Rehabilitation Robotics, Seattle, WA, USA, 24–26 June 2013; pp. 123–126.
5. Atkins, M.S.; Tien, G.; Khan, R.S.A.; Meneghetti, A.; Zheng, B. What do surgeons see: Capturing and synchronizing eye gaze for surgery applications. *Surg. Innov.* **2013**, *6*, 14–21. [CrossRef] [PubMed]
6. Eggert, T. Eye movement recordings: Methods. *Neuro-Ophthalmology* **2007**, *40*, 15–34.
7. Zu-Tao, Z.; Jia-Shu, Z. Sampling strong tracking nonlinear unscented Kalman filter and its application in eye tracking. *Chin. Phys. B* **2021**, *6*, 89–95. [CrossRef]
8. Zhang, J.; Zhang, Z. Application of a strong tracking finite-difference extended kalman filter to eye tracking. In Proceedings of the Intelligent Computing: International Conference on Intelligent Computing, Kunming, China, 16–19 August 2006; pp. 65–68.
9. Amudha, J.; Chandrika, K.R. Suitability of Genetic Algorithm and Particle Swarm Optimization for Eye Tracking System. 6th International Conference on Advanced Computing (IACC). *IEEE Access* **2016**, *23*, 165–168.
10. Yu, L.; Xu, J.; Huang, S. Eye-gaze tracking system based on particle swarm optimization and BP neural network. In Proceedings of the 12th World Congress on Intelligent Control and Automation (WCICA), Guilin, China, 12–15 June 2016; pp. 1269–1273.
11. Yan, B.; Tang, X. Gaze Estimation Based on Difference Residual Network. In Proceedings of the2021 International Conference on Computer Information Science and Artificial Intelligence (CISAI), Kunming, China, 17–19 September 2021; pp. 651–655.
12. Zhang, H.; Wang, X.; Ren, W.; Noack, B.R.; Liu, H. Improving the Reliability of Gaze Estimation through Cross-dataset Multi-task Learning. In Proceedings of the 2022 International Conference on High Performance Big Data and Intelligent Systems, Tianjin, China, 10–11 December 2022; pp. 202–206.

13. Kim, S.H.; Lee, D.J.; Kim, D.H.; Song, B.C. Continuous Engagement Estimation based on Gaze Estimation and Facial Expression Recognition. In Proceedings of the 2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Phuket, Thailand, 5–8 July 2022; pp. 937–939.
14. Liu, M.; Li, Y.; Liu, H. 3D Gaze Estimation for Head-Mounted Eye Tracking System with Auto-Calibration Method. *IEEE Access* **2020**, *8*, 104207–104215. [CrossRef]
15. Ma, J.; Zhang, X.; Wu, Y.; Hedau, V.; Chang, S.-F. Few-Shot Gaze Estimation with Model Offset Predictors. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 4893–4897.
16. Zhuang, J.; Wang, C. Attention Mechanism Based Full-face Gaze Estimation for Human-computer Interaction. In Proceedings of the 2022 International Conference on Computer Network, Electronic and Automation (ICCNEA), Xi'an, China, 23–25 September 2022; pp. 6–10.
17. Saha, B.; Islam, M.J.; Dipto, A.S.; Mostaque, S.K. An Efficient Approach for Appearance Based Eye Gaze Estimation with 13 Directional Points. In Proceedings of the 2021 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering, Rajshahi, Bangladesh, 26–27 December 2021; pp. 1–5.
18. Chang, C.C.; Ou, W.L.; Chen, H.L. Detection of Facial Directions and Features With YOLO-Based Deep-Learning Technology for Pre-Processing of Gaze Estimation. In Proceedings of the 2022 IEEE 11th Global Conference on Consumer Electronics (GCCE), Osaka, Japan, 18–21 October 2022; pp. 284–285.
19. Wan, Z.; Xiong, C.; Chen, W.; Zhang, H.; Wu, S. Pupil-Contour-Based Gaze Estimation With Real Pupil Axes for Head-Mounted Eye Tracking. *IEEE Trans. Ind. Inform.* **2022**, *18*, 3640–3650. [CrossRef]
20. Huang, H.; Ren, L.; Yang, Z.; Zhan, Y.; Zhang, Q.; Lv, J. GAZEATTENTIONNET: Gaze Estimation with Attentions. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing, Singapore, 23–27 May 2022; pp. 2435–2439.
21. Zichen, Z.; Lai, W.; Xiaofeng, L.; Zhi, L. Monocular Gaze Estimation Network based on Mixed Attention. In Proceedings of the 2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA), Changchun, China, 20–22 May 2022; pp. 377–380.
22. Wan, Z.; Xiong, C.; Li, Q. Accurate Regression-Based 3D Gaze Estimation Using Multiple Mapping Surfaces. *IEEE Access* **2020**, *8*, 166460–166471. [CrossRef]
23. Zhuang, Y.; Zhang, Y.; Zhao, H. Appearance-based gaze estimation using separable convolution neural networks. In Proceedings of the 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 12–14 March 2021; pp. 609–612.
24. Zhang, Y.; Sun, J. Research and Application of Gaze Data Processing Method. In Proceedings of the 2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 17–19 June 2022; pp. 885–890.
25. Murthy, L.R.; Biswas, P. Deep Learning-based Eye Gaze Estimation for Military Aviation. In Proceedings of the 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 5–12 March 2022; pp. 1–8.
26. Liu, G.; Yu, Y.; Mora, K.A.F.; Odobez, J.-M. A Differential Approach for Gaze Estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *43*, 1092–1099. [CrossRef] [PubMed]
27. Suzuki, G.; Takahashi, S.; Ogawa, T.; Haseyama, M. An Estimation Method of Candidate Region for Superimposing Information Based on Gaze Tracking Data in Soccer Videos. In Proceedings of the 2020 IEEE International Conference on Consumer Electronics—Taiwan (ICCE-Taiwan), Taoyuan, Taiwan, 28–30 September 2020; pp. 1–2.
28. Chang, C.C.; Ou, W.L.; Chen, H.L. YOLO-Based Deep-Learning Gaze Estimation Technology by Combining Geometric Feature and Appearance Based Technologies for Smart Advertising Displays. In Proceedings of the 2022 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE), Tainan, Taiwan, 7–10 November 2022; pp. 1–3.
29. Senarath, S.; Pathirana, P.; Meedeniya, D.; Jayarathna, S. Customer Gaze Estimation in Retail Using Deep Learning. *IEEE Access* **2022**, *10*, 64904–64919. [CrossRef]
30. Luo, Y.; Chen, J. CI-Net: Appearance-Based Gaze Estimation via Cooperative Network. *IEEE Access* **2022**, *10*, 78739–78746. [CrossRef]
31. Han, S.Y.; Cho, N.I. User-Independent Gaze Estimation by Extracting Pupil Parameter and Its Mapping to the Gaze Angle. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 1993–2000.
32. Song, X.; Guo, S.; Yu, Z.; Dong, J. An Encoder-Decoder Network with Residual and Attention Blocks for Full-Face 3D Gaze Estimation. In Proceedings of the 2022 7th International Conference on Image, Vision and Computing (ICIVC), Xi'an, China, 26–28 July 2022; pp. 713–717.
33. Zhao, K.; Hu, Z.; Zhang, Q.; Liu, J. RTHG: Towards Real-Time Head Detection And Gaze Estimation. In Proceedings of the 2022 IEEE International Conference on Robotics and Biomimetics, Jinghong, China, 5–9 December 2022; pp. 735–740.
34. Abdelrahman, A.A.; Hempel, T.; Khalifa, A. L2CS-Net: Fine-Grained Gaze Estimation in Unconstrained Environment. *arXiv* **2022**, arXiv:2203.03339.
35. He, K.; Zhang, X.; Ren, S. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

36. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 56–65.
37. Woo, S.; Park, J.; Lee, J.Y. CBAM: Convolutional block attention module. In Proceedings of the Computer Vision-ECCV, Munich, Germany, 8–14 September 2018; pp. 3–19.
38. Zhang, H.; Zu, K.; Lu, J. EPSANet: An Efficient Pyramid Squeeze Attention Block on Convolutional Neural Network. In Proceedings of the Asian Conference on Computer Vision, Macau, China, 4–8 December 2022; pp. 1161–1177.
39. Zhang, X.; Sugano, Y.; Fritz, M.; Bulling, A. Appearance-based Gaze Estimation in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4511–4520.
40. Kellnhofer, P.; Recasens, A.; Stent, S. Gaze360: Physically Unconstrained Gaze Estimation in the Wild. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2020; pp. 195–198.
41. Zhang, X.; Park, S.; Beeler, T. ETH-XGaze: A Large Scale Dataset for Gaze Estimation under Extreme Head Pose and Gaze Variation. In Proceedings of the Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 156–159.
42. Krafka, K. Eye Tracking for Everyone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2176–2184.
43. Zhang, X.; Sugano, Y.; Fritz, M.; Bulling, A. It's written all over your face: Fullface appearance-based gaze estimation. In Proceedings of the Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 2299–2308.
44. Chen, Z.; Shi, B.E. Appearance-based gaze estimation using dilated-convolutions. In Proceedings of the Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018; pp. 309–324.
45. Fischer, T.; Chang, H.J.; Demiris, Y. Rt-gene: Real-time eye gaze estimation in natural environments. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 334–352.
46. Cheng, Y.; Zhang, X.; Lu, F.; Sato, Y. Gaze estimation by exploring two-eye asymmetry. *IEEE Trans. Image Process.* **2020**, *29*, 5259–5272. [CrossRef] [PubMed]
47. Cheng, Y.; Huang, S.; Wang, F.; Qian, C.; Lu, F. A coarse-to-fine adaptive network for appearance-based gaze estimation. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 10623–10630.
48. Park, S.; Molchanov, P.; Kautz, J. Few-Shot Adaptive Gaze Estimation. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 9367–9376.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Quality of Life Prediction on Walking Scenes Using Deep Neural Networks and Performance Improvement Using Knowledge Distillation

Thanasit Rithanasophon [1], Kitsaphon Thitisiriwech [1], Pittipol Kantavat [1,*], Boonserm Kijsirikul [1], Yuji Iwahori [2], Shinji Fukui [3], Kazuki Nakamura [4] and Yoshitsugu Hayashi [5]

[1] Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Pathumwan, Bangkok 10330, Thailand; 6270111921@student.chula.ac.th (T.R.); ck.kitsaphon@gmail.com (K.T.); boonserm.k@chula.ac.th (B.K.)
[2] Department of Computer Science, Chubu University, Kasugai 487-8501, Japan; iwahori@isc.chubu.ac.jp
[3] Faculty of Education, Aichi University of Education, Kariya 448-8542, Japan; sfukui@auecc.aichi-edu.ac.jp
[4] Department of Civil Engineering, Meijo University, Nagoya 468-8502, Japan; knaka@meijo-u.ac.jp
[5] Center for Sustainable Development and Global Smart City, Chubu University, Kasugai 487-8501, Japan; y-hayashi@isc.chubu.ac.jp
* Correspondence: pittipol.k@chula.ac.th

Abstract: The well-being of residents is a top priority for megacities, which is why urban design and sustainable development are crucial topics. Quality of Life (QoL) is used as an effective key performance index (KPI) to measure the efficiency of a city plan's quantity and quality factors. For city dwellers, QoL for pedestrians is also significant. The walkability concept evaluates and analyzes the QoL in a walking scene. However, the traditional questionnaire survey approach is costly, time-consuming, and limited in its evaluation area. To overcome these limitations, the paper proposes using artificial intelligence (AI) technology to evaluate walkability data collected through a questionnaire survey using virtual reality (VR) tools. The proposed method involves knowledge extraction using deep convolutional neural networks (DCNNs) for information extraction and deep learning (DL) models to infer QoL scores. Knowledge distillation (KD) is also applied to reduce the model size and improve real-time performance. The experiment results demonstrate that the proposed approach is practical and can be considered an alternative method for acquiring QoL.

Keywords: quality of life; walking scene; walkability; semantic segmentation; object detection; deep convolutional neural networks; knowledge distillation

1. Introduction

Urban design is crucial for sustainable development in megacities. Fast-growing megacities without good urban planning can cause a lot of significant problems to residents such as traffic congestion and transportation accidents [1–5]. For this reason, policymakers need to realize the importance of sustainable development in urban design and plan both resident-friendly and environment-friendly cities.

Policymakers can use Quality of Life (QoL), one of the key performance indexes (KPI), to evaluate their urban design performance [6]. The concept of QoL [7] is to assess well-being, happiness of citizens, and individual needs including social interactions as a part of the society based on sustainable development [8–10]. According to numerous studies, people use the Quality-of-Life framework as a policy performance measurement for urban public transportation of megacities, especially in emerging countries, for example, Bangkok. The failed urban planning in Bangkok leads to severe transportation congestion that is caused by the car-oriented policies [1,2,11]. Thus, many researchers use Bangkok as a case of study for QoL, such as Alonso [12], who predicts land-use public transportation policy based on the satisfaction of dwellers, and Hayashi et al. [13] and Banister [14] proposed

that people should apply a QoL approach by implementing less car-oriented policies and encouraging the use of public transportation more.

The QoL of pedestrians is also a significant factor in the well-being of people living in the cities. Many studies propose the walkability concept to analyze and evaluate the QoL in a walking scene. Vichiensan and Nakamura [15] compare walking needs in Bangkok and Nagoya. The study highlights the importance of considering cultural and environmental factors when designing urban spaces that promote walking and active transportation. Nakamura [16] studies the correlation between walkability and QoL outcomes in car-oriented Asian cities. Vichiensan et al. [17] conclude that walking is essential to provide equitable access and mobility in a city. Sou et al. [18] developed a framework for evaluating street space by considering human emotions and values, aiming to improve communication between designers and stakeholders. Nakamura [19,20] highlights the importance of individual functions in pedestrian spaces and the effectiveness of design elements and proposes using virtual reality (VR) evaluation to improve the design process and create more effective pedestrian spaces.

Traditional tools used to evaluate Quality of Life (QoL) typically involve questionnaire-based surveys, which can be time-consuming and costly. However, the results obtained from these surveys are often limited to specific areas and timeframes, thus restricting their generalizability. Furthermore, integrating traditional QoL evaluation methods into other IT systems can present challenges and complexities. Some studies promoted new approaches to QoL by applying artificial intelligence (AI) to solve these limitations. Kantavat et al. [21] implemented deep convolutional neural networks (DCNNs), semantic segmentation, and object detection for extracting factors in transportation mobility from an image. Thitisiriwech et al. [22] proposed the Bangkok Urbanscapes dataset, the first labeled urban scene dataset in Bangkok, and models that have an excellent performance on semantic segmentation processing. Thitisiriwech et al. [23] proposed an AI approach using DCNNs and a linear regression model to extract information from images and infer the QoL score. Iamtrakul et al. [24] investigated how the built environment affects the QoL related to transportation using a combination of GIS and DL in the Sukhumvit district, Bangkok.

This study proposes a QoL evaluation method using the machine learning approach. There are two steps in the QoL assessment process. First, we conduct the information extraction using image processing, consisting of semantic segmentation and object recognition. Second, we feed the extracted information into the trained model to perform the QoL inference. In addition, we also propose a knowledge distillation (KD) framework to shorten the QoL inference time. The experiment results show that our proposed system is effective and can be considered in QoL evaluation for the walking scene.

The proposed novel method for evaluating Quality of Life (QoL) offers several distinct advantages compared to traditional approaches. One key advantage is the significant reduction in time and cost requirements. Unlike the conventional method, which is limited to specific survey areas and timeframes, our approach can be externally applied to evaluate QoL in different regions and at different points in time. Moreover, the proposed method is easily integrable into other IT systems that necessitate QoL evaluation or simulations.

Unlike previous research primarily concentrating on driving scenarios mentioned in the introduction, our study focuses on assessing QoL in the walking mode. This emphasis allows us to tailor our methodology to understand pedestrians' experiences better by considering that pedestrians observe and interact with their environment. We leverage virtual reality (VR) tools in the QoL questionnaire process to create a more immersive and realistic assessment. This enables us to capture the panoramic surroundings and the active engagement of pedestrians.

2. Literature Review

Many studies of QoL for the walking scene suggest the factors affecting walkability. Vichiensan and Nakamura [15] compared walking needs in Bangkok and Nagoya, finding that comfort and pleasurability are higher-level needs in both cities. Safety was

considered a higher-level need in Bangkok due to poorer street conditions. The study suggested that street improvements were needed to encourage walking in Bangkok and recover lost street activities in Nagoya. The study's inclusion of informal activities in the walkability evaluation provided practical street design insights for growing Asian cities. Nakamura [16] explored the relationship between walkability and QoL outcomes, particularly in car-oriented Asian cities. The study defined indicators of walkability QoL and surveyed 500 inhabitants of Nagoya city (Japan) using a questionnaire to assess their neighborhoods. Results showed that neighborhood street quality, particularly pleasurability, significantly affected QoL outcomes through their interrelationship. The study suggested that street quality was vital in land-use transport planning to improve QoL outcomes. Vichiensan et al. [17] summarized that walking was an important and sustainable mode of transportation that can provide equitable access and mobility in a city. However, to be enjoyable, convenient, and affordable, the walkway and nearby street must be attractive, vibrant, secure, uninterrupted, and well-protected from road traffic.

Some studies apply modern technology, such as AI and VR, for QoL and walkability. Sou et al. [18] proposed an AI and human co-operative evaluation (AIHCE) framework that facilitated communication design between designers and stakeholders based on human emotions and values for evaluating street space. The study suggested that the proposed framework can contribute to fostering people's awareness of streets as public goods, reflecting the essential functions of public spaces and the residents' values and regional characteristics, improving the city's sustainability. Nakamura [19,20] addressed the issue of pedestrian spaces needing more focus on individual functions and the effectiveness of design elements. The researcher proposed using VR evaluation to analyze the impact of sidewalk boundary space design on pedestrian space. The survey results showed a close relationship between boundary space design and pedestrian needs. The study also investigated the relationship between street environments, walking perceptions, and behaviors and found that VR evaluation reflects the sensitivity of walking willingness to the satisfaction of hierarchical walking needs.

Some studies conduct deep learning neural networks (DNNs) to improve the limitation of traditional QoL evaluation methods. Thitisiriwech et al. [23] suggested that the traditional QoL evaluation is costly and time-consuming. Hence, researchers proposed an AI approach using deep convolutional neural networks (DCNNs) and linear regression to predict QoL scores from driving-scene images, using DeepLab-v3+ and YOLO-v3 for knowledge extraction. The results also revealed insights into what makes Bangkokers feel happy or uncomfortable, such as wider roads and walkway spaces correlating with more delight and security and heavy traffic congestion reducing drivers' happiness. Iamtrakul et al. [24] used GIS and DL to analyze the effects of the built environment on Quality of Life in Transportation (QoLT) in Sukhumvit district, Bangkok. They found that individuals perceive QoLT differently depending on the physical characteristics of their environment. The study highlights the importance of understanding QoLT for urban planning and transportation development to achieve sustainable futures.

3. Related Theory

Deep learning (DL) is one of the machine learning techniques that is widely used, as it has been applied in numerous research studies. DL is a subfield of neural networks (NNs) and consists of an input layer, several hidden layers, and an output layer.

Deep supervised learning is a technique that uses labeled data to train the neural network model. The model then predicts output and compares the output with the target label to generate a loss value that will be used to update the weights for the model. There are various of approaches for this technique such as convolutional neural networks (CNNs), deep neural networks (DNNs), and recurrent neural networks (RNN).

This research uses these three neural networks. Image recognition is one of the convolutional neural network approaches that is described in Section 3.1, a DNN approach

is described in Section 3.2, and an RNN approach is described in Section 3.3. In addition, this research uses a bidirectional structure to implement the RNN as described in Section 3.4.

To improve the performance, this research employs knowledge distillation (KD) to reduce complexity and the consumed time of the model, as described in Section 3.5.

3.1. Image Recognition

This approach consists of two processes, object detection and semantic segmentation. The YOLOv4 model is selected for object detection process and the DDRNet-23-slim model is selected for semantic segmentation process. Details of each process are shown below.

- Object detection (YOLOv4)

A general object detection model consists of two parts; a backbone part which is pre-trained on ImageNet and a head part which is used to predict classes and bounding boxes of objects. Modern object detection adds more layers between the backbone and head, which is called the neck. The neck is usually used to collect feature maps from different stages.

YOLOv4 [25] is selected in this process. The model consists of a CSPDarknet53 backbone, an additional spatial pyramid pooling module along with a PANet path-aggregation neck, and YOLOv3 (anchor-based) head. However, there are some methods modified for efficient training and detection—CmBN, SAM, and PAN.

The accuracy of the model is 43.5% AP (65.7% AP50) for the MS COCO dataset at a real-time speed of ~65 FPS on Tesla V100.

- Semantic segmentation (DDRNet-23-slim)

Semantic segmentation is to classify a class for each pixel in the image. The result is a percentage, so the whole result is 100. Dilated convolutions are the standard for semantic segmentation because the convolutions with large dilations can handle very large images and high resolutions. Most state-of-the-art models are established from the ImageNet pre-trained backbone with the dilated convolutions. Figure 1 is an example of a dilated kernel in each rate.

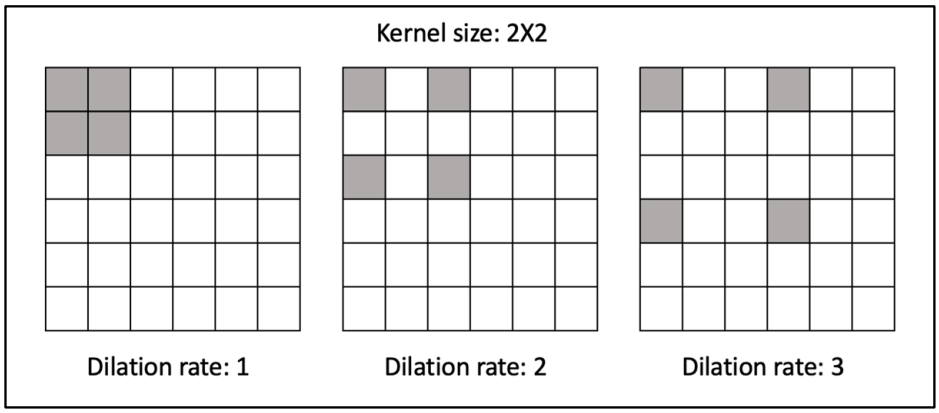


Figure 1. Example of dilated convolution kernel with rates 1, 2, and 3.

The DDRNet-23-Slim model [26] is selected in this process. The model has dual-resolution branches and multiple bilateral fusions for backbones.

Each branch has a bottleneck block at the latest layer and the low-resolution branch also has a deep aggregation pyramid pooling module before the pointwise concatenation layer that creates the result for the segmentation head to predict class labels.

This model is a smaller version of DDRNet-23 that is a trade-off between accuracy and processing speed (76.3 to 74.7 MIoU and 94 to 230 FPS).

3.2. Deep Neural Network

A deep neural network (DNN) consists of an input layer, hidden layers, and an output layer. At the hidden layer, there are stacks of several layers. Therefore, DNN is more complex than traditional neural networks and requires longer durations of time to train the model.

This research uses a Convolutional 1 Dimension (CONV1D) and fully connected neural network. The characteristics of the network are shown in Figures 2 and 3.

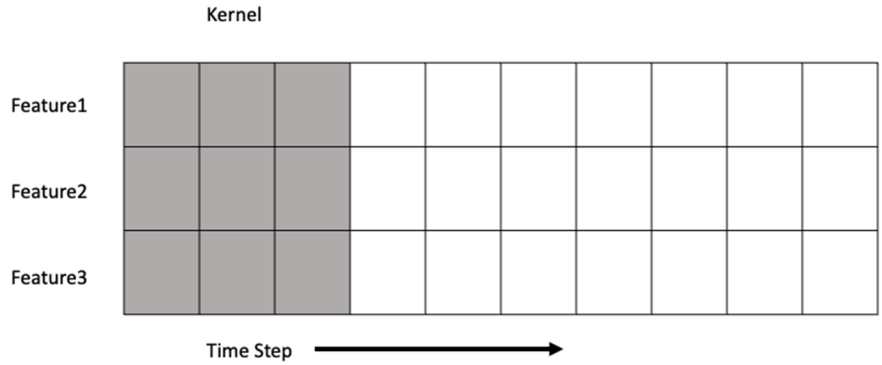


Figure 2. The characteristics of CONV1D neural network.

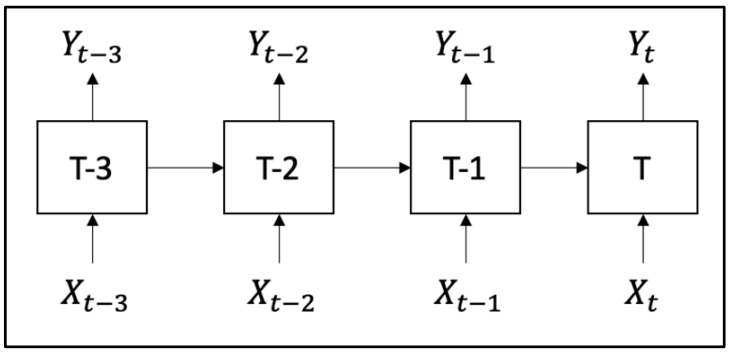


Figure 3. Recurrent Neural Network.

CONV1D is one of the convolutional neural networks whose kernel slides along only one dimension to generate a feature map. This CNN type is proper for time-series data where each feature needs to be processed at the same time through time steps.

A fully connected neural network is the neural network whose current node is connected to every node from the prior layer. At the process step, the node uses mathematic function to calculate the input to be output for the next layer. The function is called the "activation function" that is selected depending on an expected output type. Examples of traditional functions include Rectified Linear Unit (ReLU), Sigmoid, Hyperbolic tangent (tanh), and SoftMax.

3.3. Recurrent Neural Network

A recurrent neural network (RNN) is one of the traditional neural networks. This network is usually trained on sequential data, such as text and video. The output from the previous step will be used as the input for the next step.

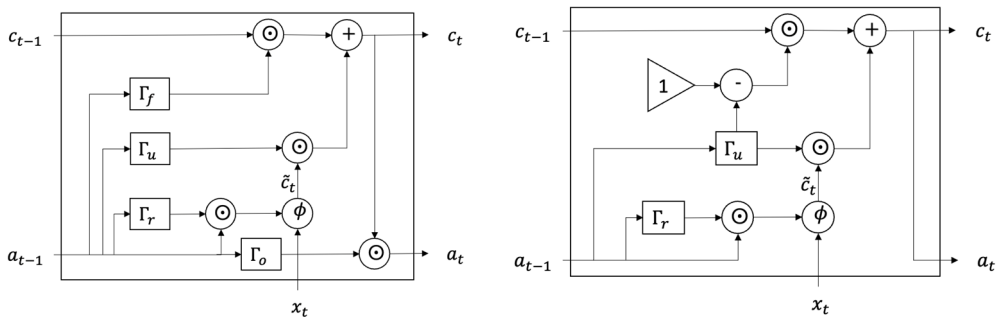
This research uses two RNN models, Long Short-Term Memory units (LSTM) and Gated Recurrent Unit (GRU), as part of the model. These two models can deal with the vanishing gradient problem encountered by traditional RNNs [27].

LSTM and GRU both have the same main component, gates as shown in Equation (1). There are four types of gates: (1) update gate, (2) relevance gate, (3) forget gate, and (4) output gate. Each of these gates has different functions as follows:

- Update gate (Γ_u)—To weigh the importance of previous information to cell state.
- Relevance gate (Γ_r)—To decide if the cell should ignore previous information or not.
- Forget gate (Γ_f)—To identify if cell state should be reset.
- Output gate (Γ_o)—To determine how much information the cell should carry.

$$\Gamma = \sigma(Wx_t + Ua_{t-1} + b) \tag{1}$$

LSTM contains all four types of gates but GRU contains only two types of gates, the update gate and relevance gate. Figure 4a,b show the architecture of LSTM and GRU, respectively.



(**a**) LSTM architecture (**b**) GRU architecture

Figure 4. LSTM architecture (**a**) and GRU architecture (**b**) [27].

3.4. Bidirectional Network

A bidirectional network is a network architecture that incorporates information from both the forward and backward layers of the model, where the outputs of these layers are typically concatenated at each time step. This structure is commonly employed with RNNs. This research uses bidirectional LSTM and bidirectional GRU networks to enhance the prediction performance. The bidirectional nature of these models proves beneficial in specific tasks, such as language translation, where the two-way processing aids in capturing contextual information effectively.

3.5. Knowledge Distillation

Knowledge distillation (KD) [28] is to distill or move knowledge from a complex pre-trained model into a smaller model without significant reduction in performance. An overview of KD is shown in Figure 5.

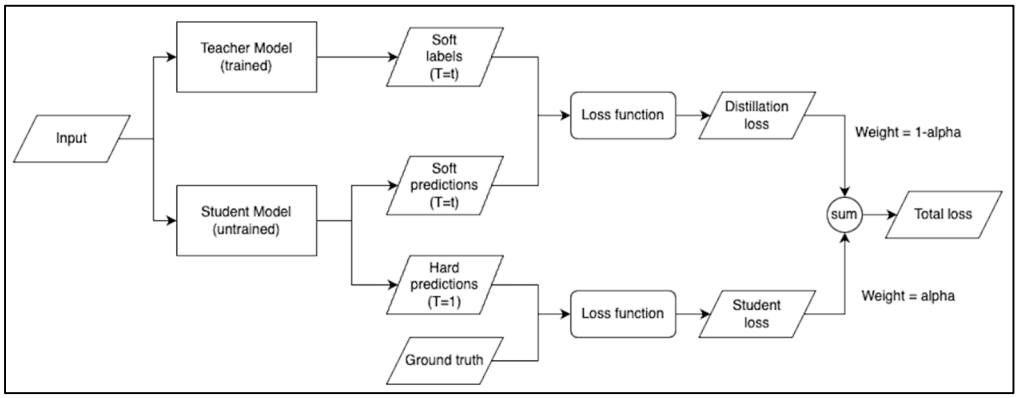


Figure 5. An overview of knowledge distillation.

Firstly, to obtain soft labels, the previous pre-trained model (teacher model) is used to make predictions, then the results are divided by temperature (T). Secondly, soft predictions are calculated by predictions from a trained smaller model (student model) and the results

are divided by temperature (T). When comparing soft labels and soft predictions, this creates a loss result which is called distillation loss. Student loss, another loss result, is from comparing predictions of trained student models to the ground truth. To obtain the total loss of the proposed system in this paper, we multiply these two results of loss with weights and then sum them. The weight of distillation loss is one minus alpha. The weight of student loss is alpha. A pseudo code of this process is shown in Figure 6.

```
Constant value is temperature and alpha

Function distillationLoss (Argument teacher output, Argument student output) {
        Divide argument teacher output by temperature to get teacher soft label
        Divide argument student output by temperature to get student soft label

        Return Compare teacher soft label with student soft label by distillation loss function
}

Function calculateTotalLoss (Argument student loss result, Argument distillation loss result) {
        Return alpha x argument student loss result + (1 - alpha) x argument distillation loss result
}

Function knowledgeDistillation (Argument input, Argument target) {
        Predict student output by student model with argument input
        Compare student output with argument target by student loss function to get student loss result

        Predict teacher output by pre-trained teacher model with argument input
        Send predicted student output and predicted teacher output to the distillationLoss function to get the distillation loss result

        Send student loss result and distillation loss result to the calculateTotalLoss function to get total loss result
        Use total loss result to adjust weights of student model
}
```

Figure 6. A pseudo code of knowledge distillation.

In summary, our proposed method consists of two steps: QoL inference and performance improvement. The QoL inference step involves the application of image recognition, DNN, RNN, and bidirectional network approaches to generate QoL inference. In the performance improvement step, knowledge distillation (KD) is applied to reduce complexity and increase the model's performance, resulting in reduced processing speed. However, it is essential to note that there is a trade-off in this process, as the accuracy may be slightly decreased.

4. Proposed Methods

This section provides details of the processes in our proposed QoL evaluation methods. Section 4.1 describes the overview of our framework, consisting of two steps, QoL inference and KD processes. Section 4.2 describes the dataset used in this research and shows the label used to compare with the prediction results. Section 4.3 describes the feature extraction from images using image recognition models. Section 4.4 describes the architecture used to train the QoL score model and shows the hyperparameters used to configure the architecture. Section 4.5 shows how to process the input and output data before the training step. Section 4.6 describes the training strategy and how to group the dataset. Section 4.7 shows the measuring equation used to evaluate the results. Section 4.8 shows the configuration, hyperparameters, and small architecture used as the student model and final architecture that is used as the teacher model for the KD process.

4.1. Our Framework

Our framework is composed of two processes; the first one is the QoL inference process, and the second one is the KD process.

The QoL inference process is described in Sections 4.3–4.7; the YOLOv4 model and the DDR-Net-23-Slim model extract the image information from the video. Then, we train a DL model using the extracted information as an input label and the data from the questionnaire survey as an output label. Figure 7 shows the overview of the QoL inference process.

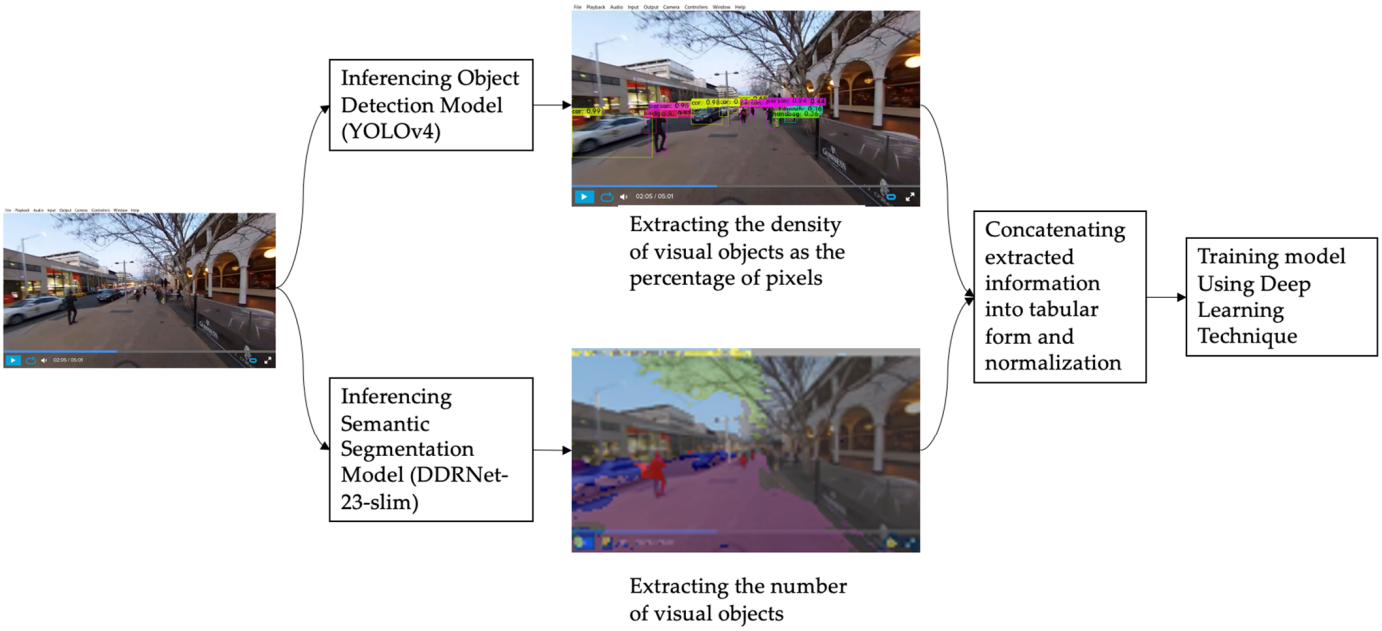


Figure 7. The overview of "the" QoL inference step.

The KD process is described in Section 4.8. We provide the configuration for this step, including the model architecture used as the student model. Figure 8 shows the overview of the KD process.

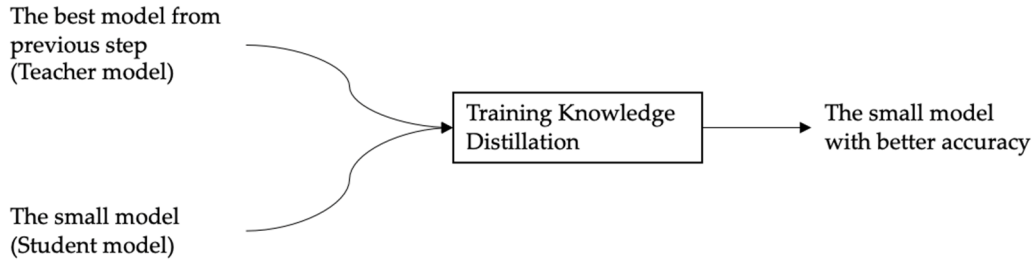


Figure 8. The overview of "the" knowledge distillation step.

4.2. The QoL Dataset

As an input, we use recorded videos of the walking scene from four cities, Canberra, Bangkok, Brisbane, and Sakae. Each city consists of three one-minute scenes. Therefore, the total number of scenes in the input is 12 scenes. The camera used in data collection is RICOH Theta S, recorded at 24 fps for Bangkok, Brisbane, and Canberra and recorded at 30 fps for Sakae.

Table 1 presents the evaluation of walkability indicators, including factors such as Building, Facility, Roof, Bench, Stall, Walking or Sitting People, Footpath Width, Pedestrian, Brightness, Bicycle Parking, Roadside Tree, Electric Pole, Parking, Zebra Crossing, and Traffic Volume. These factors, rated on a scale of 1–10, were derived from the considerations of street design and walking needs, as outlined in the reference paper [19].

Table 1. The 15 factors of evaluated score for each respondent in the questionnaire.

Building	Building
	Facility
	Roof
Activity	Bench
	Stall
	Walking or Sitting People
Footpath	Footpath Width
	Pedestrian
	Brightness
Installation	Bicycle Parking
	Roadside Tree
	Electric Pole
Roadway	Parking
	Zebra Crossing
	Traffic Volume

The questionnaire survey involved displaying virtual scenes from each location to the interviewees using a VR Google tool. The survey occurred at Nakamura Lab, Department of Civil Engineering, Meijo University. There are several benefits to using virtual reality technology for conducting the questionnaire survey. Firstly, it allows interviewees to immerse themselves in the surrounding environment of each scene, enabling them to perceive the scene better and understand the QoL in those settings compared to simply watching video clips on a computer monitor. Secondly, interviewees can control their viewing direction, enabling them to focus on specific elements such as the walking path, sky, trees, or buildings according to their preferences. Finally, because the scenes were collected from different countries, it would be impractical to take interviewees to the actual locations physically. Utilizing VR tools offers a cost-effective solution for surveying such circumstances.

4.3. Feature Extraction Using Image Recognition

This subsection details a feature extraction approach to extract features from prior-recorded videos. This approach consists of two processes, object detection and semantic segmentation.

Each video is captured into pictures frame by frame and then passed to an object detection model, YOLOv4, and a semantic segmentation model, DDRNet-23-Slim. The results of the YOLOv4 model are shown in Figure 9, which visualizes the object detected with bounding boxes, and DDRNet-23-Slim in Figure 10, which visualizes highlight colors at each pixel.

The YOLOv4 model and the DDR-Net-23-Slim model extract the image information from the video. The extracting results from YOLOv4 consist of 17 classes (as shown in Table 2), while the extracting results from DDR-Net-23-Slim consist of 19 classes (as shown in Table 3).

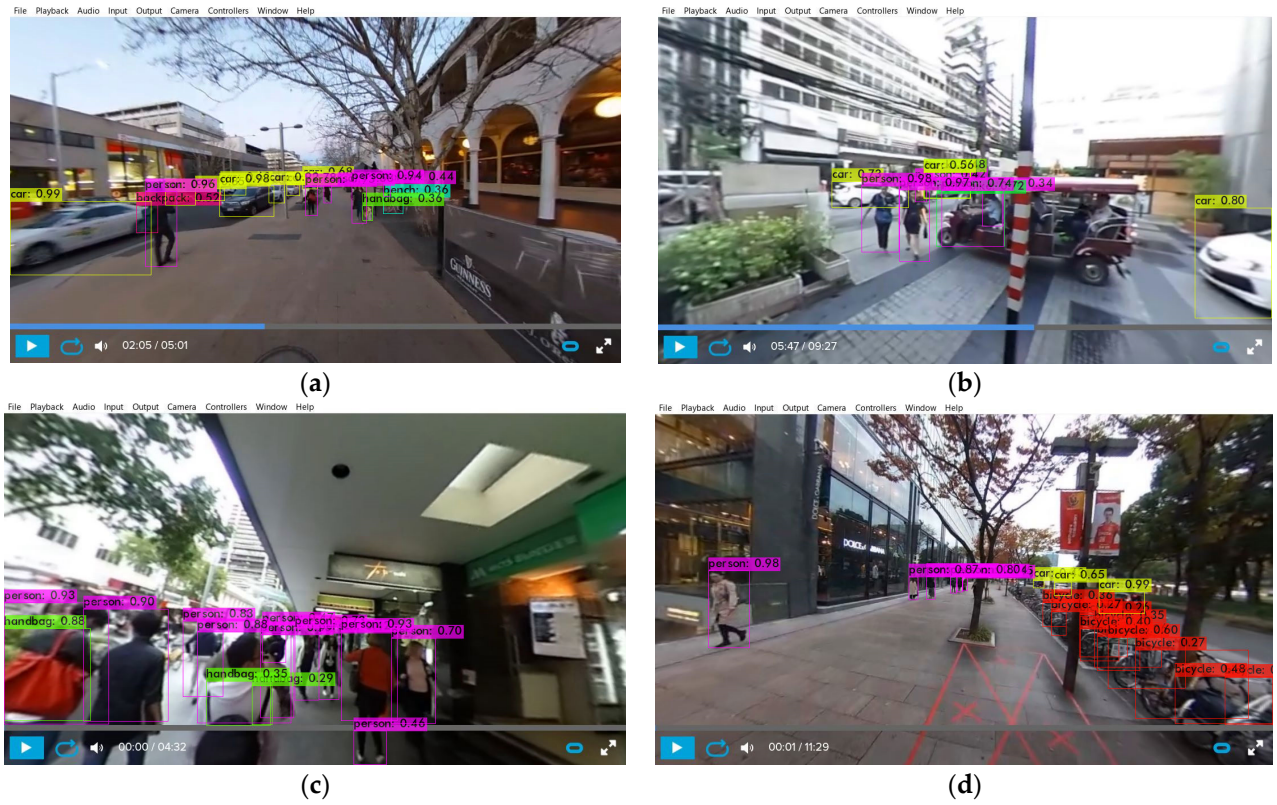


Figure 9. Example of the results from YOLOv4. (**a**) Canberra; (**b**) Bangkok; (**c**) Brisbane; (**d**) Sakae.

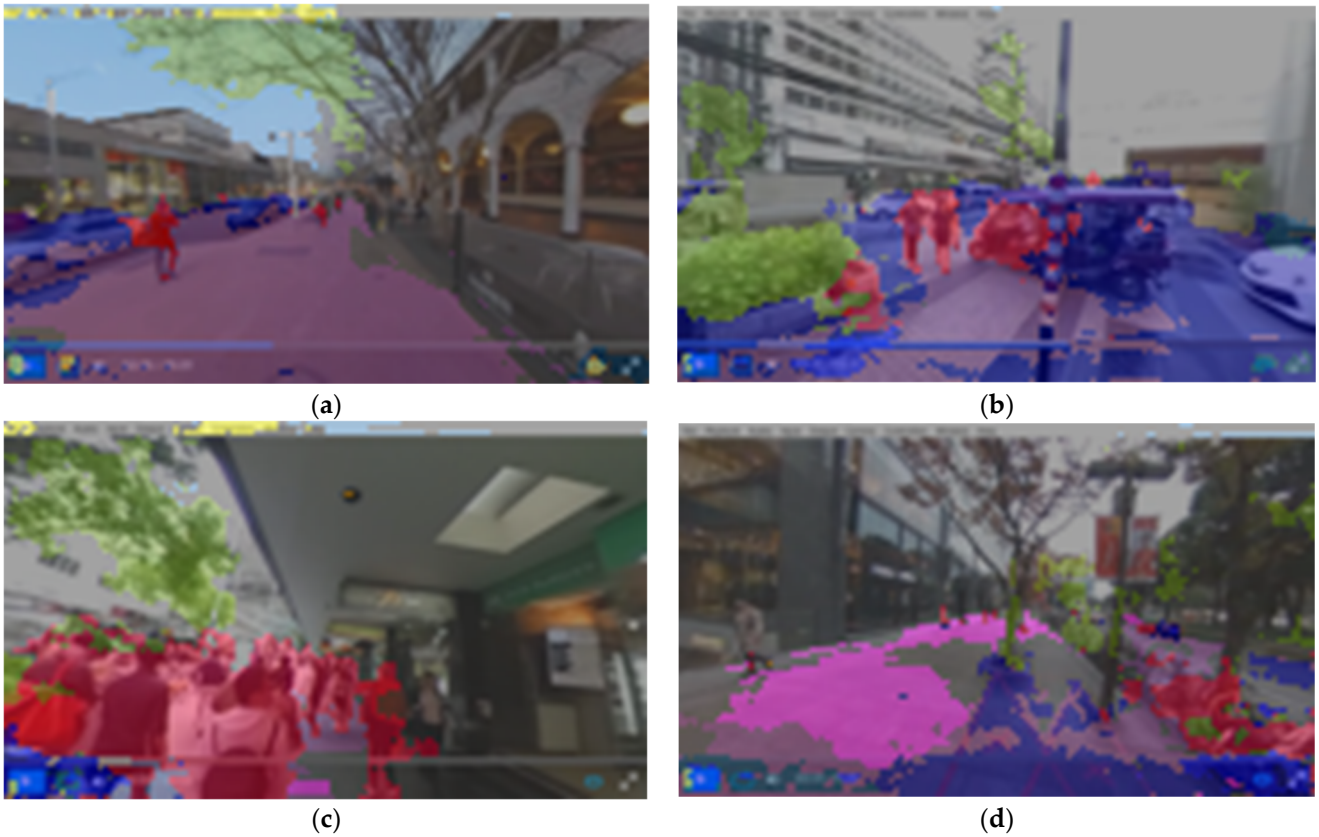


Figure 10. Example of the results from DDRNet-23-Slim. (**a**) Canberra; (**b**) Bangkok; (**c**) Brisbane; (**d**) Sakae.

Table 2. The result classes from YOLOv4 model.

1. Person	10. Traffic light
2. Bicycle	11. Fire hydrant
3. Car	12. Stop sign
4. Motorbike	13. Parking meter
5. Aeroplane	14. Bench
6. Bus	15. Backpack
7. Train	16. Umbrella
8. Truck	17. Handbag
9. Boat	

Table 3. The result classes from DDRNet-23-Slim model.

Colors							
Class	1. Road	2. Sidewalk	3. Building	4. Wall	5. Fence	6. Pole	7. Traffic light
Colors							
Class	8. Traffic sign	9. Vegetation	10. Terrain	11. Sky	12. Person		13. Rider
Colors							
Class	14. Car	15. Truck	16. Bus	17. Train	18. Motorcycle		19. Bicycle

4.4. The QoL Prediction Model Architecture

We use the TensorFlow framework to implement DNN for QoL prediction models. Figure 11 shows the QoL prediction model architecture overview, consisting of 4 boxes. The top box illustrates the architecture's first layer, in which two selectable options are CONV1D or fully connected with the tanh activation function and SoftMax. The second box presents the recurrent network layer, in which four selectable options are LSTM, bidirectional LSTM (bi LSTM), GRU, and bidirectional GRU (bi GRU). Then, the dropout layer is adopted, as shown in the third box. Finally, the stack of five consequences, fully connected with bias and dropout layers, is implemented.

For the top box's first option, CONV1D, the kernel slides along 1 dimension only to generate a feature map; this part uses 13 filter kernels with the size of 64 × 64. Another option is fully connected with the tanh activation function and SoftMax; the tanh activation function is represented as Equation (2), and a SoftMax layer is represented as Equation (3).

$$Tanh = \frac{(e^{x} - e^{-x})}{(e^{x} + e^{-x})} \tag{2}$$

$$SoftMax = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3}$$

where

z_i = input at current position;
e^{z_i} = standard exponential function for an input at current position;
K = number of total input vector;
e^{z_j} = standard exponential function for an input at position j.

Four optional RNNs can be selected for the second box. The last box has a dashed box with five numbers; the highest number must include all the lower numbers. For instance, box number 3 comprises boxes 1 to 3, which have a fully connected layer, dropout, and another fully connected layer.

Based on our hyperparameter tuning, we have determined that the first two layers in this structure should consist of 128 units each. Additionally, we have applied

a dropout layer to mitigate model overfitting and enhance generalization performance. Figures 12 and 13 are examples of QoL prediction model architecture with hyperparameters in each layer and trainable parameters of the models, including the dropout layer (20% or 0.2 dropouts for all models).

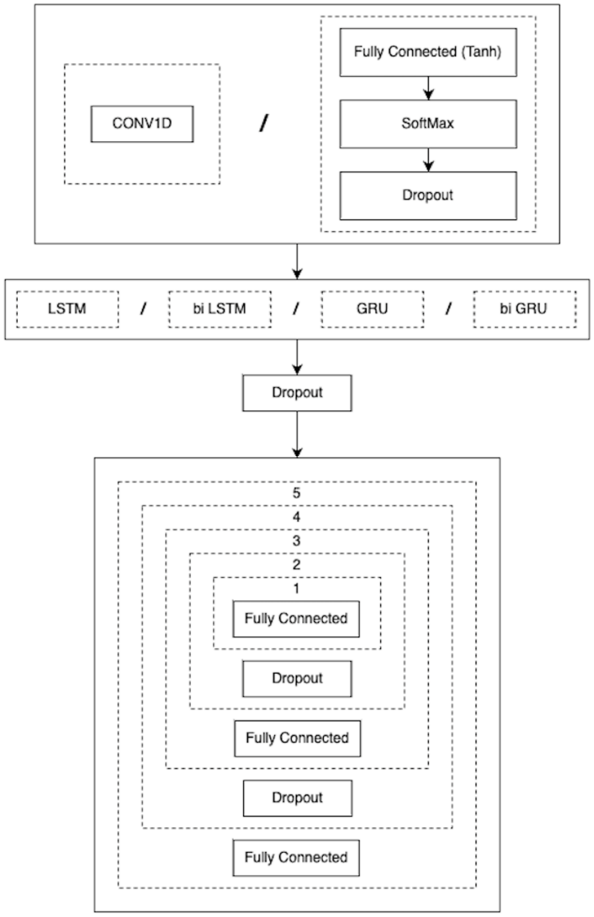


Figure 11. Overview of QoL prediction model architecture.

Layer (type)	Output Shape	Param #
input (InputLayer)	[(None, 15, 37)]	0
conv1d (Conv1D)	(None, 13, 64)	7168
lstm (LSTM)	(None, 128)	98816
dropout (Dropout)	(None, 128)	0
dense (Dense)	(None, 128)	16512
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 128)	16512
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 128)	16512
dense_3 (Dense)	(None, 15)	1935

Total params: 157,455
Trainable params: 157,455
Non-trainable params: 0

Figure 12. The architecture of the QoL prediction model selecting the options of CONV1D for the first box and LSTM for the second box.

Layer (type)	Output Shape	Param #
input (InputLayer)	[(None, 15, 37)]	0
dense (Dense)	(None, 15, 128)	4736
tf.nn.softmax (TFOpLambda)	(None, 15, 128)	0
dropout (Dropout)	(None, 15, 128)	0
lstm (LSTM)	(None, 128)	131584
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 128)	16512
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 128)	16512
dropout_3 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 128)	16512
dense_4 (Dense)	(None, 15)	1935

Total params: 187,791
Trainable params: 187,791
Non-trainable params: 0

Figure 13. The QoL prediction model architecture selecting the options of fully connected with the tanh activation function and SoftMax for the first box and LSTM for the second box.

4.5. Data Preprocessing

We utilized the dataset of one-minute scenes, dividing each scene into 30 frames. In total, there are 360 frames, with 4 cities and 3 scenes per city. To reduce the input size for the QoL prediction model, we calculated the average of features extracted from consecutive video frames. These averaged features were then grouped into 30 samples per scene, serving as input features for the QoL prediction model. The model's output labels were derived from survey results obtained from 50 respondents. The input features for the QoL prediction model are illustrated in Figure 14, consisting of three vectors. The first vector represents the output of the YOLOv4 model, containing 17 features as shown in Table 2. The second vector represents the output of the DDRNet-23-Slim model, comprising 19 features as shown in Table 3. These features were normalized to match the range of values from the output of YOLOv4. Additionally, we appended unique pseudo-demographic numbers to identify the interviewees' answers. Figure 15 provides an overview of the QoL prediction model, demonstrating its overall architecture and flow.

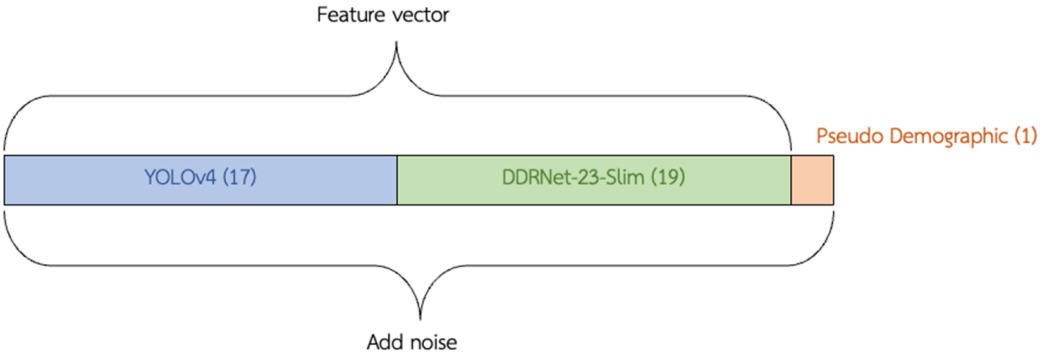


Figure 14. The input features for the QoL prediction model.

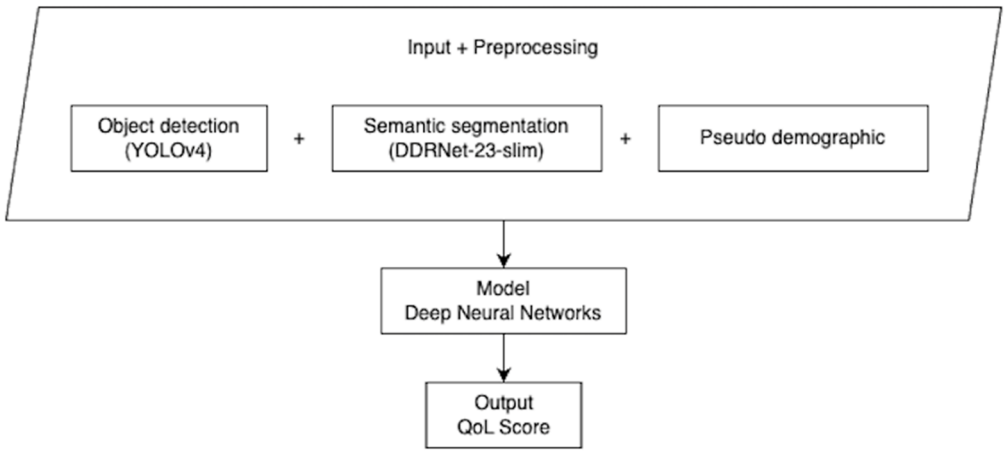


Figure 15. Input features and output labels in the training process of the QoL prediction model.

4.6. Experiment Sets and Dataset Splitting

We set up two experiment groups to evaluate our framework and models: the within-the-city and across-the-cities groups. For the within-the-city group, we arrange the training and testing data from the same city, while for the across-the-cities group, we arrange the data by shuffling along the different cities, as shown in Tables 4 and 5, respectively.

Table 4. Training and testing data for the within-the-city experiment group.

Experiment Set	Training Data	Testing Data
1	Canberra Scene2, Canberra Scene3	Canberra Scene1
2	Canberra Scene1, Canberra Scene3	Canberra Scene2
3	Canberra Scene1, Canberra Scene2	Canberra Scene3
4	Bangkok Scene2, Bangkok Scene3	Bangkok Scene1
5	Bangkok Scene1, Bangkok Scene3	Bangkok Scene2
6	Bangkok Scene1, Bangkok Scene2	Bangkok Scene3
7	Brisbane Scene2, Brisbane Scene3	Brisbane Scene1
8	Brisbane Scene1, Brisbane Scene3	Brisbane Scene2
9	Brisbane Scene1, Brisbane Scene2	Brisbane Scene3
10	Sakae Scene2, Sakae Scene3	Sakae Scene1
11	Sakae Scene1, Sakae Scene3	Sakae Scene2
12	Sakae Scene1, Sakae Scene2	Sakae Scene3

Table 5. Training and testing data for the across-the-cities experiment group.

Experiment Set	Training Data	Testing Data
1	Bangkok Scene all, Brisbane Scene all, Sakae Scene all	Canberra Scene all
2	Canberra Scene all, Brisbane Scene all, Sakae Scene all	Bangkok Scene all
3	Canberra Scene all, Bangkok Scene all, Sakae Scene all	Brisbane Scene all
4	Canberra Scene all, Bangkok Scene all, Brisbane Scene all	Sakae Scene all

4.7. Model Evaluation

Mean Square Error (MSE) is a statistical method for measuring the error between prediction output and ground truth information. The MSE equation is shown in Equation (4), where n denotes the number of labels.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{4}$$

4.8. The Knowledge Distillation Architecture

The KD consists of the teacher model, a model with high complexity and accuracy, and the student model, a small structure; the teacher model helps the student model train easily. Moreover, the model trained by this technique will have lower accuracy than the teacher model but higher accuracy than the training itself. Also, the size of the model is smaller than the teacher model. Therefore, KD is applied to reduce the model size and improve the real-time performance of the teacher model. The structure is shown in Figure 16, and the configuration is shown in Figure 17. The student model structure used for training is shown in Figure 18, and the configuration is shown in Figure 19. In our implementation, we have set the temperature parameter to 10 and the alpha parameter to 0.1, the original values recommended in the reference source [29]. Additionally, we have used MSE as the loss function for both the student and distillation models.

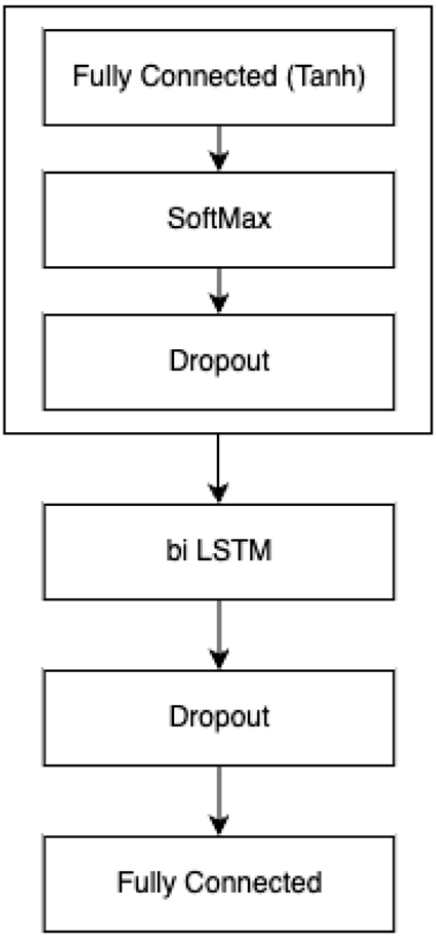


Figure 16. An architecture of the teacher model training in the knowledge distillation process.

```
 Layer (type)                 Output Shape              Param #
=================================================================
 input (InputLayer)           [(None, 15, 37)]          0

 dense (Dense)                (None, 15, 128)           4736

 tf.nn.softmax (TFOpLambda)   (None, 15, 128)           0

 dropout (Dropout)            (None, 15, 128)           0

 bidirectional (Bidirectiona   (None, 256)              263168
 l)

 dropout_1 (Dropout)          (None, 256)               0

 dense_1 (Dense)              (None, 128)               32896

 dense_2 (Dense)              (None, 15)                1935

=================================================================
Total params: 302,735
Trainable params: 302,735
Non-trainable params: 0
_________________________________________________________________
```

Figure 17. A configuration of the teacher model training in the knowledge distillation process.

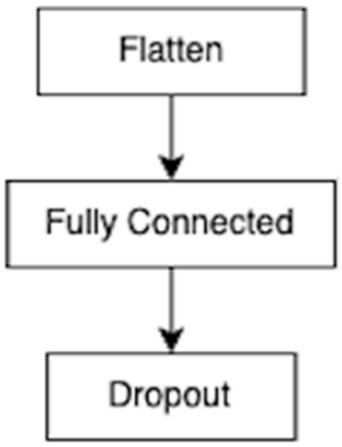


Figure 18. An architecture of the student model training in the knowledge distillation process.

```
 Layer (type)                 Output Shape              Param #
=================================================================
 input (InputLayer)           [(None, 15, 37)]          0

 flatten (Flatten)            (None, 555)               0

 dense (Dense)                (None, 256)               142336

 dropout (Dropout)            (None, 256)               0

 dense_1 (Dense)              (None, 15)                3855

=================================================================
Total params: 146,191
Trainable params: 146,191
Non-trainable params: 0
_________________________________________________________________
```

Figure 19. A configuration of the student model training in the knowledge distillation process.

5. Experiment Results

To verify the effectiveness of our proposed method for QoL inference and knowledge distillation (KD) processes, we conducted experiments using Google Collaboratory. Google Collaboratory is a cloud computing service offering a Python environment and access to high-performance CPU and GPU resources. Specifically, our experiments utilized an Intel(R) Xeon(R) CPU @ 2.20 GHz as the CPU and a Tesla T4 GPU [30].

The results of our experiments are presented in this section, which consists of two subsections. Section 5.1 showcases the outcomes of the QoL prediction model, while Section 5.2 focuses on the results obtained from applying the KD process to the models.

5.1. The Result of the QoL Prediction Model

We set the experiments at 10,000 epochs for training with a 0.001 learning rate. Figures 20 and 21 show the training loss of the within-the-city experiment group and across-the-cities experiment group, respectively. The figures show MSE changes plotted as a graph every 50 epochs, with a total of 200 plotted points.

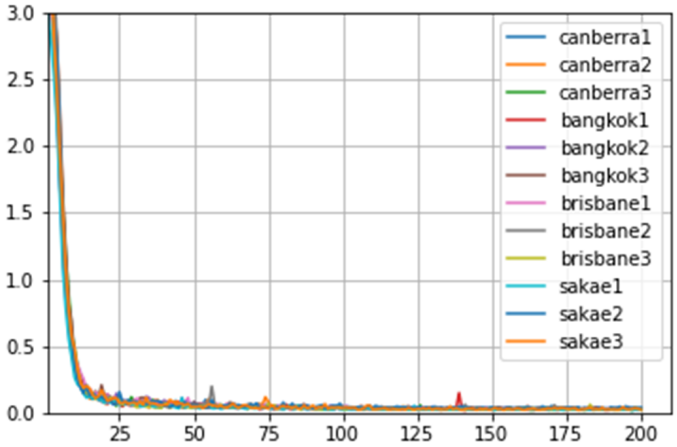


Figure 20. An example of loss graphs showing changes in each epoch of model training for the within-the-city experiment group.

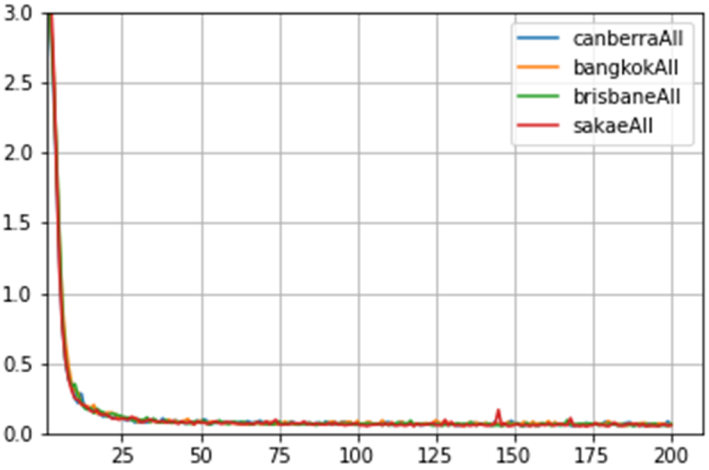


Figure 21. An example of loss graphs showing changes in each epoch of model training for the across-the-cities experiment group.

The test results with the lowest MSE from the within-the-city experiment group and the across-the-cities experiment group setups are shown in Tables 6 and 7, respectively. The results in Table 6 indicate that the fully connected tanh activation function combined with bidirectional LSTM is the best model, winning 8 out of 12 within-the-city experiment

sets. The results in Table 7 also indicate that the fully connected tanh activation function combined with bidirectional LSTM is the best model, winning in the across-the-cities experiment sets.

Table 6. The test results of the within-the-city experiment group.

Training Dataset	Testing Dataset	Model Name	MSE
Canberra Scene2, Canberra Scene3	Canberra Scene1	Tanh bi LSTM 1	7.41×10^{-3}
Canberra Scene1, Canberra Scene3	Canberra Scene2	Tanh bi LSTM 1	8.35×10^{-3}
Canberra Scene1, Canberra Scene2	Canberra Scene3	Tanh bi LSTM 1	7.92×10^{-3}
Bangkok Scene2, Bangkok Scene3	Bangkok Scene1	CONV1D bi LSTM 1	8.11×10^{-3}
Bangkok Scene1, Bangkok Scene3	Bangkok Scene2	Tanh bi LSTM 1	7.55×10^{-3}
Bangkok Scene1, Bangkok Scene2	Bangkok Scene3	CONV1D bi LSTM 1	8.62×10^{-3}
Brisbane Scene2, Brisbane Scene3	Brisbane Scene1	Tanh bi LSTM 1	7.19×10^{-3}
Brisbane Scene1, Brisbane Scene3	Brisbane Scene2	Tanh bi GRU 1	9.27×10^{-3}
Brisbane Scene1, Brisbane Scene2	Brisbane Scene3	Tanh bi LSTM 1	7.55×10^{-3}
Sakae Scene2, Sakae Scene3	Sakae Scene1	Tanh bi GRU 1	6.80×10^{-3}
Sakae Scene1, Sakae Scene3	Sakae Scene2	Tanh bi LSTM 1	1.14×10^{-2}
Sakae Scene1, Sakae Scene2	Sakae Scene3	Tanh bi LSTM 1	8.90×10^{-3}

Table 7. The test results of the across-the-cities experiment group.

Training Dataset	Testing Dataset	Model Name	MSE
Bangkok Scene all, Brisbane Scene all, Sakae Scene all	Canberra Scene all	Tanh bi LSTM 1	9.73×10^{-3}
Canberra Scene all, Brisbane Scene all, Sakae Scene all	Bangkok Scene all	Tanh bi LSTM 1	1.20×10^{-2}
Canberra Scene all, Bangkok Scene all, Sakae Scene all	Brisbane Scene all	Tanh bi LSTM 1	1.16×10^{-2}
Canberra Scene all, Bangkok Scene all, Brisbane Scene all	Sakae Scene all	Tanh bi LSTM 1	1.17×10^{-2}

5.2. The Results of the Knowledge Distillation

We apply the KD process to the fully connected tanh activation function combined with bidirectional LSTM as it is the model with the lowest MSE in both the within-the-city and the across-the-cities experiment groups. The distillation results of the within-the-city group are shown in Tables 8 and 9, while the distillation results of the across-the-cities group are shown in Tables 10 and 11.

Table 8. The MSE results of the within-the-city group.

Training Dataset	Testing Dataset	Student Model MSE	Teacher Model MSE	Distilled Model MSE
Canberra Scene2, Canberra Scene3	Canberra Scene1	2.92	7.16×10^{-3}	6.16×10^{-2}
Canberra Scene1, Canberra Scene3	Canberra Scene2	3.50	9.08×10^{-3}	5.24×10^{-2}
Canberra Scene1, Canberra Scene2	Canberra Scene3	3.62	8.26×10^{-3}	2.77×10^{-2}
Bangkok Scene2, Bangkok Scene3	Bangkok Scene1	3.40	6.09×10^{-3}	3.30×10^{-2}
Bangkok Scene1, Bangkok Scene3	Bangkok Scene2	3.49	8.46×10^{-3}	6.86×10^{-2}
Bangkok Scene1, Bangkok Scene2	Bangkok Scene3	4.04	6.93×10^{-3}	8.91×10^{-2}
Brisbane Scene2, Brisbane Scene3	Brisbane Scene1	3.93	7.74×10^{-3}	5.07×10^{-2}
Brisbane Scene1, Brisbane Scene3	Brisbane Scene2	3.666	7.26×10^{-3}	5.99×10^{-2}
Brisbane Scene1, Brisbane Scene2	Brisbane Scene3	3.567	6.53×10^{-3}	5.34×10^{-2}
Sakae Scene2, Sakae Scene3	Sakae Scene1	2.968	8.73×10^{-3}	4.80×10^{-2}
Sakae Scene1, Sakae Scene3	Sakae Scene2	3.632	9.45×10^{-3}	5.08×10^{-2}
Sakae Scene1, Sakae Scene2	Sakae Scene3	3.641	5.14×10^{-3}	2.93×10^{-2}
Average MSE		3.530	7.57×10^{-3}	5.20×10^{-2}

Table 9. The computational time results of the within-the-city group.

Training Dataset	Testing Dataset	Teacher Model Computational Time (ms)	Distilled Model Computational Time (ms)
Canberra Scene2, Canberra Scene3	Canberra Scene1	7.518	1.271
Canberra Scene1, Canberra Scene3	Canberra Scene2	7.076	0.861
Canberra Scene1, Canberra Scene2	Canberra Scene3	6.804	0.893
Bangkok Scene2, Bangkok Scene3	Bangkok Scene1	6.830	0.908
Bangkok Scene1, Bangkok Scene3	Bangkok Scene2	7.100	0.868
Bangkok Scene1, Bangkok Scene2	Bangkok Scene3	7.023	0.836
Brisbane Scene2, Brisbane Scene3	Brisbane Scene1	7.051	1.306
Brisbane Scene1, Brisbane Scene3	Brisbane Scene2	6.867	0.893
Brisbane Scene1, Brisbane Scene2	Brisbane Scene3	6.991	0.948
Sakae Scene2, Sakae Scene3	Sakae Scene1	10.182	1.295
Sakae Scene1, Sakae Scene3	Sakae Scene2	7.208	0.882
Sakae Scene1, Sakae Scene2	Sakae Scene3	8.152	1.203
Average Computation Time		7.400	1.014

Table 10. The MSE results of the across-the-cities group.

Training Dataset	Testing Dataset	Student Model MSE	Teacher Model MSE	Distilled Model MSE
Bangkok Scene all, Brisbane Scene all, Sakae Scene all	Canberra Scene all	3.81	9.65×10^{-3}	1.23
Canberra Scene all, Brisbane Scene all, Sakae Scene all	Bangkok Scene all	3.55	1.13×10^{-2}	1.28
Canberra Scene all, Bangkok Scene all, Sakae Scene all	Brisbane Scene all	3.51	1.14×10^{-2}	1.14
Canberra Scene all, Bangkok Scene all, Brisbane Scene all	Sakae Scene all	3.87	1.04×10^{-2}	1.26
Average MSE		3.69	1.07×10^{-2}	1.23

Table 11. The computational time results of the across-the-cities group.

Training Dataset	Testing Dataset	Teacher Model Computational Time (ms)	Distilled Model Computational Time (ms)
Bangkok Scene all, Brisbane Scene all, Sakae Scene all	Canberra Scene all	2.330	0.299
Canberra Scene all, Brisbane Scene all, Sakae Scene all	Bangkok Scene all	2.308	0.291
Canberra Scene all, Bangkok Scene all, Sakae Scene all	Brisbane Scene all	2.392	0.292
Canberra Scene all, Bangkok Scene all, Brisbane Scene all	Sakae Scene all	2.302	0.322
Average Computation Time		2.333	0.301

Table 8 shows the results of MSE for each experiment set in the within-the-city group. The results reveal that the distilled models have a lower MSE than the student model but a higher MSE than the teacher model. The conclusion shows that the KD MSE results are as expected for the within-the-city group.

The prediction results obtained from the teacher and distilled models are presented in Figure 22. We have selected examples for each factor from the evaluated QoL score factors. The training set consists of Canberra Scene1 and Canberra Scene2, while the testing set consists of Canberra Scene3. The visualization of the results indicates that both models' expected and predicted results are consistent. Additionally, it is observed that the computation time and model size are smaller for the distilled model compared to the teacher model in the within-the-city experiment groups.

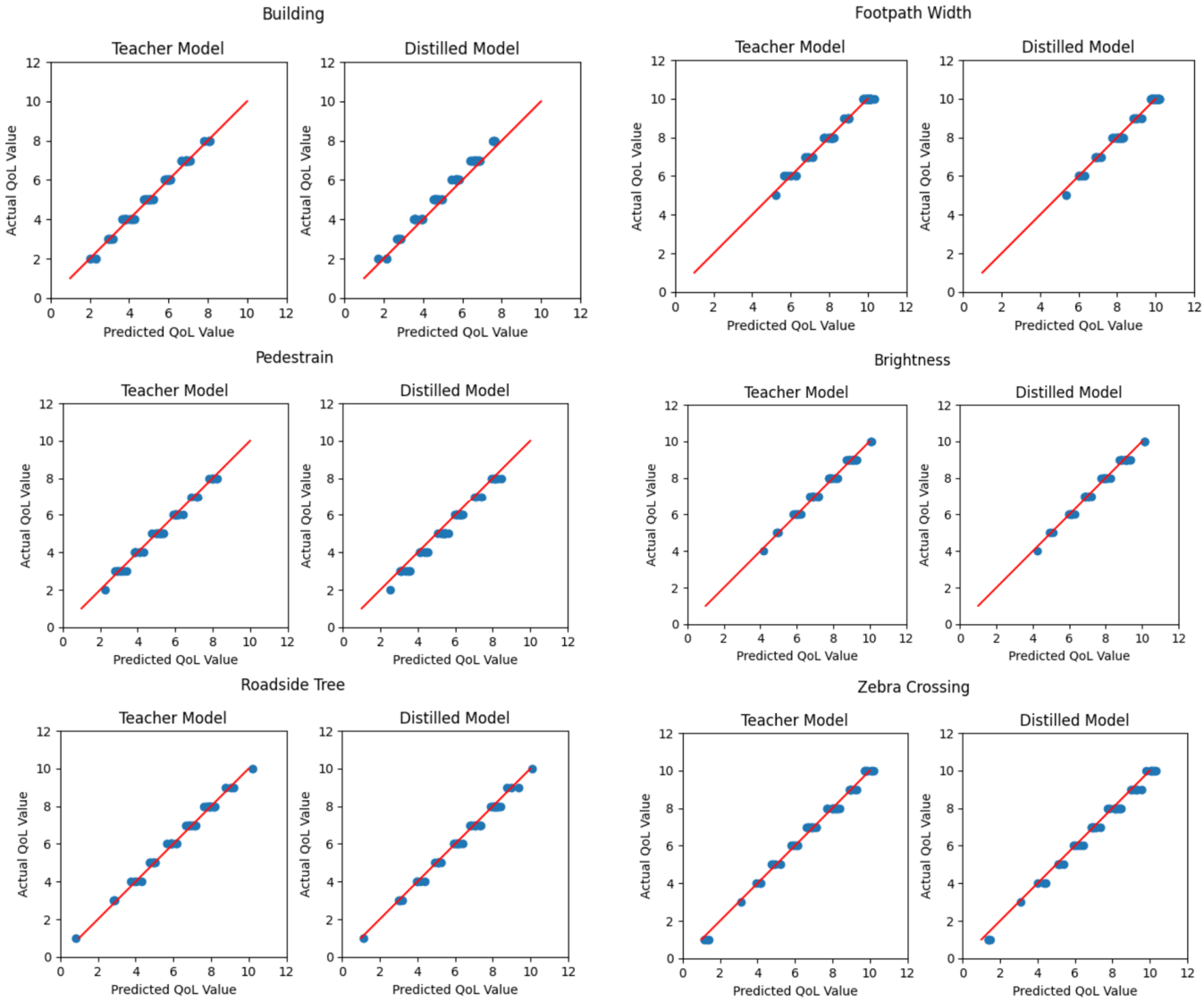


Figure 22. The comparison graph plots between the teacher and distilled models' actual results (*Y*-axis) and predicted results (*X*-axis) selected from a testing set of Canberra Scene3, within-the-city experiment groups.

Table 9 shows the results of the computation time for each experiment set in the within-the-city group. The comparison of the results shows that the distilled model computes the output while consuming a lower computational time than the teacher model. The conclusion indicates that the KD computational time results are as expected for the within-the-city group.

Table 10 shows the results of MSE for each experiment set in the across-the-cities groups. The results show that the distilled models have a lower MSE than the student model but a higher MSE than the teacher model. The conclusion is that the KD MSE results are as expected for the across-the-cities group.

Figure 23 illustrates the prediction outcomes of both the teacher and distilled models. To evaluate the QoL score factor for the across-the-cities experiment group, we chose Bangkok, Brisbane, and Sakae Scene all as the training set, while the Canberra Scene all served as the testing set. The visualization reveals that the teacher model's expected and predicted results closely align with the diagonal line. However, the results from the distilled model are scattered more than the within-the-city experiment group. Moreover,

the distilled model demonstrates reduced computation time and size compared to the teacher model.

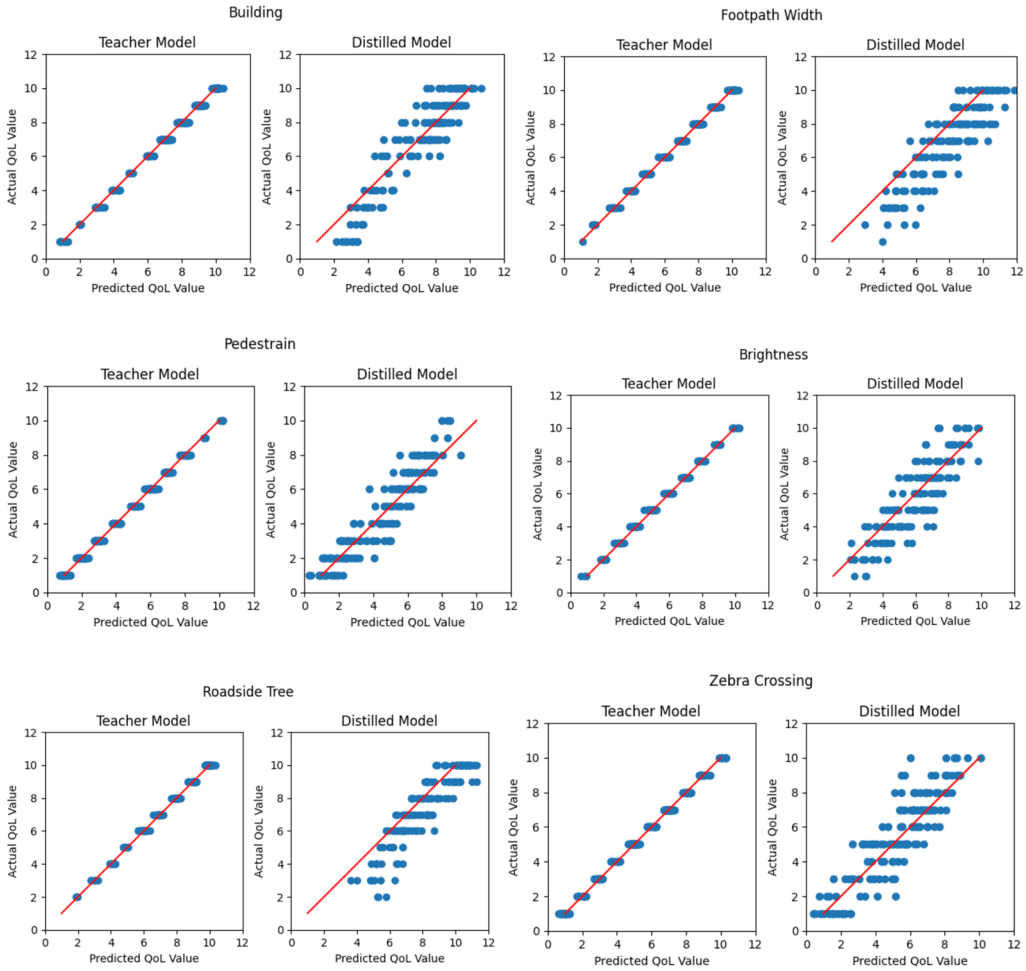


Figure 23. The comparison graph plots between the teacher and distilled models' actual results (*Y*-axis) and predicted results (*X*-axis) selected from a testing set of Canberra Scene all, across-the-cities experiment groups.

Table 11 indicates the results of the computation time for each experiment set in the across-the-cities groups. The results show that the distilled model outperforms the teacher model due to lower computational time. The KD computational time results are as expected for the across-the-cities group.

6. Discussion

6.1. The Result of the QoL Prediction Model

The fully connected tanh activation function combined with bidirectional LSTM is the best model, providing the lowest MSE in almost all experiment sets. The lowest MSEs are 7.19×10^{-3} with Brisbane Scenes1 and 2 as the training dataset and Scene3 as the testing dataset for the within-the-city experiment group as shown in Table 6, and 9.73×10^{-3} with Bangkok Scene all, Brisbane Scene all, and Sakae Scene all as the training dataset and Canberra Scene all as the testing dataset for the across-the-cities experiment group as shown in Table 7.

6.2. The Results of the Knowledge Distillation

The knowledge distillation (KD) is applied to the best model architecture from the previous part (fully connected tanh activation function combined with bidirectional LSTM)

to reduce complexity and the time consumed for QoL prediction by transferring knowledge to the student model. The comparison results show that the distilled model is less time-consuming than the complex model but outputs the QoL with a higher MSE. Additionally, the distilled model boasts a smaller size, approximately half that of the teacher model, as depicted in Figures 17 and 19.

The MSE from the within-the-city experiment group (as shown in Table 8) shows that the distilled model has an average MSE of 5.20×10^{-2}. The MSE from the across-the-cities group (as shown in Table 10) shows that the distilled model has an average MSE of 1.23. Furthermore, the results of the computational time (as shown in Tables 9 and 11, respectively) show that the distilled model can output the QoL value at a lower average computation time, reducing from 7.40 to 1.01 for the within-the-city experiment group and from 2.33 to 0.30 for the across-the-cities group.

7. Conclusions

Megacities prioritize the well-being of their residents, making urban design and sustainable development crucial. Quality of Life (QoL) is an effective key performance index (KPI) used to measure a city plan's efficiency. Walkability is significant for city dwellers, and the traditional questionnaire survey approach to evaluate it is costly, time-consuming, and limited in its evaluation area. To address these limitations, we propose using artificial intelligence (AI) technology to evaluate walkability data collected through a questionnaire survey using virtual reality (VR) tools.

The proposed method involves using deep convolutional neural networks (DCNNs) for knowledge extraction and deep learning (DL) models to infer QoL scores. In our case, the experiment results indicate that a smaller model outperforms a larger model due to a low number of layers for both experiments that train the data within and across the city. KD can aid the model by using a large and complicated model to train a smaller model, resulting in higher accuracy and a lower time consumption. In summary, the experiment results show that the proposed approach is practical and can be an alternative method for acquiring QoL scores.

Author Contributions: Conceptualization, T.R., K.T., P.K., B.K. and Y.I.; methodology, T.R., K.T. and P.K.; software, T.R. and K.T.; validation, T.R., K.T., P.K. and B.K.; formal analysis, T.R., K.T. and P.K; investigation, T.R., K.T., P.K., B.K., Y.I. and S.F.; resources, Y.I., S.F. and K.N..; data curation, Y.I., S.F. and K.N.; writing—original draft preparation, T.R.; writing—review and editing, T.R., P.K. and B.K.; visualization, T.R.; supervision, P.K., B.K., Y.I. and S.F.; project administration, T.R., K.T., P.K. and Y.H.; funding acquisition, Y.I. and Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Science and Technology Research Partnership for Sustainable Development (SATREPS), Japan Science and Technology Agency (JST)/Japan International Cooperation Agency (JICA) "Smart Transport Strategy for Thailand 4.0" (Chair: Yoshitsugu Hayashi, Chubu University, Japan) under Grant JPMJSA1704, and by Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (C)(20K11873) and by Chubu University Grant.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

QoL	Quality of Life
QoLT	Quality of Life Transportation
KPI	Key Performance Index
AI	Artificial Intelligence
DCNNs	Deep Convolutional Neural Networks
DL	Deep Learning
NN	Neural Network
RNN	Recurrent Neural Network
CNN	Convolutional Neural Network
DNN	Deep Neural Network
KD	Knowledge Distillation
YOLO	You Only Look Once
DDRNet	Deep Dual-resolution Networks
CSPNet	Cross Stage Partial Network
SPP	Spatial Pyramid Pooling
PANet	Path Aggregation Network
CmBN	Cross mini-Batch Normalization
SAT	Self-adversarial training
SAM	Spatial Attention Module
mIoU	mean Intersection over Union
FPS	Frames Per Second
CONV1D	Convolutional 1 Dimension
ReLU	Rectified Linear Unit
Tanh	Hyperbolic Tangent
LSTM	Long Short-Term Memory units
GRU	Gated Recurrent Unit
MSE	Mean Square Error

References

1. Nakamura, K.; Wasuntarasook, V.; Gu, F.; Vichiensan, V.; Kii, M.; Hayashi, Y. Evaluation for Low-carbon Land-use Transport Development with QOL Indexes in Asian Developing Megacities: A Case Study of Bangkok. *J. East. Asia Soc. Transp. Stud.* **2015**, *11*, 1047–1063.
2. Nakamura, K.; Morita, H.; Vichiensan, V.; Togawa, T.; Hayashi, Y. Comparative analysis of QOL in station areas between cities at different development stages, Bangkok and Nagoya. *Transp. Res. Procedia* **2017**, *25*, 3188–3202. [CrossRef]
3. Besleme, K.; Mullin, M. Community indicators and healthy communities. *Natl. Civ. Rev.* **1997**, *86*, 43–52. [CrossRef]
4. Diener, E.; Suh, E. Measuring quality of life: Economic, social, and subjective indicators. *Soc. Indic. Res.* **1997**, *40*, 189–216. [CrossRef]
5. Kachi, N.; Kato, H.; Hayashi, Y.; Black, J. Making cities more compact by improving transport and amenity and reducing hazard risk. *J. East. Asia Soc. Transp. Stud.* **2005**, *6*, 3819–3834.
6. Briassoulis, H. Sustainable development and its indicators: Through a (planner's) glass darkly. *J. Environ. Plan. Manag.* **2001**, *44*, 409–427. [CrossRef]
7. Felce, D.; Perry, J. Quality of life: Its definition and measurement. *Res. Dev. Disabil.* **1995**, *16*, 51–74. [CrossRef] [PubMed]
8. Veenhoven, R. Freedom and happiness: A comparative study in forty-four nations in the early 1990s. In *Subjective Well-Being Across Cultures*; MIT Press: Cambridge, MA, USA, 2000; pp. 257–288.
9. Hayashi, Y.; Sugiyama, I. Dual strategies for the environmental and financial goals of sustainable cities: De-suburbanization and social capitalization. *Built Environ.* **2003**, *29*, 8–15. [CrossRef]
10. Doi, K.; Kii, M.; Nakanishi, H. An integrated evaluation method of accessibility, quality of life, and social interaction. *Environ. Plan. B Plan. Des.* **2008**, *35*, 1098–1116. [CrossRef]
11. Wasuntarasook, V.; Hayashi, Y. A historic review on consequences of critical events leading revolution in Mass Rapid Transit in Bangkok. In Proceedings of the 10th International Conference of Eastern Asia Society for Transportation Studies, Taipei, Taiwan, 9–12 September 2013; pp. 9–12.
12. Alonso, W. *Location and Land Use*; Harvard University Press: Cambridge, MA, USA, 1964.
13. Hayashi, Y.; Mai, X.; Kato, H. The role of rail transport for sustainable urban transport. In *Transport Moving to Climate Intelligence*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 161–174.
14. Banister, D. The sustainable mobility paradigm. *Transp. Policy* **2008**, *15*, 73–80. [CrossRef]
15. Vichiensan, V.; Nakamura, K. Walkability perception in Asian cities: A comparative study in Bangkok and Nagoya. *Sustainability* **2021**, *13*, 6825. [CrossRef]

16. Nakamura, K. The relationship between walkability and QOL outcomes in residential evaluation. *Cities* **2022**, *131*, 104008. [CrossRef]
17. Vichiensan, V.; Fukada, A.; Malaitham, S. Perceived walkability evaluation in transit station catchment areas in Bangkok. In Proceedings of the 2022 Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, 27–29 July 2022; pp. 1–6.
18. Sou, K.; Shiokawa, H.; Yoh, K.; Doi, K. Street Design for Hedonistic Sustainability through AI and Human Co-Operative Evaluation. *Sustainability* **2021**, *13*, 9066. [CrossRef]
19. Morita, K.; Nakamura, K. VR Pedestrian Space Evaluation Considering Sidewalk Boundary Space Design. *J. Jpn. Soc. Civ. Eng. D3 Civ. Eng. Plan.* **2019**, *75*, I_137–I_143.
20. Nakamura, K. Experimental analysis of walkability evaluation using virtual reality application. *Environ. Plan. B Urban Anal. City Sci.* **2021**, *48*, 2481–2496. [CrossRef]
21. Kantavat, P.; Kijsirikul, B.; Iwahori, Y.; Hayashi, Y.; Panboonyuen, T.; Vateekul, P.; Achariyaviriya, W. Transportation mobility factor extraction using image recognition techniques. In Proceedings of the 2019 First International Conference on Smart Technology & Urban Development (STUD), Chiang Mai, Thailand, 13–14 December 2019; pp. 1–7.
22. Thitisiriwech, K.; Panboonyuen, T.; Kantavat, P.; Iwahori, Y.; Kijsirikul, B. The Bangkok Urbanscapes Dataset for Semantic Urban Scene Understanding Using Enhanced Encoder-Decoder with Atrous Depthwise Separable A1 Convolutional Neural Networks. *IEEE Access* **2022**, *10*, 59327–59349. [CrossRef]
23. Thitisiriwech, K.; Panboonyuen, T.; Kantavat, P.; Kijsirikul, B.; Iwahori, Y.; Fukui, S.; Hayashi, Y. Quality of Life Prediction in Driving Scenes on Thailand Roads Using Information Extraction from Deep Convolutional Neural Networks. *Sustainability* **2023**, *15*, 2847. [CrossRef]
24. Iamtrakul, P.; Chayphong, S.; Kantavat, P.; Hayashi, Y.; Kijsirikul, B.; Iwahori, Y. Exploring the Spatial Effects of Built Environment on Quality of Life Related Transportation by Integrating GIS and Deep Learning Approaches. *Sustainability* **2023**, *15*, 2785. [CrossRef]
25. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
26. Hong, Y.; Pan, H.; Sun, W.; Jia, Y. Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes. *arXiv* **2021**, arXiv:2101.06085.
27. Amidi, A.; Amidi, S. Recurrent Neural Networks Cheatsheet. Available online: https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks (accessed on 1 November 2022).
28. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
29. Borup, K. Knowledge Distillation. 2020. Available online: https://keras.io/examples/vision/knowledge_distillation (accessed on 1 November 2022).
30. Available online: https://colab.research.google.com (accessed on 19 June 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Text Emotion Recognition Based on XLNet-BiGRU-Att

Tian Han [1,2], Zhu Zhang [1,2,*], Mingyuan Ren [1], Changchun Dong [1], Xiaolin Jiang [1] and Quansheng Zhuang [2]

[1] Jinhua Advanced Research Institute, Jinhua 321013, China; htopen@foxmail.com (T.H.); rmy2000@126.com (M.R.); hitdongcc@163.com (C.D.); jlynner@163.com (X.J.)
[2] School of Measurement and Communication Engineering, Harbin University of Science and Technology, Harbin 150080, China; izqshust@tom.com
* Correspondence: zhuzhang.zz@foxmail.com

Abstract: Text emotion recognition (TER) is an important natural language processing (NLP) task which is widely used in human–computer interaction, public opinion analysis, mental health analysis, and social network analysis. In this paper, a deep learning model based on XLNet with bidirectional recurrent unit and attention mechanism (XLNet-BiGRU-Att) is proposed in order to improve TER performance. XLNet is used to build bidirectional language models which can learn contextual information simultaneously, while the bidirectional gated recurrent unit (BiGRU) helps to extract more effective features which can pay attention to current and previous states using hidden layers and the attention mechanism (Att) provides different weights to enhance the 'attention' paid to important information, thereby improving the quality of word vectors and the accuracy of sentiment analysis model judgments. The proposed model composed of XLNet, BiGRU, and Att improves performance on the whole TER task. Experiments on the Interactive Emotional Dyadic Motion Capture (IEMOCAP) database and the Chinese Academy of Sciences Institute of Automation (CASIA) dataset were carried out to compare XLNet-BiGRU-Att, XLNet, BERT, and BERT-BiLSTM, and the results show that the model proposed in this paper has superior performance compared to the others.

Keywords: text emotion recognition; XLNET; BiGRU; attention; BERT

1. Introduction

The area of text emotion recognition has garnered significant interest in recent years, largely due to the proliferation of digital communication devices. Text emotion recognition involves analyzing the emotions expressed within text to gain a more comprehensive understanding of its contents, including its underlying sentiment, purpose, and expression [1]. This technology has been widely adopted across different industries, including business, education, and communication [2].

The underlying principle behind detecting emotional content in text is rooted in natural language processing, an expanding research domain that has gained tremendous momentum in the wake of a burgeoning volume of online comments. The core operation in recognizing emotional content in text involves a systematic text preprocessing phase that entails segmenting words, annotating parts of speech, and analyzing syntactic and semantic structures, among others [3]. The subsequent stage is text feature extraction, in which quantized word vectors derived from the text and fed into the classifier to categorize emotions [4,5]. These extracted features are represented in a format amenable to recognition by computer algorithms. Currently, numerous widely employed algorithms exist for extracting statistical features [6]. One-hot encoding is an encoding technique that yields a binary representation of classified values, and has proven moderately effective [7]. Another commonly used text representation technique is the Term Frequency–Inverse Document Frequency (TF-IDF) method, which explores word frequency to create a simple and rapid representation of text [8–10]. In summary, this technology serves as a means of mitigating noise by assigning weights to different elements. The algorithm involves

calculating the frequency of words or phrases in a particular text and the frequency of those same elements across the entire corpus. While the idea behind the algorithm is relatively straightforward, particular attention must be paid to selecting effective features, determining feature weights, and designing classification algorithms. Recent proposals in the literature include a novel network structure comprising a Bidirectional Long Short Term Memory (BiLSTM) model and hierarchical attention mechanism [11,12]. Within this structure, data are encoded and transmitted to the BiLSTM using a single key, then the output is stratified. Finally, the processed features are classified using a softmax classifier. Experimental results demonstrate the structure's high accuracy in text classification.

The Bidirectional Encoder Representations from Transformers (BERT) pre-trained model, has been hailed as a breakthrough in Natural Language Processing (NLP) due to its superior level of language comprehension [13]. However, there are certain limitations associated with the BERT model, such as its fixed input length size and word piece embedding issues, as well as computational complexities [14,15]. To address these concerns, several new pre-trained models such as Generalized Auto-regression Pre-training for Language Understanding (XLNet) [16], Robustly optimized BERT pre-training Approach (RoBERTa) [17], and DistilBERT [18] have been proposed. The effectiveness of BERT and its variants in the field of NLP has led researchers to shift their focus towards exploring text emotion recognition using this methodology [19,20]. This article presents an enhanced model of XLNET utilizing this approach for the purpose of accurately recognizing emotions within text.

In summary, to improve recognition accuracy based on existing text emotion recognition technologies, we need to optimize several problems. First of all, when processing information, we need to integrate and understand contextual information. secondly, we need to optimize the model to reduce dependence on long text. Thirdly, we need to enhance the representation and extract more effective features of text. Finally, it is better to reduce the complexity of the model. Based on the above objectives, we proposed a model composed of XLNet, BiGRU and Att. XLNet can process short sentences and obtain relatively rich semantic information by using context information to achieve effective representation of text. BiGRU can also combine the context bidirectional semantic information to effectively extract semantic features, while the parameter quantity is significantly reduced compared to LSTM. The attention mechanism can further allocate the importance of each feature through weights, achieving effective emotional classification output. In this paper, the proposed model for TER is presented in detail, comparative experiment is performed and the result is discussed.

2. Materials and Methods

2.1. Dataset

In this research, text data from two emotion datasets, namely, CASIA and IEMOCAP, were utilized to verify the model proposed.

The CASIA Chinese emotional corpus was initially curated by the esteemed Institute of Automation at the Chinese Academy of Sciences. The repository comprises of recordings by four proficient speakers and encompasses six distinct emotional states (anger, happiness, sadness, fear, surprise, and neutrality), totalling an impressive 9600 pronunciations. A noteworthy aspect of the corpus is that while it contains 300 identical texts, the different emotional renderings of each exhibit diverse acoustic and rhythmic performances when subjected to precise comparative analysis. In addition, the corpus features 100 distinct texts which the literal meaning suggests as having inherent emotional leanings, facilitating more consistent and accurate emotion depiction. For our experiment, we handpicked the identical texts in corpus for use a experimental data, with the first 200 readings assigned to the training set and the residual 100 allocated for testing. Furthermore, the experiment involved partitioning of the dataset [21].

The Interactive Emotional Binary Motion Capture Database (IEMOCAP) was collected by the Speech Analysis and Interpretation Laboratory (SAIL) of the University of Southern

California (USC). This database is an invaluable resource for research and modeling in the field of multimodal and expressive human communication; it contains over twelve hours of data recorded from ten actors engaged in binary conversation scenes along with detailed motion capture information that identifies the subjects' facial expressions and hand movements. Additionally, the database includes interactive settings stimulating specific emotions such as happiness, anger, sadness, and depression in emotionally scripted and naturally occurring conversation scenarios [22].

2.2. Text Emotion Recognition Model Based on XLNet

This paper proposes a novel approach to enhance the accuracy of sentiment analysis through the development of a model based on XLNet with bidirectional recurrent unit and attention mechanism (XLNet-BiGRU-Att). The proposed methodology addresses the challenges associated with vectorization of short text sentiment analysis. Unlike traditional language models that only consider unidirectional information, the proposed XLNet model is a two-way modeling language model that simultaneously captures information in context both above and below, making the resulting word vectors more semantically rich. In addition, the BiGRU-Attention network layer effectively filters important information within limited text space and assigns different weights to filtered word vectors in order to enhance their "attention power". These measures collectively improve the performance of the sentiment analysis model. The architecture of XLNet-BiGRU-Att is shown in Figure 1, which is mainly composed of thr input layer, XLNet-BIGRU-Att model layer, and output layer. In the input layer, every word in the input sentence is transferred into a word vector by the embedding function, where the raw word vectors $[x_1, x_2, \cdots, x_t]$ represent the input of XLNet. In XLNet, the raw word vectors are processed by two-stream self-attention and the vectors are calculated concurrently in two channels: the content stream $h_T^{(2)}$, and the query stream $g_T^{(2)}$. As the query stream contains the position information, it is used as the output of XLNet during the prediction process. The BiGRU layer is used for extraction of deep emotional features; it contains both forward and backward GRUs, as shown in Figure 1, where h is the hidden state. Through BiGRU, word vectors can be used to more fully learn the relationships between contexts and perform semantic encoding. The attention layer assigns corresponding probability weights to different word vectors in order to further extract text features and highlight the key information of the text. In the end, the output layer is a fully connected layer and a softmax function is used to provide the emotional classification result.

XLNet is a state-of-the-art model for pre-training in semantic understanding. It builds upon previous models such as mask language models and autoregressive (AR) language models, and addresses specific challenges in the pre-training stage of BERT. One key advantage of XLNet is its ability to handle inconsistencies between the mask flag and the fine-tuning process. Additionally, it effectively resolves the dependency problem between masked words. This is achieved through a unique approach that reconstructs the input text in a permutation and composition manner. Unlike BERT, XLNet applies this approach during the fine-tuning stage, using the Transformer attention mask matrix and double-flow attention mechanism to achieve different combinations and permutations. This allows for the integration of contextual features into the model training process.

XLNet is based on the autoregression (AR) language model. XLnet uses the idea of random sorting to solve the problem of AR models being unable to introduce two-directional text information. The random sorting process simply sorts the sequence number of each position randomly. After sorting, the words prior to each individual word can be used to predict its probability. Here, the previous words may be drawn from either the words prior to the current word in the original sentence or from the words after it, which is equivalent to using the bidirectional sequence information.

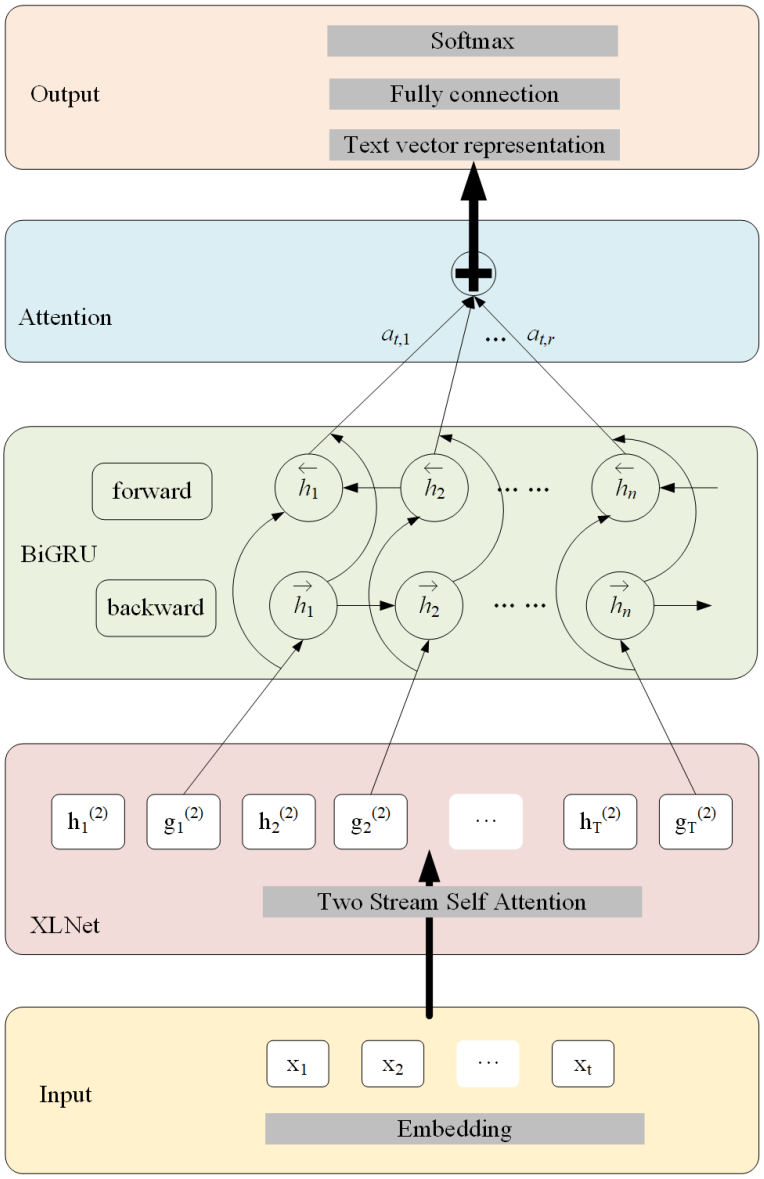


Figure 1. The XLNet-BIGRU-Att architecture.

First, the Permutation Language Model (PLM) is constructed, which disrupts the order of the original word vectors. Assuming that the original sequence is $x(x_1, x_2, x_3, x_4)$, then its arrangement is $(1, 2, 3, 4)$ and the word order is fully arranged as $(1, 3, 4, 2)$, $(2, 4, 3, 1)$, $(3, 2, 4, 1)$, etc. Figure 2 presents an example of the prediction of x_3 based on the different orders produced by sequential factorization.

In Figure 2, x_i represents the input word vectors, $mem^{(0)}$ is the hidden state of the input layer, $mem^{(1)}$ is the hidden state of the first layer, $h_i^{(1)}$ is the predicted result of the ith word in the first layer, $h_i^{(2)}$ is the predicted result of the ith word in the second layer, and x_3 is the output predicted result, which can be understood as a word vector encoding. As shown in Figure 2a, when the factorization order is $(3, 2, 4, 1)$, the prediction of x_3 cannot involve attention to any other words, and can only be predicted based on the previous hidden state *mem*. As shown in Figure 2b, when the factorization order is $(2, 4, 3, 1)$, the semantics of x_3 can be predicted according to the meaning of x_2 and x_4. Unlike the AR language model, this method captures the contextual semantic information of x_4, which achieves the purpose of obtaining contextual semantics.

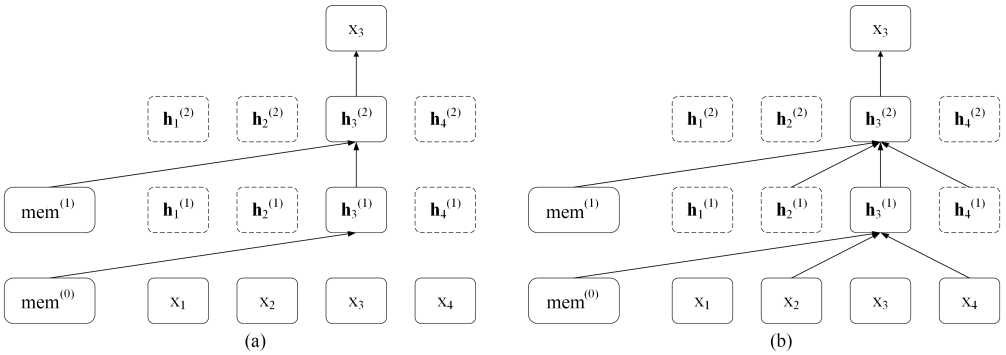


Figure 2. Different orders produced by sequential factorization: (**a**) factorization order 3, 2, 4, 1 and (**b**) factorization order 2, 4, 3, 1.

Second, all the factorization orders should be sampled. For a given length T of text sequence x, $T!$ results can be obtained by fully sorting the sequence. However, when the text sequence is too long, the complexity of the algorithm increases. The random sorting language model needs to randomly sample $T!$ results to remove useless sequences. Random sampling of all the factorization orders of the text sequence is realized by Equation (1):

$$\max_{\theta} E_{z \sim Z_T} \left[\sum_{t=1}^{T} \lg p_\theta(x_{Z_t} | x_{z<t}) \right] \tag{1}$$

In Equation (1), Z_T is the set of sequences with length T composed of sequences arranged from the original sequence, z is the sequence sampled from Z_T, z_t is the value at the position of t in sequence z, $z < t$ represents the tth element and the preceding $(t-1)$ elements in sequence z, and $E_{z \sim Z_T}$ is the expectation of the sampling result.

Finally, the later words in the sequence are predicted; for example, the original language sequence is arranged in the order $(1, 2, 3, 4)$ and the language sequence after it is fully arranged as $(2, 3, 4, 1)$ and $(1, 4, 2, 3)$. When the predicted position is x_3, the word order $(2, 3, 4, 1)$ can only be used to obtain the semantic information of x_2, while when the word order is $(1, 4, 2, 3)$, the semantic information of x_1, x_4, and x_2 can be obtained. Therefore, the model prefers that the predicted words be located at the end of the sentence, as this can provide better contact with the context semantic information.

2.2.1. Attention Mask in XLNet

The principle of the attention mask mechanism is to cover the part to be predicted inside the transformer to ensure that it does not play any role in the prediction process. For example, the original language sequence arrangement is $(1, 2, 3, 4)$ and the sampling sequence order is $(2, 4, 3, 1)$. The first line of the mask is used to predict x_1, the second line of the mask is used to predict x_2, and so on. The white dot means that the word in this position is masked, while the black dot means that the information of the word in this position can be contacted. As shown in Figure 3, when the semantic information of x_3 is predicted, the context of x_2 and x_4 can be used; thus, the positions of x_2 and x_4 in the third line of the mask are shaded. Because the sequence order is $(2, 4, 3, 1)$ and x_2 is first in the sequence, there is no reference information available for x_2 and the second line is fully masked. Following this theory, the mask matrix of the sequence $(2, 4, 3, 1)$ is shown in Figure 3. Black circle indicates the available data, and white circle means the data is masked.

The attention mask approach is used in PLM to realize random sorting and predict word vectors in different sequence orders. On the one hand, PLM solves the shortcoming of AR language models with respect to contacting the contextual semantics; on the other hand, PLM uses a mask matrix to replace the [mask] tag in the BERT model while maintaining the dependency between multiple predictors. For example, for a given sequence (x_1, x_2, x_3, x_4,

x_5, x_6, x_7), for XLNet, to predict x_5 and x_3 the random sampling sequence can be selected as $(1, 4, 2, 3, 5, 7, 6)$. Then, the probability of x_5 and x_3 is $p(x_5|x_1, x_2, x_3, x_4) + p(x_3|x_1, x_2, x_4)$. When the model predicts the semantics of the word x_3, the information of x_5 does not affect the prediction of x_3. When the semantic information of x_5 is predicted, however, the model can use the semantic information of x_3, thereby solving the problem that in other models all of the semantic information is independent and cannot be connected. This makes PLM an important theoretical foundation for the excellent performance of XLNet.

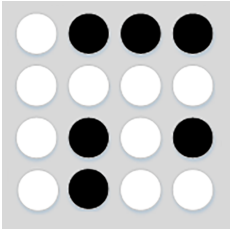


Figure 3. Mask matrix of the sequence (2,4,3,1).

2.2.2. Two-Stream Self-Attention Mechanism

While PLM can allow XLNet to relate to the context semantics, the original AR language model does not consider the location information of the words used in prediction. For example, given a sequence (x_1, x_2, x_3, x_4), one of the permutations is $(2, 4, 3, 1)$. When the predicted statement is x_3, its probability is $p(x_3|x_2, x_4)$. The second arrangement order is $(2, 4, 1, 3)$. When the predicted word is x_1, its probability is $p(x_1|x_2, x_4)$. At this time, the semantic equivalent probability of the two is predicted; in fact, however, the semantic content of x_1 and x_3 is largely different. The main reason for this problem is that AR language models predict words based on the content before the predicted word, and as such do not need to consider the position information of the words in the word order. On the other hand, the random sorting language mechanism requires the word order to be rearranged completely. After the position information of the words is disrupted, the model cannot determine the position information of the predicted words in the original sequence. Therefore, we proposed solving this problem using the two-stream self-attention mechanism.

The two-stream self-attention mechanism can solve the problem of the position of the target prediction word being ambiguous due to the random disordering of the word order by the random sorting language model. Two-stream attention consists of the content stream and query stream. The objective function of the traditional AR language model for a sequence with length T is shown in Equation (2):

$$P_\theta(X_{zt} = \mathrm{x}|x_{z<t}) = \frac{\exp\left(e(x)^{\mathrm{T}} h_\theta(x_{z<t})\right)}{\sum_{x'} \exp\left(e(x)^{\mathrm{T}} h_\theta(x_{z<t})\right)} \tag{2}$$

In the equation, z is the sequence obtained by full array random sampling from a sequence x with length T, zt represents the sequence number of the position of t in the sampling sequence, x is the word to be predicted, and $e(x)$ is the embedding of x. The content hidden state $h_\theta(x_{z<t})$ encodes the content of x and additionally encodes the above information, while not containing any location information, while $p_\theta(X_{z_t}|x_{z<t})$ demonstrates that for the prediction of words in position t in the sorting sequence, the probability is calculated from the words corresponding to the sequence number before the position of t. Because PLM disrupts the sequence order, it is necessary to "explicitly" add the location information of the word to be predicted into the original sequence, meaning that Equation (2) is updated as Equation (3):

$$P_\theta(X_{zt} = \mathrm{x}|x_{z<t}) = \frac{\exp\left(e(x)^{\mathrm{T}} g_\theta(x_{z<t}, z_t)\right)}{\sum_{x'} \exp\left(e(x)^{\mathrm{T}} g_\theta(x_{z<t}, z_t)\right)} \tag{3}$$

In the equation, z represents the query implicit state, including the words before t position and the position information of the word x to be predicted; it only encodes the context and location information of the prediction word x, and does not encode the content information of x. The updating processes of the content hidden state $h_\theta(x_{z<t})$ and query hidden state $g(x_{z<t}, z_t)$ are shown in Equations (4) and (5), respectively, where m represents the number of layers in the network layer. Usually, the query hidden state $g(0)$ is initialized as a variable w in layer 0 and the content hidden state $h(0)$ is initialized as the embedding of the word, that is, $e(x)$. The first layer of data is calculated according to layer 0 and then calculated layer by layer, with Q, K, and V being the results of linear transformation of the input data according to different weights.

$$h_{z_t}^{(m)} \leftarrow Attention\left(Q = h_{z_t}^{(m-1)}, KV = h_{z \le t}^{(m-1)}; \theta\right) \tag{4}$$

$$g_{z_t}^{(m)} \leftarrow Attention\left(Q = g_{z_t}^{(m-1)}, KV = h_{z \le t}^{(m-1)}; \theta\right) \tag{5}$$

The purpose of the two-stream self-attention mechanism is to obtain the location information of x_t without obtaining the content information when predicting a word x_t. For words other than x_t, the location information and content information should be provided. For example, if the given sequence is (x_1, x_2, x_3, x_4) and the sampling sequence is assumed to be $(3, 2, 4, 1)$, the probability of the word with sequence number 1 can be predicted and then its probability can be calculated according to x_3, x_2, and x_4. The working principle used for calculating the probability of x_1 using the content stream and query stream is shown in Figure 4.

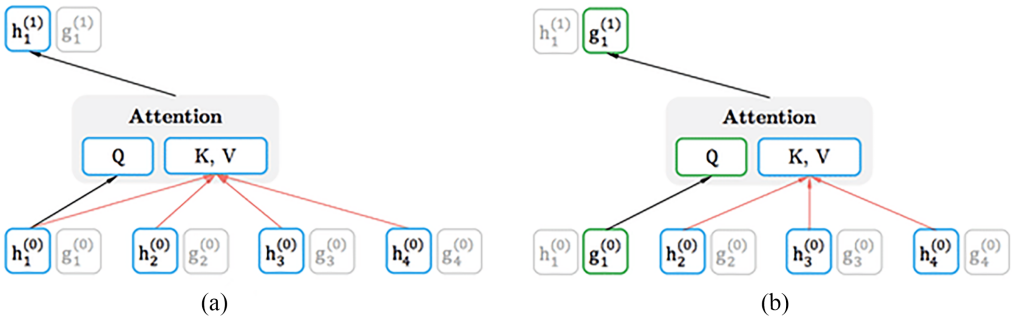


Figure 4. Schematic diagram of content stream and query stream: (**a**) content stream and (**b**) query stream.

As shown in Figure 4a, the content stream encodes both the context information, and the self-information of the predicted word. As shown in Figure 4b, the query stream encodes the location information of the predicted word and other content information with the exception of the self-information of the predicted word.

Figure 5 shows the principle of two-stream self-attention when the sampling order of the sequence (x_1, x_2, x_3, x_4) is $(3, 2, 4, 1)$. From the bottom to the top of Figure 5, starting from layer 0, the content stream h and query stream g are initialized by $e(x)$ and w, respectively. The first layer output $h(1)$ and $g(1)$ are respectively calculated through the content mask and query mask, then the second layer output is calculated in the same way. It can be observed from the mask matrix on the right that calculating the content stream is a standard transformer calculation process. When predicting the semantics of x_1, the semantic information of all words can be used, and when predicting the semantics of the word x_4, the semantic information of x_2, x_3, and x_4 can been used. In the content stream, the semantic information of the word itself can participate in predicting itself. When calculating the query stream, the attention mask functions somewhat differently. When predicting the semantic content of word x_1, its own content is masked, and the word is predicted only through the semantic content of x_2, x_3, and x_4. Similarly, when predicting x_2,

only use the semantic information of x_3 can be used. According to the above explanation, it can be deduced that in the query stream the information of the predicted word itself is not used for its own prediction. The difference between the content stream and query stream is whether the self-semantic information of the predicate itself.

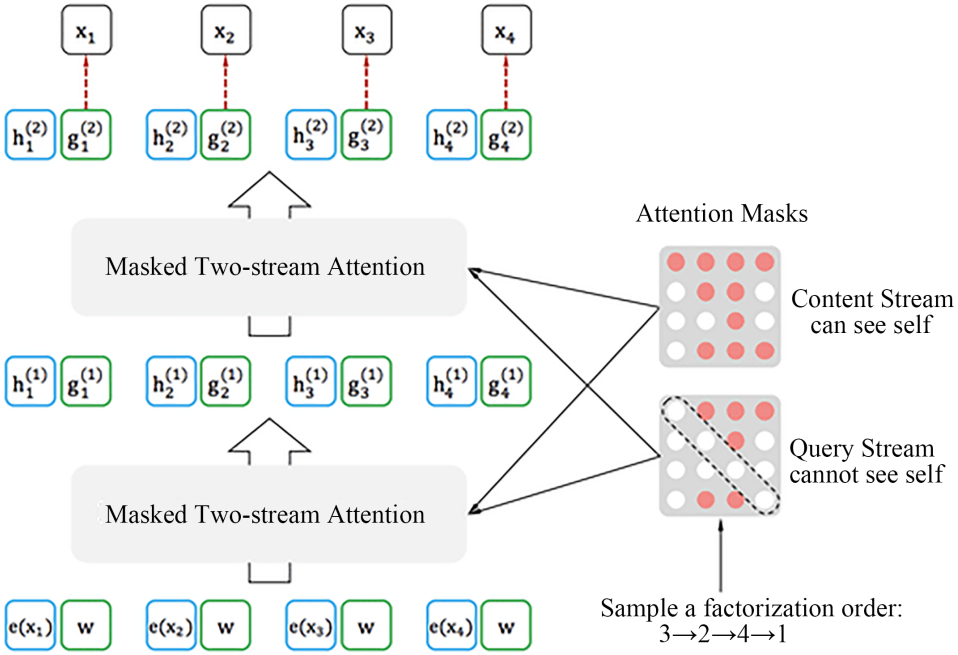


Figure 5. Schematic diagram of the two-stream self-attention mechanism.

2.3. Bi-Directional Gated Recurrent Unit

A gated recurrent unit (GRU) is a type of lightweight recurrent neural network that differs from other neural networks due to its internal gate structure. This unique structure enables the network to determine which data are relevant and which data can be discarded based on their relationship. This structure facilitates effective data transmission within the network while effectively controlling redundant information. As a result, the GRU partially addresses the issue of long-term dependence in neural networks. As shown in Figure 6, the structure of the GRU consists of three main components: the reset gate (r_t), update gate (z_t), and hidden state (h_t); these work together to extract temporal information and obtain long-term dependencies. In addition, h_{t-1} is the hidden state of the prior time point and $\tilde{h}_t$ is the candidate hidden state. The individual functions of these gates and hidden states are described in further detail below, while the structure of the GRU is displayed in Figure 6.

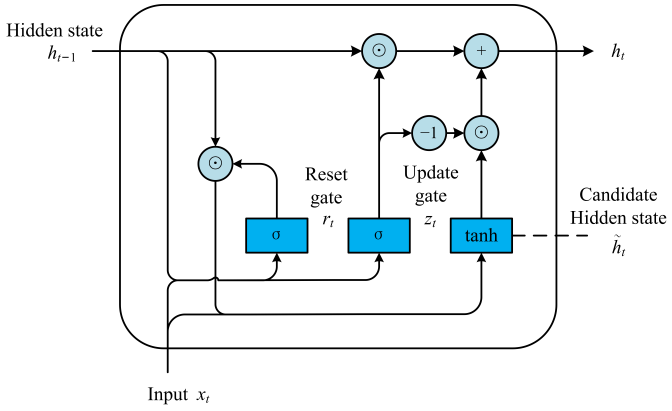


Figure 6. Structural diagram of a gated recurrent unit.

The input for the reset and update gates is obtained through the fully connected layer that operates on the current input x_t and the hidden state of the previous time point h_{t-1}. The reset gate r_t is calculated by Equation (6), and σ is a Sigmoid function which is used as an activation function. Thus, the result of the reset gate is a vector with a value between 0 and 1 which is similar to the gate in circuit. This value is then used to determine the relevancy of the hidden state of the previous time point. The candidate hidden state $\widetilde{h_t}$ is determined by the input x_t, reset gate r_t, and prior hidden state h_{t-1}, as shown in Equation (7); the prior hidden state contains the information of both the current time and previous time, and is an important component of the current hidden state, although it is not the real current hidden state.

$$r_t = \sigma(x_t w_{xr} + h_{t-1} w_{hr} + b_r) \tag{6}$$

In the above equation, t is the current time point, $t-1$ is the last time point, x_t is the input vector of the current time, h_{t-1} is the hidden state of the previous time point, w_{hr} and w_{xr} make up the weight matrix of the reset gate, and b_r is the bias of the reset gate.

$$\widetilde{h_t} = \tanh(x_t w_{xh} + (r_t \odot h_{t-1}) w_{hh} + b_h) \tag{7}$$

Here, $w_x h$ and $w_h h$ make up the weight matrix of the candidate hidden state and b_h is the bias of the candidate hidden state.

The update gate r_t is calculated using Equation (8) and σ is a Sigmoid function which is used as an activation function. Thus, the result of the reset gate is a vector with a value between 0 and 1, which is then used to determine the element of the previous hidden state that should be updated by the current candidate hidden state. For this reason, it is called the update gate.

$$z_t = \sigma(x_t w_{xz} + h_{t-1} w_{hz} + b_z) \tag{8}$$

In this equation, x_t is the current input vector, h_{t-1} is the hidden state of the previous time point, w_{xz} and w_{hz} make up the weight matrix of the update gate, and b_z is the bias of the update gate.

The hidden state is the output of the GRU calculated using the candidate hidden state and the previous hidden state in Equation (9):

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \widetilde{h}_t \tag{9}$$

Here, z_t is the update gate, h_{t-1} is the previous hidden state, $\widetilde{h_t}$ is the current candidate hidden state, and h_t is the current hidden state, which is the ouput of the GRU.

For the GRU, the output is only affected by the current input and the previous hidden state, and is not related to the subsequent state, making the GRU a unidirectional model. However, in order to extract effective features it is necessary to pay attention to the current information, the previous information, and the the subsequent information. It can be easily understood that combining these contexts makes it easier to understand textual information. Based on this idea, BiGRU is proposed to combine forward GRU and backward GRU, making the output result the combination of the two GRUs by weight matrix. The structure of BiGRU is presented in Figure 7; it is composed of an input layer, forward GRU, backward GRU, and output layer. In the figure, $\overrightarrow{h_t}$ is the hidden state of the forward GRU, $\overleftarrow{h_t}$ is the hidden state of the backward GRU, x_t represents the input vectors of BiGRU, h_t is the ouput hidden state of BiGRU, and $x_t \in R^{k \times d}$, where k is the batch size and d is the length of the input vectors. The forward hidden state $\overrightarrow{h_t} \in R^{k \times d}$ and the backward hidden state $\overleftarrow{h_t} \in R^{k \times d}$ are calculated using Equations (10) and (11).

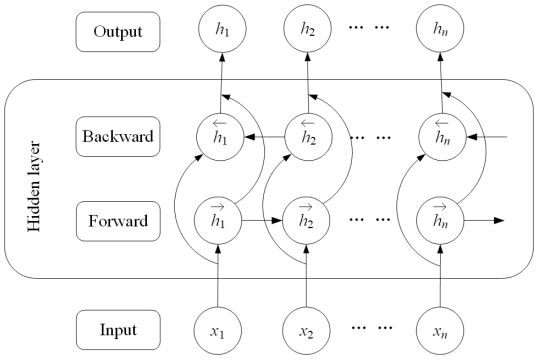


Figure 7. Structural diagram of the bidirectional gated recurrent unit [23].

$$\overrightarrow{h_t} = GRU(x_t, \overrightarrow{h_{t-1}}) \tag{10}$$

$$\overleftarrow{h_t} = GRU(x_t, \overleftarrow{h_{t+1}}) \tag{11}$$

In the above equations, GRU means the calculation process of the unidirectional GRU, x_t is the input vector, $\overrightarrow{h_{t-1}}$ is the hidden state of the forward GRU, and $\overleftarrow{h_{t+1}}$ is the hidden state of the backward GRU.

The final output hidden state of the BiGRU combines the outputs of the forward and backward GRUs using a weight matrix calculated by Equation (12):

$$h_t = w_{\underset{h_t}{\rightarrow}} \overrightarrow{h_t} + w_{\underset{h_t}{\leftarrow}} \overleftarrow{h_t} + b_t \tag{12}$$

In the equation, $w_{\underset{h_t}{\rightarrow}}$ and $w_{\underset{h_t}{\leftarrow}}$ make up the weight matrix of the output layer and b_t is the bias of the output layer.

3. Results

The XLNet-BiGRU-Att model was compared with the BERT-BiLSTM, BERT, and XLNet emotional analysis models. The parameter settings of each model are shown in Table 1. In the table, Batch Size is the amount of data for each parameter update, Epochs is the number of iterations in the training process, Embedding is the word vector dimension, and Lr is the learning rate set by the model. The parameters for XLNet-BiGRU-Att proposed in this paper are provided at the bottom of the table.

Table 1. Values of model parameters.

Model	Batch Size	Epochs	Embedding	Lr
BERT-BiLSTM	128	50	100	0.001
XLNet	256	50	100	0.001
Bert	64	300	768	0.001
XLNet-BIGRU-Att	256	300	768	0.001

Batch size is an important parameter for the training process, for which sizes of 2, 4, 8, 16, 32, 64, 128, 256, 512…, are typically used. Within the appropriate range, the larger the batch is, the more accurate the descent direction is. Smaller batch sizes cause larger vibration, while larger batch sizes reduce the convergence speed, meaning that more epochs are needed, and may cause memory overflow. Convergence experiments with different batch sizes of 64, 128, and 256 were conducted; the learning curve is shown in Figure 8. It can be seen that the learning curve has the highest vibration with a batch size of 64. Although it reaches the minimum loss very fast, the loss begin to increase after 100 epochs. Therefore the training process with a batch size of 64 is not stable over 400 epochs. When the batch size is 128, the loss decreases slowly, and the model does not reach the best state

after 400 epochs. When the batch size is 256, the learning curve reaches convergence in about 250 epochs, and the vibration is smaller than with batch sizes of 64 or 128. With a batch size of 512 the training process would cause memory overflow. Thus, the best parameters are a batch size of 256 and 300 epochs.

Increasing the dimension of embedding can enhance the richness and meticulousness of text representation to an extent. The embedding dimension of BERT has two versions, 768 and 1024. In order to ensure a fair comparison with BERT, the same embedding dimensions were selected for XLNet-BiGRU-Att. As the 1024 version has three times the parameters of the 768 version, the latter was selected in order to reduce the computational burden.

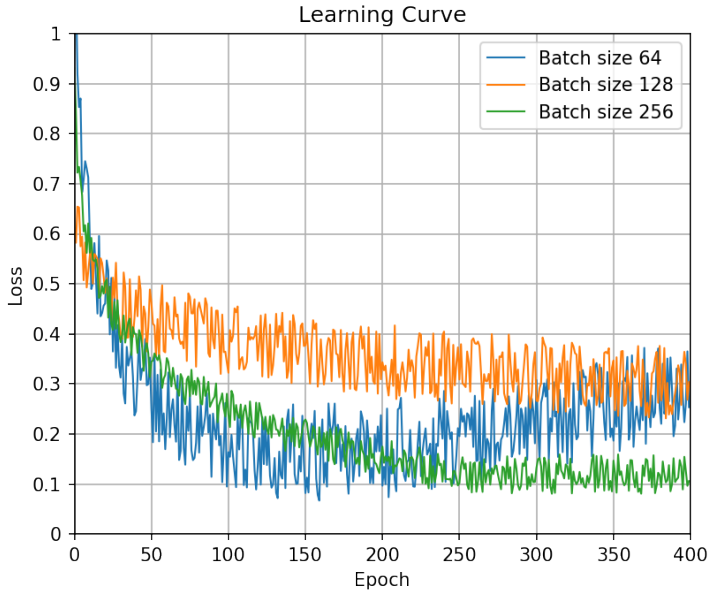


Figure 8. Learning curves with different batch sizes.

BERT and XLNet are the basic models which are usually used for text representation, and their results can be directly used for emotion classification with a fully connected network and softmax function. Another approach is to add feature extraction models on top of BERT or XLNet for classification, such as BERT-BiLSTM and the XLNet-BiGRU-Att model proposed in this paper. Therefore, the experimental performance of BERT, XLNet, BERT-BiLSTM, and XLNet-BiGRU-Att were compared on the two datasets introduced in Section 2.1. The experimental results on IEMOCAP are shown in Table 2, and a bar chart is used for visual comparison in Figure 9.

Table 2. Evaluation results of different models on the IEMOCAP dataset.

	Accuracy	**Precision**	**Recall**	**F1-Measure**
BERT-BiLSTM	87.92	87.25	86.26	86.75
XLNet	88.64	88.35	87.52	87.93
Bert	85.86	86.11	85.64	85.87
XLNet-BIGRU-Att	91.71	89.23	89.24	89.23

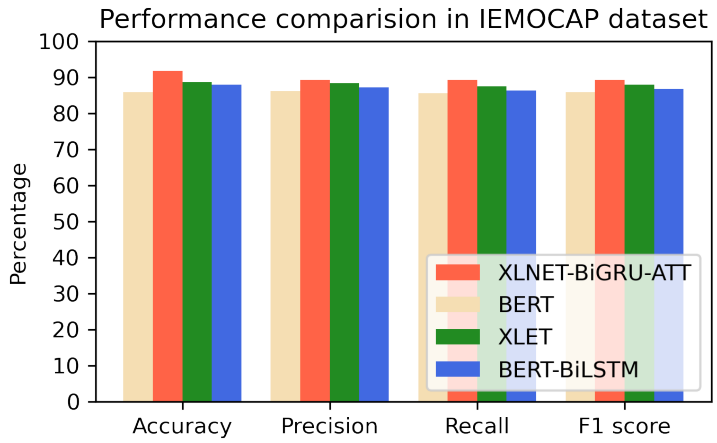


Figure 9. Performance comparison on the IEMOCAP dataset.

The same experiments were carried out on the CASIA dataset. The detailed results are shown in Table 3, and a bar chart is shown in Figure 10.

Table 3. Evaluation results of different models on the CASIA dataset.

	Accuracy	Precision	Recall	F1-Measure
BERT-BiLSTM	81.92	79.91	83.64	81.73
XLNet	83.64	79.75	82.41	81.05
Bert	79.86	76.27	80.43	78.29
XLNet-BIGRU-Att	85.71	80.60	84.60	82.55

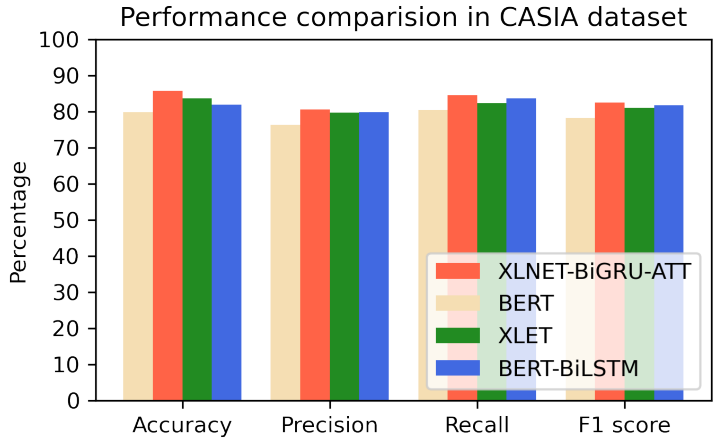


Figure 10. Bar chart showing the evaluation results of different models on the CASIA dataset.

From the experimental results, it can be seen that the pre-trained models can calculate effective word vectors for TER, and they all perform well on the both datasetsto calculate the word vector. However, compared to the others, the model proposed in this paper has the best accuracy on the IEMOCAP and CASIA datasets, reaching 91.17% and 85.71%, respectively.

4. Discussion

Upon examining the respective datasets, it is apparent that the IEMOCAP dataset provides superior performance results in comparison to the CASIA dataset. IEMOCAP contains nine distinct emotions: Neutral, Happy, Sad, Angry, Surprise, Fear, Disgust, Excited, and Other. Nonetheless, the distribution of sample categories in the nine-emotion dataset is noticeably imbalanced, with the majority of samples belonging to the four

fundamental emotions (Happy, Sad, Angry, and Neural). As a consequence, many academic studies concentrate solely on these basic emotions. To enable a more accurate comparison of the results, this paper similarly focuses on the four basic emotions within the IEMOCAP dataset. On the contrary, the CASIA dataset comprises six emotions, thereby posing a greater challenge for model classification. Hence, based on the inherent difficulty of the classification problem, it is established that the IEMOCAP dataset yields more favorable results than the CASIA dataset.

The experiment primarily focused on a comparison of two models, BERT and XLNet, with regard to text emotion recognition. The absolute values of the results indicate that both models are proficient in this area. However, the relative results suggest that XLNet outperforms BERT due to its employment of attention, larger hidden layers, and additional positional encoding elements. On the other hand, BERT-BiLSTM primarily prioritizes optimization of the bidirectional LSTM network structure, attention mechanism, and long-term dependency modeling, resulting in better performance than BERT on related tasks. Our experiments indicate that the performance of BERT-BiLSTM is comparable to that of XLNet.

Although both BERT and XLNet perform well on tasks involving semantic representation of textual information, our experiments show that their performance when directly applying semantic representation results to emotion classification is not as good as that attained by models with additional feature extraction capability. For example, BERT-BiLSTM performs better than BERT on both datasets; similarly, XLNet-BILSTM-Att performs better than XLNet. Therefore, the above experiments demonstrate that the semantic representations of BERT and XLNet involve general natural language understanding. When dealing with specialized tasks such as emotion recognition, adding specific feature extraction modules can improve the performance of models.

This paper presents enhancements to XLNet achieved by incorporating bidirectional gate recurrent units and an attention masking mechanism into its architecture. The bidirectional transmission feature of BiGRU serves to effectively process long sequence data, making the model better suited for text mining applications. Additionally, the attention mask mechanism connects contextual semantics and maintains the dependency of multiple word predictions, which improves overall model accuracy. The performance of XLNet is boosted when supported by bidirectional gate recurrent units and attention masking mechanisms, as shown by our experimental results. Specifically, the proposed XLNet-BiGRU-Att model achieved the highest accuracy of all tested models, reaching at 91.71% on the IEMOCAP dataset and 85.71% on the CASIA dataset.

5. Conclusions

Text emotion recognition is a crucial aspect of natural language understanding that finds extensive application in domains such as human–computer interaction, public opinion analysis, mental health analysis, and social network analysis. The development of advanced technologies in the fields of computing, the internet, and artificial intelligence has considerably amplified the significance of this field. In this paper, the XLNet-BiGRU-Att model is proposed, building on XLNet by leveraging BiGRU and Att to achieve enhanced performance. This paper has presented and analyzed the structure of the model and described the experiments we carried out on the IEMOCAP and CASIA datasets. The results show that our proposed XLNet-BiGRU-Att model outperforms BERT, BERT-BiLSTM, and XLNet, achieving an accuracy of 91.71% on IEMOCAP and 85.71% on CASIA. The theoretical and experimental analysis and discussion presented in this paper confirm the suitability of XLNet-BiGRU-Att for text emotion recognition.

Author Contributions: Methodology, T.H.; Writing—original draft preparation, Z.Z.; Conceptualization, C.D.; Data curation, M.R.; Supervision, X.J.; Software, Q.Z.; Project administration, T.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Jinhua Science and Technology Bureau, grant number 2022-1-046, and by the Jinhua Advanced Research Institute, grant number G202209 and G202207.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Publicly available datasets were analyzed in this study. The data can be found at: http://www.chineseldc.org/resource_info.php?rid=76 and https://sail.usc.edu/iemocap/iemocap_release.htm (accessed on 1 January 2023).

Acknowledgments: This research was supported by the Jinhua Science and Technology Bureau and the Jinhua Advanced Research Institute.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Acheampong, F.A.; Nunoo-Mensah, H.; Chen, W. Transformer models for text-based emotion detection: A review of BERT-based approaches. *Artif. Intell. Rev.* **2021**, *54*, 5789–5829. [CrossRef]
2. Batbaatar, E.; Li, M.; Ryu, K.H. Semantic-emotion neural network for emotion recognition from text. *IEEE Access* **2019**, *7*, 111866–111878. [CrossRef]
3. Alswaidan, N.; Menai, M.E.B. A survey of state-of-the-art approaches for emotion recognition in text. *Knowl. Inf. Syst.* **2020**, *62*, 2937–2987. [CrossRef]
4. Yang, X.; Feng, S.; Wang, D.; Zhang, Y. Image-text multimodal emotion classification via multi-view attentional network. *IEEE Trans. Multimed.* **2020**, *23*, 4014–4026. [CrossRef]
5. Nandwani, P.; Verma, R. A review on sentiment analysis and emotion detection from text. *Soc. Netw. Anal. Min.* **2021**, *11*, 81. [CrossRef] [PubMed]
6. Yao, T.; Zhai, Z.; Gao, B. Text classification model based on fasttext. In Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Information Systems, Tianjin, China, 20–22 March 2020.
7. Al-Shehari, T.; Alsowail, R.A. An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniques. *Entropy* **2021**, *23*, 1258. [CrossRef] [PubMed]
8. Sundaram, V.; Ahmed, S.; Muqtadeer, S.A.; Reddy, R.R. Emotion analysis in text using TF-IDF. In Proceedings of the 11th International Conference on Cloud Computing, Data Science & Engineering, Nodia, India, 28–29 January 2021.
9. Cahyani, D.E.; Patasik, I. Performance comparison of tf-idf and word2vec models for emotion text classification. *Bull. Electr. Eng. Inform.* **2021**, *10*, 2780–2788. [CrossRef]
10. Khan, R.; Qian, Y.; Naeem, S. Extractive based text summarization using K-means and TF-IDF. *Int. J. Inf. Eng. Electron. Bus.* **2019**, *10*, 33.
11. Li, J.; Xu, Y.; Shi, H. Bidirectional LSTM with hierarchical attention for text classification. In Proceedings of the IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference, Chengdu, China, 20–22 December 2019.
12. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338. [CrossRef]
13. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
14. Gonzalez, J.A.; Hurtado, L.F.; Pla, F. TWilBert: Pre-trained deep bidirectional transformers for Spanish Twitter. *Neurocomputing* **2021**, *426*, 58–69. [CrossRef]
15. Acheampong, F.A.; Wenyu, C.; Nunoo-Mensah, H. Text-based emotion detection: Advances, challenges, and opportunities. *Eng. Rep.* **2020**, *2*, e12189. [CrossRef]
16. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 5753–5763
17. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
18. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
19. Kumar, P.; Raman, B. A BERT based dual-channel explainable text emotion recognition system. *Neural Netw.* **2022**, *150*, 392–407. [CrossRef] [PubMed]
20. Sams, A.S.; Zahra, A. Multimodal music emotion recognition in Indonesian songs based on CNN-LSTM, XLNet transformers. *Bull. Electr. Eng. Inform.* **2023**, *12*, 355–364. [CrossRef]
21. Li, Y.; Tao, J.; Chao, L.; Bao, W.; Liu, Y. CHEAVD: A Chinese natural emotional audio–visual database. *J. Ambient. Intell. Humaniz. Comput.* **2017**, *8*, 913–924. [CrossRef]

22. Yu, Y.; Kim, Y.J. Attention-LSTM-attention model for speech emotion recognition and analysis of IEMOCAP database. *Electronics* **2020**, *9*, 713. [CrossRef]
23. Han, T.; Zhang, Z.; Ren, M.; Dong, C.; Jiang, X.; Zhuang, Q. Speech Emotion Recognition Based on Deep Residual Shrinkage Network. *Electronics* **2023**, *12*, 2512. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

An Improved Unscented Kalman Filtering Combined with Feature Triangle for Head Position Tracking

Xiaoyu Yu [1,2], Yan Zhang [2], Haibin Wu [2,*] and Aili Wang [2,*]

[1] College of Electron and Information, University of Electronic Science and Technology of China, Zhongshan Institute, Zhongshan 528402, China
[2] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China
* Correspondence: woo@hrbust.edu.cn (H.W.); aili925@hrbust.edu.cn (A.W.)

Abstract: Aiming at the problem of feature point tracking loss caused by large head rotation and facial occlusion in doctors, this paper designs a head-position-tracking system based on geometric triangles and unscented Kalman filtering. By interconnecting the three feature points of the left and right pupil centers and the tip of the nose, they form a coplanar triangle. When the posture of the doctor's head changes due to rotation, the shape of the corresponding geometric triangle will also deform. Using the inherent laws therein, the head posture can be estimated based on changes in the geometric model. Due to the inaccurate positioning of feature points caused by the deflection of the human head wearing a mask, traditional linear Kalman filtering algorithms are difficult to accurately track feature points. This paper combines geometric triangles with an unscented Kalman Filter (UKF) to obtain head posture, which has been fully tested in different environments, such as different faces, wearing/not wearing masks, and dark/bright light via public and measured datasets. The final experimental results show that compared to the linear Kalman filtering algorithm with a single feature point, the traceless Kalman filtering algorithm combined with geometric triangles in this paper not only improves the robustness of nonlinear angle of view tracking but also can provide more accurate estimates than traditional Kalman filters.

Keywords: head position estimation; viewpoint tracking; unscented Kalman filter; Kinect

1. Introduction

With the rise of the era of artificial intelligence, there are many ways to realize human–computer interaction in the virtual world. The traditional mouse operation and keyboard input can no longer meet people's needs. Virtual Reality (VR) is developing multimedia and multi-channel direction, such as gesture interaction, body interaction, behavior capture, tactile anti-touch, eye movement tracking, voice response, etc. [1]. Among them, head-position-tracking technology is an important research topic in the VR field.

Visual angle tracking generally refers to the process of automatically detecting the relative position of the human pupil or estimating the direction of vision. However, the eye movement process is always accompanied by various kinds of noise from the external environment, such as external light interference, and mask occlusion, which makes the timing positioning of the eye tracking target error. There is also noise accompanied by the eye movement itself, such as the large deflection of the head, eye shaking, blinking, and other unconscious actions, which make the pupil-tracking algorithm need to be fast and robust. The purpose of visual-angle-tracking technology is to predict the target position and pose (position, rotation angle, motion speed, acceleration, and direction, etc.) of the following frames based on statistical or learning methods according to the initial state and characteristics of the eyes in the tracked human head posture and finally draw the motion trajectory of the tracked object in the entire video sequence.

Scholars at home and abroad have been engaged in the research of perspective tracking for decades, and a large number of excellent tracking algorithms and evaluation systems have emerged, aiming to design a tracker with good robustness and high accuracy. From the perspective of the development of perspective-tracking technology, the current perspective-tracking system is mainly divided into an invasive perspective-tracking system and a non-invasive perspective-tracking system [2]. In the intrusive system, users need to wear optoelectronic auxiliary equipment designed by the developer. Using the movement information of the head and eyeball collected by this equipment, the geometric model of the direction of the human eye line of sight is established to estimate the focus position of the human eye line of sight. Invasive visual-angle-tracking systems generally require relatively expensive hardware (similar to glasses and helmets) to be fixed on the human head.

Marquard [3] proposed the direct observation method to realize eye movement tracking, which aims to reduce patient recognition errors using eye movement tracking data and direct observation mode. MacNeil [4] described a method to estimate the corrected EOG signal by calibrating it with the industry-standard pupillary corneal reflex (PCR) eye tracker. In the same year, Katona [5] used an eye tracker to understand the code via the evaluation based on eye tracking, analyze the gaze route, and measure and evaluate the eye movement parameters to determine the readability and understandability of the code. With the eye movement instrument also playing a huge role in medicine, Fabio [6] and others used eye movement instruments for cognitive rehabilitation to analyze patients' attention, which studied the longitudinal effect of cognitive rehabilitation on patients. Recently, Chugh S. [7] developed a complete convolution neural network based on U-Net architecture, which solved the detection and matching problems when there are false and missing reflections. This is a significant improvement on the most advanced learning-based XR eye-tracking system, which reports 2–3° fixation errors.

Although eye trackers are widely used at present, there are many problems. For example, a complete set of eye trackers is expensive, and the construction environment is complex, which is not suitable for the perspective tracking of doctors' surgery. With the continuous emergence of emerging technologies, artificial intelligence, pattern recognition, and other new technologies are providing more possibilities for perspective-tracking technology [8], especially in the field of medical imaging, which can play an auxiliary role in doctors' surgery or assist patients with mental disorders via non-invasive technology.

In medicine, the commonly used method for tracking the position of the head is to use single or multiple sensors [9,10] in this positioning system and sample these sensors to measure the changes in the rotation of the doctor's head. The measurement quantities in this method are usually the size of the magnetic field, the reduction value of the light source projected by the camera after collection, and the inertia value of the sensor. These changes are calculated and demonstrated to obtain the translational and rotational information values of the target object [11,12].

These methods usually require the placement of equipment near the doctor's head or operating table, and the accuracy and robustness of tracking are subject to the resolution and accuracy of the equipment. Therefore, how to obtain the doctor's head position and posture with the help of a small amount of external equipment on the basis of existing medical equipment and without affecting the normal operation of the doctor and apply it to minimally invasive surgery has been a research hotspot for scholars at home and abroad in recent years.

In order to accurately obtain the position and posture of the human head, many scholars have conducted research on this. According to the different implementation methods in positioning, domestic and foreign scholars' research can be mainly divided into two parts. One is the head-positioning technology based on sensors, which obtains the position information of the doctor or patient by arranging sensor devices on their body, thereby achieving target tracking; The second is based on visual-head-positioning

technology, which mainly requires external devices such as Microsoft's Kinect [13–15]. The camera captures facial images and processes the images to achieve target tracking.

Non-invasive eye tracking based on video images has caused a lot of in-depth research in academia and industry [16]. New instruments and equipment, such as high-resolution image acquisition equipment and external light source, are increasingly applied to the information acquisition of the line-of-sight-tracking system. New technologies such as image processing, pattern recognition, and artificial neural network provide more feasible methods for line-of-sight tracking, which is moving towards a non-invasive, portable, and low-cost trend [17].

Some researchers rely on non-contact eye image acquisition equipment for line-of-sight tracking research. Zhang et al. [18] developed a simplified 3D line-of-sight tracking technology with stereo vision. Compared with other 3D systems, this paper uses the first image of two LEDs to estimate the 3D corneal center and pupil center. Secondly, the fixation point on the screen is calculated by intersecting with the estimated 3D gaze. Li's method based on depth sensor usually uses Kinect to obtain depth information and converts the POR from the center of the eye, and performs 3D gaze estimation under natural head motion, where the center of the eye is calculated using the inner eye angle as the anchor point for head posture tracking [19]. Although the model is simple, the proposed screen calibration method is too complex. Zanetti et al. [20] used Kinect equipment to reduce the line-of-sight tracking error based on pupil–cornea reflection in 3D scenes and improve the accuracy of the subject's eye feature tracking under infrared light irradiation. Elmadjian [21] revealed the limitations of the widely used 3D gaze estimation technology and proposed a new calibration procedure, which uses the uncalibrated head-worn binocular eye tracker and RGB-D camera to track the 3D gaze within the scene volume. The results show that although the interaction technology with a focus on gaze should be realized in 3D, it is still a challenge to accurately estimate in this setting.

In the past three years, Gonzal et al. [22] proposed two gaze area estimation modules integrated into the driving simulator. One uses the 3D Kinect device, and the other uses the virtual reality Oculus Rift device, which can detect the driver's gaze area in each route processing frame of the seven areas divided by the driving scene. For the stereo environment, Han et al. [23] used the 3D vision estimation method provided by Intel Realsense 3D camera for the 3D environment. By combining Move Eye Tracking (MET) with Visual Simultaneous Localization and Mapping (VSLAM) technology, we can capture individuals' gazes in open spaces and irregular viewing paths. Liu developed [24] a 3D model-based line-of-sight tracking method based on a single camera and single light source system, which uses iris features instead of pupil features used in most traditional model-based 3D gaze-tracking systems, which is simpler than the system configuration required by existing technical methods. Because the position of the fixation point in the perspective tracking requires the depth information of the face, Chen Qin [25] and others selected the RealsenseD455 depth camera to obtain RGB images and depth maps and used the three-dimensional TFGazeNet and SSRGazeNet neural networks to track the perspective, and the experiments verified the feasibility of the algorithm. However, the camera is expensive, and other depth cameras with the same function can be considered to replace it, and the perspective-n-points (PNP) method can be used to obtain the face position information.

The Kalman filter is widely used in the field of moving target detection [26]. The position and direction of the objects depend on the Kalman filter, extended Kalman filter, and unscented Kalman filter to reduce the error [27]. Sultan M.S. [28] combines robot kinematics, vision, and force sensors to independently perform hand–eye coordination tasks, and the proposed method accurately determines the camera pose of single-view objects. The corners are indirectly detected from the intersection of lines, and the lines are obtained using RANSAC (Random Sample Consumus) algorithm. The final pose correction is improved by the Kalman filter to gradually update. Munir F. [29] proposed an extension of spatiotemporal context learning based on Kinect and Kalman filtering. Spatiotemporal context learning provides the most advanced accuracy in general object tracking, but its

performance will be affected by object occlusion. By combining with the Kalman filter, the proposed method can simulate the dynamics of eye movement and provide reliable eye movement tracking under occlusion. Pan [30] designed a combination of a convolution neural network and Kalman filter to estimate and analyze the real-time changes in eye position. Kalman filter is used to track the human eye and eliminate jitter interference. In recent years, Garapati [31] has combined a modified gain extended Kalman filter (MGEKF) with a particle filter to improve the performance of the filter. Huang et al. [32] proposed a view-tracking algorithm based on a multi-model Kalman filter, which uses a multi-model Kalman filter to improve the efficiency of eye tracking in images and uses multiple models to estimate objects. The algorithm consists of two parts. The first step is to identify the initial position of the eyes using Support Vector Machine (SVM). In the second part, the multi-model Kalman filter is used to predict the eye position in the next frame, which is based on the constant speed and acceleration model of normal people.

The angle-of-view tracking method using Kalman filtering for the above algorithm is a linear system estimation algorithm. In a real visual-angle-tracking environment, head and eye movements have a high degree of nonlinearity of the likelihood model, and standard Kalman filters are no longer optimal in real environments. Feng [33] proposed a new improved UKF based on Double Unscented Transformation (DUT), starting with the inherent mechanism of UKF and the dynamic state model of the human hand, with the goal of improving the accuracy of human hand tracking. Zutao et al. proposed a new driver fatigue detection scheme based on nonlinear unscented Kalman filtering and eye tracking. Assuming that the probability distribution ratio approximates any nonlinear function or transformation, unscented transformation (UT) can be used to achieve nonlinear eye tracking, which uses a set of deterministic sigma points to match the posterior probability density function of eye movement. Hannuksela J. [34] studied human head posture estimation based on facial features. Special consideration is given to facial feature extraction methods suitable for real-time systems. The extended Kalman filtering (EKF) framework is used to track and estimate the three-dimensional posture of the rotating head in an image sequence. During the tracking process, the correspondence between the rigid head model and the three extracted facial features (eyes and mouth) is used to solve the pose. However, there are some limitations in the current implementation. Users who wear glasses, occlusion, and shadows can cause system problems. Bankar R. et al. [35] described performance improvements in head gesture recognition. Firstly, an improved Camshift-based face-tracking algorithm is used to improve head gesture recognition. UKF is used to suppress noise and correct the shortcomings of inaccurate positioning. To overcome the shortcomings of existing gesture-tracking methods in terms of accuracy and speed, Du X. proposed a new YOLOv4 model that combines Kalman filtering with real-time hand-tracking methods. The new algorithm can solve the problems of detection speed, accuracy, and stability in hand-tracking technology. The YOLOv4 is used to detect the current frame tracking target, and Kalman filtering is applied to predict the next position and boundary box size of the target based on the current position of the target. Tian L. et al. proposed [36] Consistent Extended Kalman Filtering (CEKF) for maneuvering target tracking (MTT) with nonlinear uncertain dynamics and applied it to manual position tracking.

In summary, the previous algorithm has two problems: (1) combining linear Kalman filtering with eye motion tracking by assuming that the human eye motion is with uniform motion mode; (2) Most visual-angle-tracking systems aim at the problem of errors in positioning and tracking a single feature point. Different visual-angle-tracking algorithms are proposed based on real-time images of the left and right eyes, respectively.

This paper introduces a head-position-tracking system, which uses a combination of geometric triangles and unscented Kalman filtering to analyze and predict head and eye movement information for the next frame. Firstly, aiming at the practical needs of doctors' surgical scenarios and limitations such as data collection, a simulation platform for operating room perspective tracking was built using Kinect v2 and combining different organ and soft tissue models. Secondly, a doctor's head posture model is established using

the geometric triangle method, using three relatively stable feature points: the nose and the center of the pupil, and the relative position changes with the change of the head posture, presenting certain geometric features to convert the change of the head posture into a geometric triangle deformation. Finally, the most suitable head position orientation is obtained by combining the unscented Kalman filter with geometric triangle tracking to track the doctor's visual angle based on the position and pose of the previous moment and the weight of the feature point position and pose at the current moment. The performance of the head-position-tracking system is evaluated qualitatively and quantitatively.

2. Materials and Methods

This section describes the overall architecture and methods of the proposed eye-movement-tracking system, using triangular models to obtain head posture, eye movement information, and unscented Kalman filtering for perspective prediction and tracking.

The geometric relationship of the head in the perspective-tracking system is shown in the Figure 1, R(camera–world) represents the matrix of the world coordinate system under the camera coordinate system, R(world–head) represents the posture matrix of the doctor's head relative to the world coordinate system, and R(camera–head) represents the posture matrix of the doctor's head under the camera coordinate system.

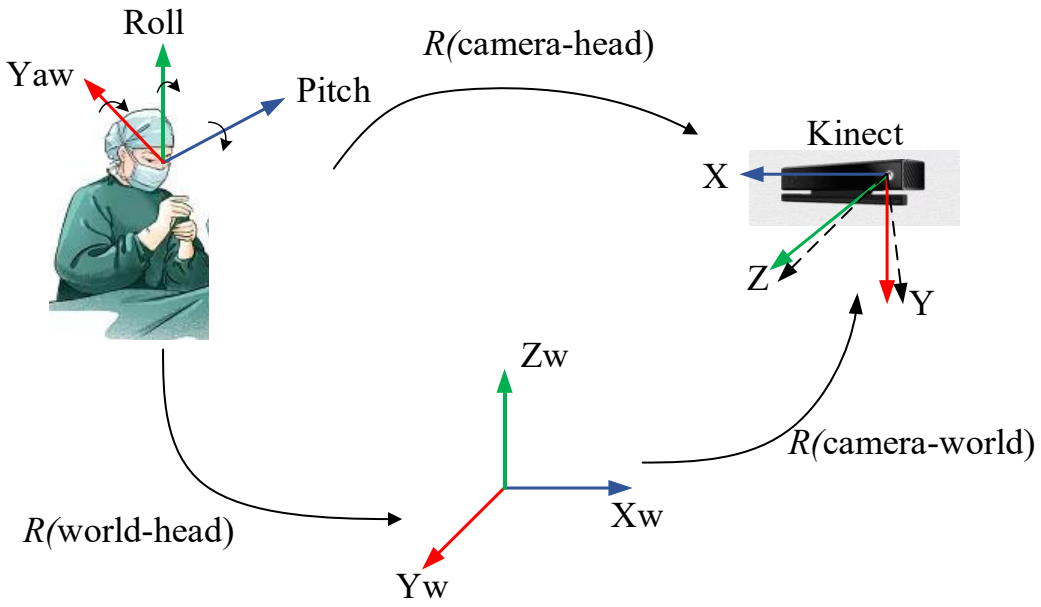


Figure 1. Geometric relations in head pose estimation.

In order to more intuitively describe the doctor's perspective-tracking system, this article has built an operating room perspective-tracking simulation platform, as shown in Figure 2. The entire platform includes Kinect v2 sensors to obtain doctor's head information, patient's body surface and internal cavity, and surgical forceps. Due to the non-invasive head position tracking used in this article, doctors do not need to wear complex and heavy equipment on the head, which is convenient for doctors to operate. This method can achieve both single feature point tracking and multiple feature point tracking. Faced with the problem of losing tracking of a certain frame in the video, it is not necessary to recalibrate and is more suitable for long-term perspective-tracking research [37].

The software module is responsible for performing different tasks, mainly including the following sub modules: face detection and feature point localization, triangular model acquisition of head posture, and unscented Kalman filter prediction and tracking.

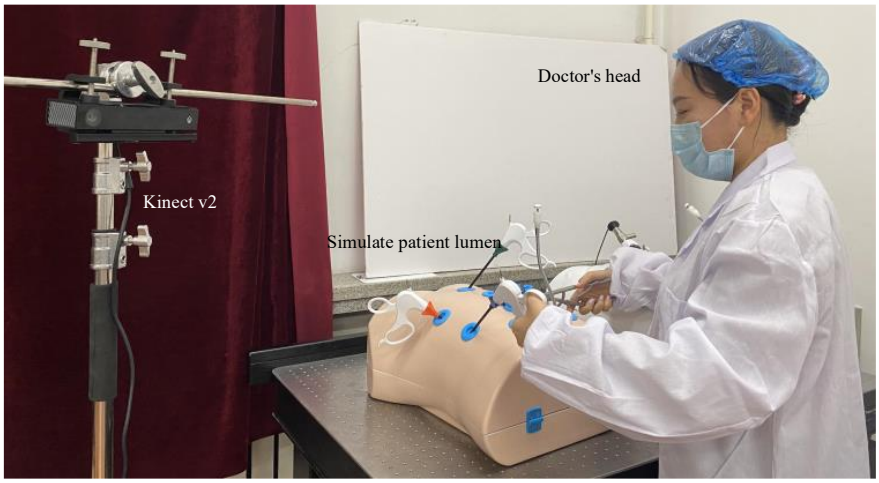


Figure 2. Construction of head-position-tracking simulation platform in operating room.

In this paper, the geometric model method is used to estimate the head posture. A doctor's head posture model is established using a geometric triangle method, using three relatively stable feature points of the nose and pupil center, and their relative positions change with the head posture, presenting certain geometric features. Therefore, after locating the eyes and mouth, the estimation of the head posture is achieved. The flow chart of the proposed head position tracking is shown in Figure 3. Firstly, the current measurement value of Kinect was used as input, and the facial information of doctors with different gestures was obtained via Adaboost face detection algorithm combined with MB-LBP features. The angle-positioning model of feature triangle was analyzed and established via ASM feature-point-positioning method and pixel coordinates combined with depth information to solve the problem that the head position and posture of doctors wearing masks change frequently, resulting in the loss of feature point tracking. Finally, combining the head-attitude-positioning algorithm of feature triangle and the tracking algorithm of untraced Kalman filter, the real-time prediction and tracking of the doctor's perspective during surgery are carried out, and the predicted value of the next frame is output.

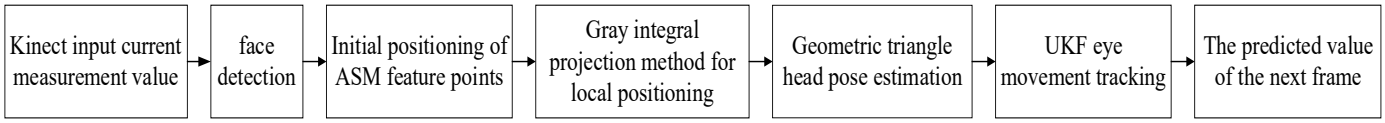


Figure 3. The flow chart of the proposed head position tracking.

2.1. Location of Left and Right Pupil and Nasal Tip Feature Points

In view of the complex surgical environment of doctors, ASM and AAM algorithms are used to obtain 68 feature points of the face. Due to the excessive computation of feature points, which can increase the speed of program operation, it is necessary to reduce the number of feature points in order to analyze and experiment with the real-time stability of feature points. Therefore, based on the global ASM model [38] localization, local secondary localization is performed for this region to obtain three feature points, namely, the center of the pupil and the tip of the nose.

For the positioning of the tip of the nose, the gray value of the lower nostril generally differs greatly from the neighbors. Therefore, the approximate region of the tip of the nose is determined using the gray integration projection method of the nostril region [39]. The grayscale integral projection is a projection distribution feature of the grayscale value of an image in certain directions. It is set $F(x, y)$ as the grayscale value of the pixel (x, y) in the

image, and the size is M × N. The horizontal and vertical grayscale integration functions of an image are shown in Equations (1) and (2), respectively, as follows:

$$H(y) = \sum_{x=0}^{N-1} F(x,y) \tag{1}$$

$$V(x) = \sum_{x=0}^{M-1} F(x,y) \tag{2}$$

Typically, the nostrils have the lowest brightness values in the approximate area of the nose. Therefore, first, a simple graying process can be performed on the color image of the nose region to convert it into a grayscale image that reflects the brightness level; Then, convert the grayscale image into a binary image to find the location of the nostrils.

If the set of all pixels in the left nostril region is X which contains N (X) pixel points. The centroid of the left nostril region can be expressed as (x_i, y_i), where $x_i = (\sum_{x \in X} x)/N(X)$, $y_i = (\sum_{y \in X} y)/N(X)$. Similarly, the centroid of the right nostril region is (x_r, y_r).

If the lateral distance between the left and right nostril points is d, it is generally believed that the tip of the nose is located at $d/2$ above the middle perpendicular of the two nostrils. Let the midpoint of the left and right nostril points be (x_o, y_o), where $x_o = (x_i + x_r)/2$, $y_o = (y_i + y_r)/2$ and then $d = x_r - x_i$. The coordinates of the center of the nostril are (x_m, y_m), where $x_m = x_o$, $y_m = d/2$.

The integral projection method is also used for locating the center of the pupil. According to the gray distribution characteristics of the left and right eye images, the boundary of the binocular region is extracted, and the binocular region is demarcated separately to achieve independent pupil localization. Figure 4a,b are grayscale integrated projections in the vertical and horizontal directions, respectively. As the gray values of both the eyebrow and eye regions are lower than the gray values of their surrounding regions, it can be seen from Figure 4a that the first valley bottom corresponds to the eyebrow region, and the second valley bottom corresponds to the pupil region. As the gray value of the pupil is lower than the gray value of its surrounding area, it can be seen from Figure 4b that the first valley corresponds to the left pupil area, and the second valley corresponds to the right pupil area.

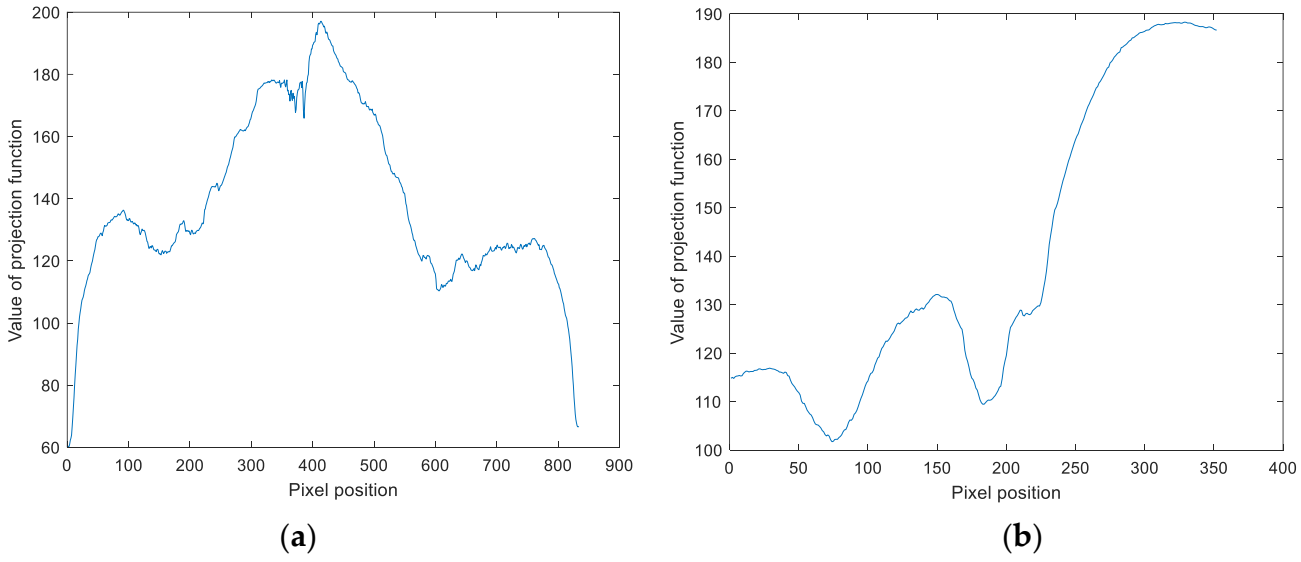
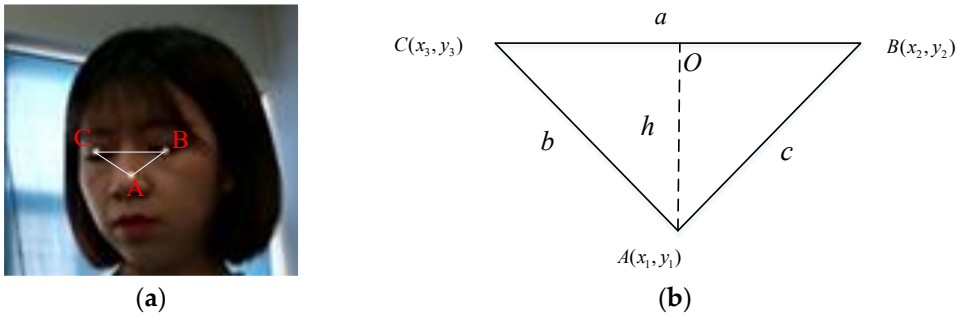


Figure 4. Gray integral projection in different directions: (**a**) vertical direction and (**b**) horizontal direction.

2.2. Geometric Triangular Head Pose Estimation Model

Using Kinect combined with geometric triangles to track the three degrees of freedom parameters of the head direction around the X, Y, and Z axes, the mathematical models are established, respectively. As shown in Figure 5, changes in the rotation angle in the state of head posture deflection such as pitch, yaw, and roll are converted into triangular deformations, which are represented by Pitch, Yaw, and Roll, respectively.



(**a**) (**b**)

Figure 5. Characteristic triangle model. (**a**) Face triangle model; (**b**) feature triangle.

During head position tracking, the geometric triangle is formed by the center of the pupil, and the tip of the nose changes with the doctor's head and eye movements and presents certain geometric characteristics. According to the geometric features of the face, when the head is facing the camera, the Euler angle is defined as 0, and the left and right eyes and nose form an isosceles triangle ΔABC. When the head is deflected, the triangle also undergoes a corresponding position shift as the geometric position of the feature points changes, resulting in deformation. By analyzing the geometric features of ΔABC, you can determine the current head posture.

According to the coordinates of three characteristic points, namely nose tip coordinates $A(x_1, y_1)$, left eye coordinates $B(x_2, y_2)$, and right eye coordinates $C(x_3, y_3)$, the three side length of ΔABC can be obtained as follows:

$$a = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} \tag{3}$$

$$b = \sqrt{(x_3 - x_1)^2 + (y_3 - y_1)^2} \tag{4}$$

$$c = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{5}$$

The three side length a, b, c correspond to $\angle A$, $\angle B$, and $\angle C$, and their angular sizes are expressed as follows:

$$\angle A = \pi - \angle B - \angle C \tag{6}$$

$$\angle B = \arccos(\frac{1}{2c}(\frac{1}{a}(c^2 - b^2) + a)) \tag{7}$$

$$\angle C = \arccos(\frac{1}{2b}(a - \frac{1}{a}(c^2 - b^2))) \tag{8}$$

The height corresponding to the side length a is expressed as follows:

$$h = \frac{2\sqrt{p(p-a)(p-b)(p-c)}}{a} \tag{9}$$

where, $p = \frac{a+b+c}{2}$. Based on the obtained pixel coordinates of the eyes and nose tip, a facial feature triangle is constructed, and a mathematical model of head posture is established and derived. The doctor's head posture parameters are estimated based on the changes in the position of the triangle.

In order to accurately study the technology of doctors' head angle of view and orientation tracking, we establish triangular models for the rotation of the head around different coordinate axes to determine the head and angle of view posture.

(1) Head posture positioning relative to the X-axis (pitch)

The pitch angle is the solution corresponding to the rotation of the doctor's head about the X-axis. When the doctor raises his head or nods, the pitch angle changes to α, and for the frontal image in the triangle model, the side length a of the feature triangle changes, but the corresponding height h does not change. Based on prior knowledge of the distribution of facial organs, the positional relationship between the nose and the eyes can be obtained as follows:

$$h' = 0.8\mathrm{a} \tag{10}$$

where h' is the change in the height of the triangle when the doctor's head changes, and a is the distance between the eyes after turning the head.

$$a = \arccos\frac{2\sqrt{p(p-a)(p-b)(p-c)}}{0.8a^2} \tag{11}$$

When the pitch angle of the doctor's head changes, the angle between the two triangles corresponding to the change will also change. When the doctor raises his head, the corresponding angle α is a negative number; When the doctor lowers his head, the corresponding angle α is positive.

(2) Head posture positioning relative to the Y-axis (yaw)

The solution of the yaw angle corresponds to the rotation of the doctor's head around the Y-axis. When the doctor's head turns to the left or right, the pitch angle changes marked as Ψ, and the shape of the feature triangle changes.

If $y_2 > y_3$, the head is deflected to the left, and the deflection angle Ψ is expressed as follows:

$$\Psi = \arccos(\frac{y_2 - y_1}{c}) - \frac{1}{2}\angle C \tag{12}$$

If $y_2 < y_3$, the head is deflected to the right, and the deflection angle Ψ is expressed as follows:

$$\Psi = \arccos(\frac{y_3 - y_1}{c}) - \frac{1}{2}\angle C \tag{13}$$

When the doctor's head is tilted to the left, the corresponding angle Ψ is positive; When the doctor's head is tilted to the right, the corresponding angle Ψ is negative.

(3) Head posture positioning relative to the Z-axis (roll)

When the doctor turns his head left or right, the change of the roll angle is set to φ as follows:

$$\varphi = \arccos\frac{a}{a'} \tag{14}$$

where a is the distance between the pupils after turning the head, and a' is the distance between the pupils in the forward direction. During head turning analysis, the side length a of the feature triangle changes, but the corresponding height does not change.

$$a' = \frac{h}{0.8} \tag{15}$$

The solution for obtaining the final roll angle is obtained as follows:

$$\varphi = \arccos\frac{0.8a}{h} \tag{16}$$

When the head's roll angle changes, the corresponding change φ between the two triangles will also change. When the doctor turns left, the corresponding angle φ is negative;

When the doctor turns his head to the right, the corresponding angle φ is a positive number. By modeling the feature triangles formed by the left and right pupil and nose tip, a head posture estimation model is established based on the geometric deformation of the head shape and the relative position of the facial feature points. The geometric model composed of multiple feature points overcomes the disadvantage of losing the location of a single feature point and converts the angle of head posture transformation into geometric surface deformation.

2.3. Traceless Kalman Filtering Tracking Combined with Triangles

Using the traditional classic Kalman filtering algorithm, we assume that the head and eye movements are uniform, thereby introducing linear Kaman filtering to complete head position tracking. However, under the interference of different factors, it is difficult to ensure a strictly linear system. Especially in the doctor's head-position-tracking system applied in this paper, the doctor's eye movements are frequent and complex and usually exhibit nonlinear motion. The assumption of linear motion is unreasonable, which results in the head deflection. The deviation of the eyes can lead to the loss of tracking, resulting in significant errors.

In order to improve the accuracy and robustness of head position tracking, we need to introduce nonlinear Kaman filtering for motion optimization. Currently, the commonly used nonlinear Kalman filtering algorithms include Extended Kalman Filter (EKF) and unscented Kalman filter. EKF assumes that the expected value of the result distribution is equal to the expected value of the map and considers only its local environment via linearization. The difference is that UKF [40] has a strong modeling uncertainty capability using unscented transformation (UT), only requires the first few moments of random parameters, and is easy to implement, which can overcome the shortcomings of local linearization of EKF to estimate expected values and distributions more accurately. For linear motion systems, the tracking and prediction effects using EKF and UKF are not significantly different from those using linear KF; However, for nonlinear systems, UKF has better performance than EKF, and as the degree of nonlinearity increases, the gap between the two algorithms will become increasingly large [41]. This paper considers using UKF algorithm to study a head-position-tracking system when doctors' heads undergo significant head deflection when transferring surgical forceps to nurses and adjusting the position of projectors.

Theoretically, UKF uses traceless transformation to process nonlinear system models, uses system state expansion to obtain sampling points and perform nonlinear transformation, thereby obtaining the probability distribution and corresponding statistical characteristics of random variables, and then combines them with the standard Kalman filtering framework for calculation. The first frame image in the video is collected via Kinect to obtain the iterative initial value of the observation state, and then UKF is used to track the trajectory of the doctor's head and eyes. The proposed algorithm completes the modeling process of three-dimensional coordinates and three rotation angles using the UKF algorithm for the centroid of the feature triangle. The specific process of the UKF algorithm is as follows.

(1) Initialization.

$$\overline{x}_0 = E[x_0] \tag{17}$$

$$P_0 = E[(x_0 - \overline{x}_0)(x_0 - \overline{x}_0)^T] \tag{18}$$

The above formula is to substitute a non-linear function with a sigma point to obtain the corresponding non-linear function values, and then calculate the transformed mean $\overline{x}_0$ and covariance P_0 from these point sets.

(2) Select $2n + 1$ sigma points x_k in set form.

$$\varepsilon_{k-1} = [\overline{x}_{k-1|k-1}\overline{x}_{k-1|k-1} \pm (\sqrt{(n+1)P_{k-1|k-1}})] \tag{19}$$

Since the perspective-tracking model has n = 6 states in this paper, the number of sampling points is set to $2 \times n + 1 = 13$.

(3) Time update.

$$\varepsilon_{k|k-1} = f(\varepsilon_{k-1}) \tag{20}$$

$$\hat{x}_{k|_{k-1}} = \sum_{i=0}^{2n} W_{i,e}\varepsilon_{i,k|k-1} \tag{21}$$

$$P_{k|k-1} = \sum_{i=0}^{2n} W_{i,c}(\varepsilon_{i,k|k-1} - \hat{x}_{k|_{k-1}})(\varepsilon_{i,k|k-1} - \hat{x}_{k|_{k-1}})^T \tag{22}$$

$W_{i,e}$ and $W_{i,c}$ represent the mean and covariance of sigma points.

(4) Measurement update.

$$\widehat{Z}_{k|k-1} = h(\varepsilon_{k|k-1}) \tag{23}$$

$$\widehat{Z}_{k|k-1} = \sum_{i=0}^{2n} W_{i,e}Z_{i,k|k-1} \tag{24}$$

(5) Estimate the state and covariance.

$$P_{Z_KZ_K} = \sum_{i=0}^{2n} W_{i,c}(Z_{i,k|k-1} - \widehat{Z}_k)(Z_{i,k|k-1} - \widehat{Z}_k)^T \tag{25}$$

The above formula represents the point set $Z_{k|k-1}$ obtained by substituting sigma points into the nonlinear function in the UT transformation.

$$P_{\varepsilon_KZ_K} = \sum_{i=0}^{2n} W_{i,c}(\varepsilon_{i,k|k-1} - \hat{x}_k)(Z_{k|k-1} - \widehat{Z}_k)^T \tag{26}$$

$$K_k = P_{\varepsilon_{k|k-1}Z_{k|k-1}}P^{-1}_{Z_{k|k-1}Z_{k|k-1}} \tag{27}$$

Formulas (26) and (27) represent the mean and covariance of the weighted nonlinear function $Z_{k|k-1}$ in the UT transform, respectively. $W_{i,e}$ and $W_{i,c}$ are specifically defined as follows:

$$\begin{cases} W_{0,\mathrm{e}} = \frac{l}{n+l} \\ W_{0,c} = \frac{l}{n+l} + (1 - \alpha^2 + \beta) \\ W_{i,e} = W_{i,c} = \frac{l}{2(n+l)}, i = 1, 2, \ldots, 2n \end{cases} \tag{28}$$

where $l = \alpha^2(n + m) - n$. m is the scale factor, and l indicates the scaling parameter. If x is a multidimensional variable, the scale factor is taken as $m = 3 - n$; α is a positive number, and β is a non-negative weight coefficient.

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k[Z_k - \widehat{Z}_{k|k-1}] \tag{29}$$

$$P_k = P_{k|k-1} - K_kP_{Z_KZ_K}K_k^T \tag{30}$$

Repeat the above steps to achieve the calculation of the unscented Kalman filtering algorithm.

The proposed algorithm in this paper aims to obtain the optimal estimation of three-dimensional coordinates and angles in the perspective-tracking environment. Therefore, an unscented Kalman filtering perspective-tracking model combining feature triangles is proposed. An overall model is established for the characteristic triangle centroid composed of three characteristic numbers, namely, the center of the left and right pupil and the tip of the nose. Similar to linear Kalman filtering, UKF also includes three stages: state prediction, prediction of measured values, and state update. The main difference is that for the prediction stage, UT transform is used to approximate the probability distribution of head and eye movement states, making traceless Kalman filtering suitable for nonlinear doctor's head-position-tracking systems.

In this paper, the following nonlinear equations are used to simulate visual angle motion, namely, the following:

$$x = x_0 + vt + \frac{1}{2}at^2 \tag{31}$$

$$x'_{k+1} = v_0 + A_k \sin(w_k t) \tag{32}$$

$$a_{k+1} = x''_{k+1} = A_k w_k \cos(w_k t) \tag{33}$$

where the initial values of x_0 and v_0 are both 0, the acceleration a conforms to a sinusoidal distribution function, and process noise v_k is taken into account. $A_k = 0.08$ m/s and $w_k = \pi rad/s$. In the eye movement state, Newton's motion formula is used, and the results are shown in the following state matrix:

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{34}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{35}$$

where Δt is the interval from k to $k + 1$, and F and H are the state transition matrices.

Compared to the previously assumed perspective-tracking system with uniform motion, this section uses a nonlinear model that changes over time and uses a nine-dimensional vector space $[p_x, p_y, p_z, v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T$ as the state variable of the process equation, where $[p_x, p_y, p_z]^T$ is the three-dimensional coordinate of the facial feature point or feature triangle, $[v_x, v_y, v_z]^T$ is the component of the linear velocity of the facial feature point on the X, Y, and Z coordinate axes, and $[\omega_x, \omega_y, \omega_z]^T$ is the component of the rotational angular velocity on the X, Y, and Z coordinate axes. This article focuses on the study of coordinates and rotation angles. Therefore, the state variable of the process equation of a UKF-based visual tracking nonlinear motion system is $k[p_x, p_y, p_z, \omega_x, \omega_y, \omega_z]^T$, that is, $[X_W, Y_W, Z_W, \alpha, \Psi, \varphi]^T$. The form of the state equation and measurement equation is expressed as follows:

$$x_{k+1} = Fx_k + v_k \tag{36}$$

$$y_{k+1} = Hx_k + n_k \tag{37}$$

where x_k represents the state vector of the system time k, which is the input vector of a known time k, that is, $x_k = [X_W, Y_W, Z_W, \alpha, \Psi, \varphi]^T$.$y_k$ is the measurement vector, v_k is the system process noise, and the measurement noise matrix is represented by n_k, which are all zero mean noise. F represents the transfer matrix, and H represents the observation matrix.

As shown in Equations (36) and (37), Q is the noise covariance matrix of the transfer process, and R is the noise covariance matrix of the measurement process.

$$Q = E\{w_k \, w_k^T\} \tag{38}$$

$$R = E\{v_k \, v_k^T\} \tag{39}$$

The advantages of the proposed algorithm are summarized as follows:

(1) Real-time performance: In the process of perspective tracking, the video stream collected by Kinect camera is 30 fps, and the detection time of the whole system starts from the incoming of each frame image until the output interface displays the perspective pose information. According to the experiment, the average processing time of a single frame is 27.66 ms, which meets the real-time head pose estimation of doctors at a speed of 30 fps.

(2) Robustness: whether under dark light, strong light, or face-blocking environment, the feature triangle can be positioned more accurately, its position and pose information can be obtained, and the corresponding rotation angle changes are consistent with the pre-results.

(3) Accuracy: When forecasting and tracking the same dataset, the root-mean-square error of the improved UKF algorithm combined with feature triangles in this paper is less than that of the KF algorithm of single feature points, which is because when the head is greatly deflection and the face is blocked, the linear KF algorithm will lead to the loss of feature point tracking and generate a large error. The improved UKF algorithm combined with feature triangle overcomes the shortcomings of linear KF, has strong anti-noise ability, and is suitable for nonlinear perspective-tracking models.

Among the evaluation criteria for target-tracking algorithms, a high-performance target tracker mainly includes Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), and Root Mean Square Error (RMSE) [42].

MOTA is used as the evaluation standard for target-tracking accuracy, which is defined as follows:

$$\text{MOTA} = 1 - \frac{FN + FP + IDSW}{GT} \in (-\infty, 1] \tag{40}$$

FN is the total number of missed reports in the whole video; FP is the total number of false positives in the whole video; IDSW is the number of times that are not tracked correctly in frame t, and GT is the sum of the number of the entire video. The closer MOTA is to 1, the better the performance of the filter tracker is. Moreover, MOTA can provide an intuitive measurement of the performance of the head-tracking algorithm independent of the accuracy of target detection.

MOTP is another measure of tracking error, which describes the ability to accurately locate the target. Its calculation formula is as follows:

$$\text{MOTP} = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{41}$$

$d_{t,i}$ represents the distance between the predicted pose and the real pose of the feature points of the detected target i in frame t, and the larger the value, the stronger the accurate positioning ability of the target. c_t indicates the number of successful traces in the current frame. The closer the MOTP is to 1, the higher the accuracy of the nonlinear Kalman filter is.

In the field of target tracking, RMSE is used to evaluate tracking performance as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{\text{i}=1}^{N}(x_i - \hat{x}_i)^2} \tag{42}$$

The closer the RMSE is to 0, the better the prediction effect of the tracking point is, and the better the running effect of the algorithm is.

3. Results

3.1. Dataset Description

In order to verify the applicability of the proposed methods from multiple perspectives, this section conducts experiments and analyses on the public dataset Biwi Kinect Head Pose Data [43] dataset, which uses Kinect to collect nearly 15,000 images, including depth images and RGB images of 20 experimental objects with heads rotating in different directions, as shown in Figure 6. The real dataset collects facial images of eight people (four boys and four girls) using Kinect v2 sensor, each with 100 images of different poses.

Figure 6. Samples of public and real datasets for tracking. (**a**) Samples of public dataset; (**b**) samples of real dataset.

Since the dataset in this paper is a video of different faces, the current state of all frames needs to be listed to show all changes in each dataset, which cannot be shown in this current paper. This paper only chooses some frames as representatives for description.

3.2. Experiments Comparison and Analysis

In order to test the effectiveness and performance of the proposed algorithm in tracking human eyes, this paper tested the tracking performance of the geometric triangle combined with an unscented Kalman filtering algorithm composed of multiple feature points and the single feature point combined with linear Kalman filtering algorithm in video sequences captured by Kinect cameras.

As shown in Figure 7, the visualization effect of tracking arbitrary frame images using single feature points combined with linear Kalman filtering is shown. Real dataset1 and real dataset2 in Figures 7 and 8 represent the true values obtained from different faces selected in the laboratory, and each true value in this paper was obtained by measuring the state of each frame via Kinect. In Figure 7, the blue color represents the actual measurement value obtained by Kinect, and the red color represents the Kalman filter prediction value.

The measured value (represented in blue font) and Kalman prediction value (represented in black font) are output in real time. The serial number represents the number of human faces in the selected dataset, respectively.

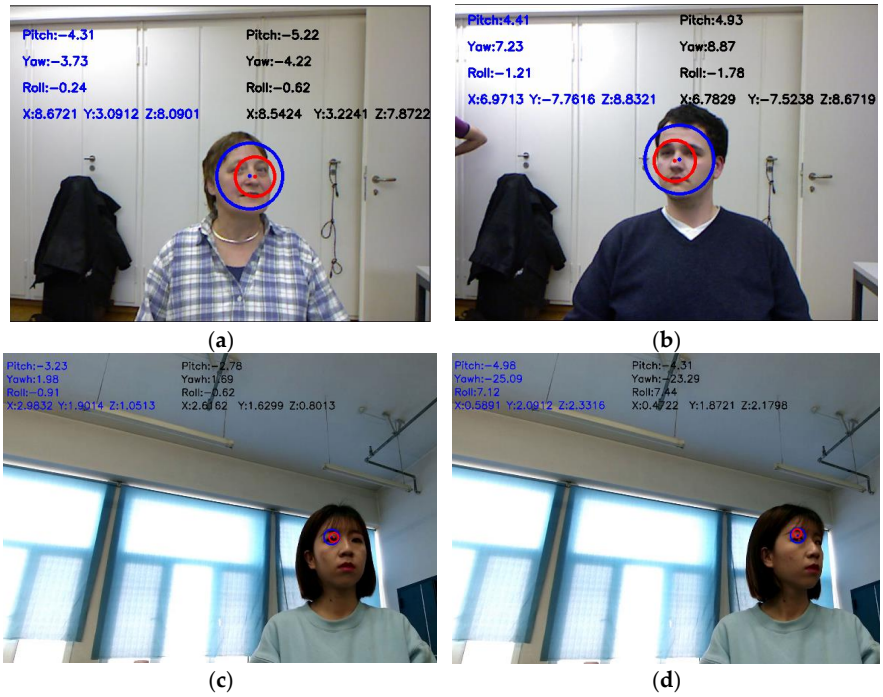


Figure 7. Single feature point tracking using Kalman filter. (**a**) Turn right (public dataset 11), (**b**) forward direction (public dataset 13), (**c**) forward direction (real dataset 1), and (**d**) turn right (real dataset 1).

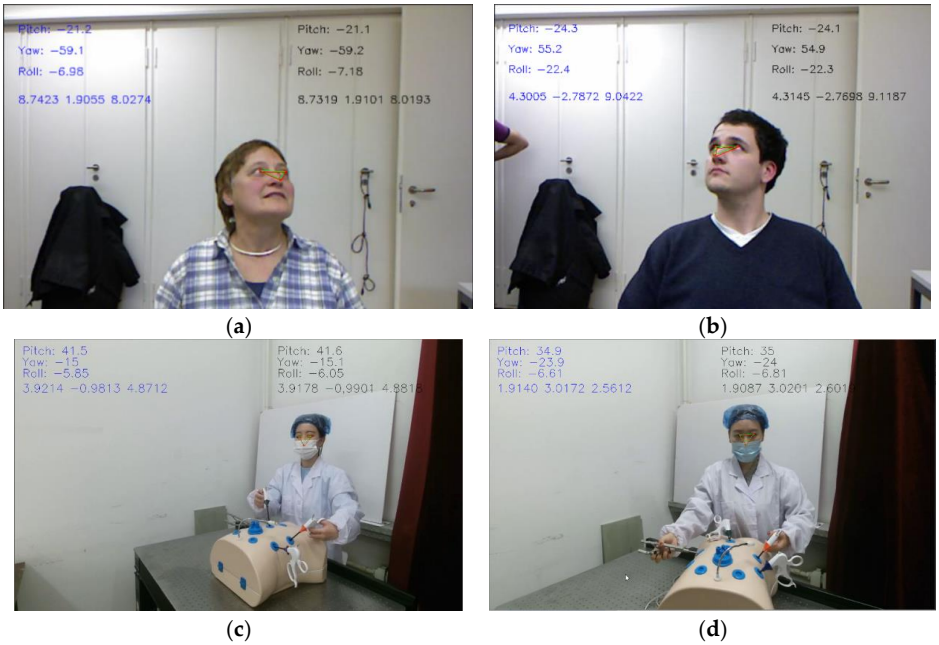


Figure 8. Feature triangle + UKF attitude estimation. (**a**) Right deflection (public dataset 1), (**b**) left deflection (public dataset 13), (**c**) left deflection (real dataset 2), and (**d**) forward direction (real dataset 2).

As seen from Figure 7, using linear Kalman filter single point tracking can better obtain posture for facial and head postures with slight deflection without wearing a mask.

The predicted value and the tracking point of the measured value have a good fitting effect, and the measured value and predicted value of the current frame are output in real-time.

The above experiments assume that the head motion is a linear model, with no significant deflection of head and eye movements, and no significant occlusion of the face. However, this model has the following drawbacks: (1) The method based on a single feature point is susceptible to noise interference and has poor robustness; (2) Linear Kalman filtering cannot meet the nonlinear motion of large head deflection. Therefore, this paper proposes an improved nonlinear Kalman filter combined with geometric triangles.

In order to verify the feasibility and implementation effect of a geometric triangle combined with an unscented Kalman filter in a doctor's head-position-tracking system, this study determines the initial position of the head posture and sets the parameters of the unscented Kalman filter, and the simulation results are shown in Figure 8. The green triangle represents the measured value of the head posture, and the red triangle represents the tracking prediction value of UKF. The blue text on the left represents real-time measurement values, and the black text on the right represents real-time Kalman tracking prediction values. From the numerical results, it can be seen that the UKF algorithm can accurately estimate the posture of the next frame regardless of head deflection or face occlusion.

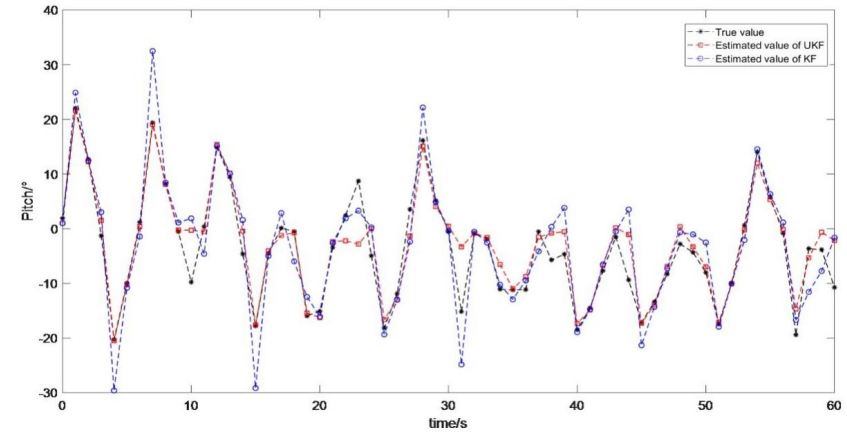
From Figure 8, it can be seen that the improved feature triangle-based unscented Kalman perspective-tracking environment in this article can overcome the limitations of traditional Kalman filtering and is more suitable for head tracking from a doctor's perspective for estimating the position and posture of a face with a large head deflection, as shown in Figure 8a,b. Both the Biwi public dataset and the real dataset collected by the surgical simulation platform can accurately track posture information.

Figure 9 shows the comparison between the rotation angles Pitch, Yaw, and Roll around the X, Y, and Z axes, respectively, and the true values after the traditional KF algorithm and the improved UKF tracking correction combining feature triangles in this article. As seen from the figure, the deviation between the angle information obtained by UKF is represented by the red line, and the actual value is small, while the deviation of the KF algorithm is represented by the blue line is large. The results show that the UKF algorithm combined with feature triangles can effectively reduce noise in non-linear head movements and has better tracking performance than traditional KF algorithms with single feature points in environments with large head deflections or occluding faces. The results in Figure 9 show that the UKF algorithm combined with feature triangles can effectively reduce noise in non-linear head movements and has better tracking performance than traditional KF algorithms with single feature points in environments with large head deflection or occlusion of the face.

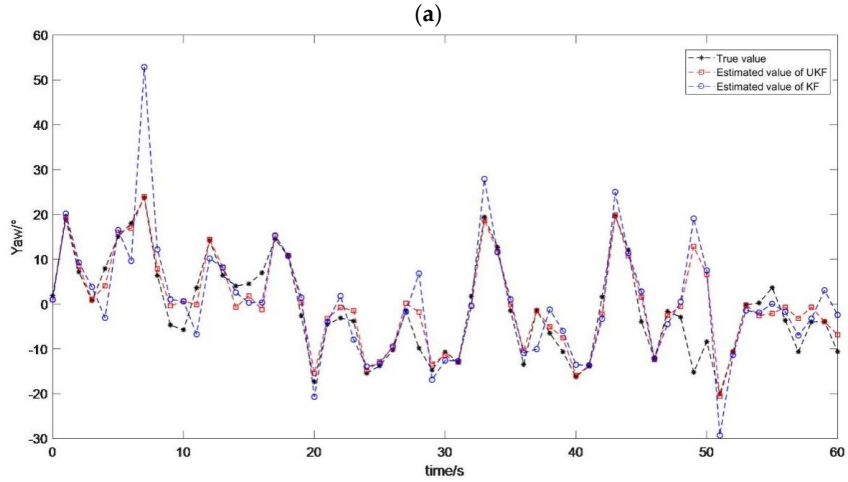
In order to more clearly compare the error between the predicted value and the measured value after using different algorithms, the center of gravity of the linear Kalman filter tracking feature triangle, the center of gravity of the UKF tracking feature triangle, and the actual value obtained by Kinect are extracted, thereby outputting three 3D trajectory data in the world coordinate system. Figure 10 gives three tracking trajectories of Kinect true values and estimated KF and UKF values under conditions of small unobstructed head deflection, large unobstructed head deflection, and occluded head deflection.

As seen from Figure 10, when the head posture movement is slow and approximately linear, both filtering methods can achieve tracking and filtering of measured data. It is shown that the classical KF algorithm and the improved feature triangle combined with the UKF algorithm in this paper have similar tracking effects in linear systems with approximate uniform motion, and both algorithms have good fitting effects with real trajectories. Moreover, linear Kalman filtering is more suitable for linear systems due to its simpler structure and lower algorithm complexity. However, when there is a significant deflection or occlusion of the head posture in nonlinear motion, the predicted value based on the KF algorithm will have a significant error from the actual value, while the UKF algorithm still has a good tracking effect. This is because the UKF algorithm approximates

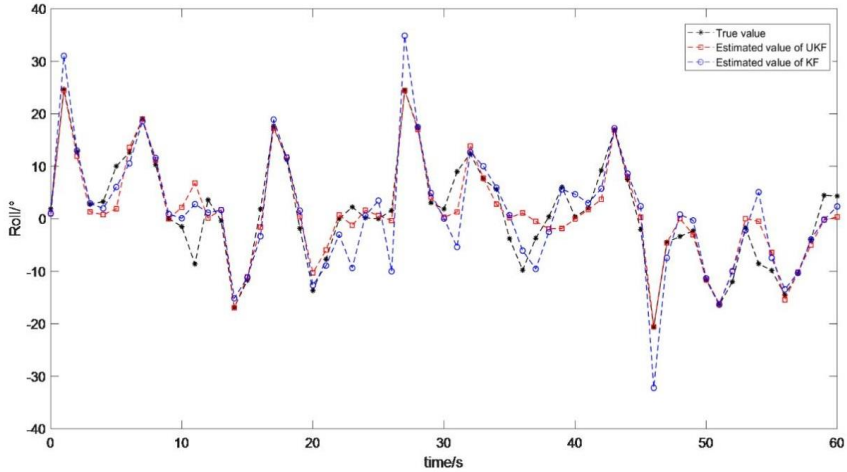
the posterior mean and variance of the nonlinear system via a traceless transformation and still maintains good tracking performance in nonlinear motion systems with head deflection and occlusion.



(**a**)



(**b**)



(**c**)

Figure 9. Comparison chart of rotation angle tracking of different algorithms: (**a**) Pitch, (**b**) Yaw, and (**c**) Roll.

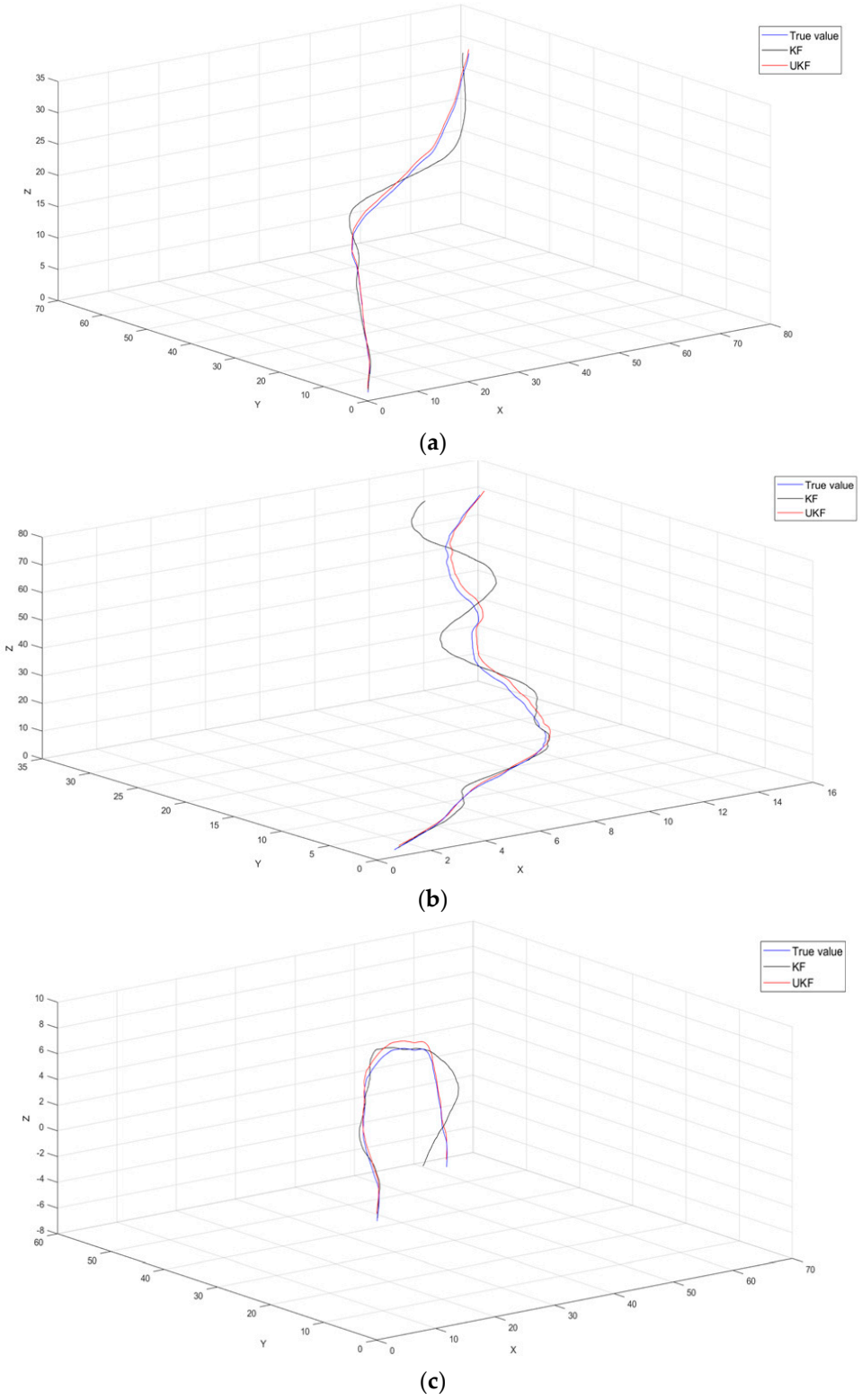


Figure 10. Comparison chart of 3D trajectory prediction in different attitudes: (**a**) Head's small rotation without occlusion, (**b**) Head's large rotation without occlusion, and (**c**) Head's rotation with occlusion.

4. Discussion

Although eye tracking using linear Kalman filtering in the past has made interference judgments and predictions, the tracking results still cannot meet expectations when the

target is severely occluded or occluded for a long time, and there may still be tracking failures. Moreover, in the actual tracking process, there are often situations where multiple interference factors occur simultaneously. Compared to single-factor interference, multiple-factor interference (occlusion, lighting, deflection, etc.) has a more severe impact on target tracking and can even directly lead to tracking failure. In contrast to the nonlinear unscented Kalman filter used in this paper, the probability density distribution function similar to the nonlinear function is used, a large number of determined sample data is used to approach the posterior probability density function of the state variable, and the derivative is not calculated by the Jacobian matrix, which reduces the amount of calculation. Additionally, the UKF algorithm combined with feature triangles can effectively reduce noise and has better tracking performance than traditional KF algorithms with single feature points in environments with large head deflection or occluded faces. The UKF algorithm improved in this article, which combines the feature triangle method compared to the traditional single point linear Kalman filter for target tracking and significantly improves the perspective tracking ability.

In order to test the performance of the algorithm in this paper, the following two sets of experiments were carried out using different algorithms on the same dataset, respectively: comparing the tracking performance of the feature triangle center of gravity combined with the linear KF algorithm and the nonlinear UKF algorithm; The single feature point UKF algorithm for the left pupil, right pupil, and nose tip is averaged based on the results and compared experimentally with the center of gravity of the feature triangle formed by multiple feature points. The comparison results are shown in Tables 1 and 2.

Table 1. Comparison of tracking accuracy between KF and UKF algorithms.

Methods	MOTA (%)	MOTP (%)
KF + The center of gravity of characteristic triangle	76.2	80.9
UKF + The center of gravity of characteristic triangle	81.7	84.9

Table 2. Comparison of UKF combined with single and multiple feature points tracking.

Methods	MOTA (%)	MOTP (%)
Pupil of the left eye	78.9	83.8
Pupil of the right eye	78.4	83.1
Nose	77.9	82.5
Average of three feature points	78.4	83.1
The center of gravity of characteristic triangle	81.7	84.9

By analyzing Table 1, it can be seen that the feature triangle combined with the UKF tracking algorithm used in this article approximates the posterior mean and variance of nonlinear systems via traceless transformation. Compared to the KF algorithm, it improves the MOTA index by 5.5% and MOTP by 4%. Therefore, the UKF algorithm combined with feature triangles performs better in head posture estimation.

According to Table 2, using the UKF algorithm for single-point tracking of the left pupil, right pupil, and nasal tip, and averaging the three tracking results, the MOTA and MOTP for target tracking are 78.4% and 83.1%, respectively. Using the UKF algorithm to track the center of gravity of the feature triangle improves the MOTA to 81.7% and MOTP to 84.9%. Therefore, the improved UKF algorithm combined with the feature triangle method in this paper has a higher tracking effect and accuracy for complex occlusion environments compared to the traditional linear Kalman filter tracking.

Table 3 gives the root mean square error of the KF algorithm combined with triangle tracking and the UKF algorithm combined with triangle tracking. As seen from Table 3, when predicting and tracking the same dataset, the root mean square error of the improved

UKF algorithm combined with feature triangles in this article is smaller than that of the KF algorithm with a single feature point. This is because when the head is deflected significantly and the face is occluded, the linear KF algorithm will cause feature point tracking loss and generate large errors. The improved UKF algorithm combined with feature triangles overcomes the shortcomings of linear KF and has strong noise resistance, which is suitable for nonlinear head-position-tracking models.

Table 3. Error comparison of different tracking algorithms.

Results	KF + Triangle	UKF + Triangle
X/cm	1.53	1.39
Y/cm	1.76	1.44
Z/cm	1.39	1.17
Pitch/°	1.67	1.24
Yaw/°	1.38	1.12
Roll/°	1.57	1.16

The main comparison in this article is between the true values provided in the public dataset and the predicted values obtained by the algorithm in this article, as well as the measured and predicted values obtained from the experimental dataset via camera calibration, model establishment, parameter conversion, and other work. KF and UKF tracking comparisons are conducted on the above data, and the issue of inaccurate feature point localization caused by the head deflection of doctors wearing masks is addressed, which makes traditional linear Kalman filtering algorithms unable to accurately track feature points. The improved feature triangle and UKF tracking algorithm in this paper approximate the posterior mean and variance of the nonlinear system via traceless transformation to update the time and measurement of the doctor's head state.

However, the perspective tracking mentioned in previous literature mostly focuses on the driver's head posture, which is different from the application environment in this article. The improved feature triangle-based unscented Kalman perspective-tracking environment in this article can overcome the limitations of traditional Kalman filtering for large head deflection and pose estimation of masked faces, making it more suitable for head tracking from a doctor's perspective. Accurate tracking of pose information can be achieved on both the Biwi public dataset, and the dataset collected on the surgical simulation platform. Thus, the effectiveness of the algorithm proposed in this paper was verified from two aspects: the actual test dataset and the public test dataset.

5. Conclusions

Aiming at the problem that the head deflection of doctors wearing masks can easily lead to the loss of feature point positioning, and the linear Kalman filtering algorithm cannot accurately track feature points. This paper implements an improved perspective-tracking model that combines feature triangles and unscented Kalman filtering algorithm. From the real-time output data, it can be seen that the maximum angular error between the predicted value and the measured value of the improved unscented Kalman filter is 0.3°, and the maximum coordinate error is 1.74 cm, which meets the experimental requirements. Compared to the linear Kalman filtering algorithm, the tracking accuracy of the geometric triangle-based unscented Kalman filtering algorithm in this paper has been improved by 3.3%, and the accuracy has been improved by 4%. Compared to tracking the average of three feature points, the algorithm in this paper improves the accuracy of target tracking by 5.5% and accuracy by 1.8%, which can meet the needs of doctors for surgery.

Author Contributions: Conceptualization, X.Y., Y.Z., H.W. and A.W.; methodology, X.Y. and Y.Z.; software, Y.Z.; validation X.Y. and Y.Z.; writing—review and editing X.Y., Y.Z., H.W. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the high-end foreign experts introduction program (G2022012010L) and Major Science and Technology Projects of Zhongshan City in 2022 (2022A1020).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: https://data.vision.ee.ethz.ch/cvl/gfanelli/head_pose/head_forest.html#db (accessed on 24 March 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Klaib, A.F.; Alsrehin, N.O.; Melhem, W.Y.; Bashtawi, H.O.; Magableh, A.A. Eye tracking algorithms, techniques, tools, and applications with an emphasis on machine learning and Internet of Things technologies. *Expert Syst. Appl.* **2021**, *166*, 114037. [CrossRef]
2. Liu, M.; Li, H.; Dai, H. Appearance-based Gaze Estimation Using Multi-task Neural Network. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *806*, 012054. [CrossRef]
3. Marquard, J.L.; Henneman, P.L.; He, Z.; Jo, J.; Fisher, D.L.; Henneman, E.A. Nurses' behaviors and visual scanning patterns may reduce patient identification errors. *J. Exp. Psychol. Appl.* **2011**, *17*, 247. [CrossRef] [PubMed]
4. MacNeil, R.R.; Gunawardane, P.D.S.H.; Dunkle, J.; Zhao, L.; Chiao, M.; de Silva, C.W.; Enns, J.T. Using electrooculography to track closed-eye movements. *J. Vis.* **2021**, *21*, 1898. [CrossRef]
5. Katona, J. Clean and dirty code comprehension by eye-tracking based evaluation using GP3 eye tracker. *Acta Polytech. Hung.* **2021**, *18*, 79–99. [CrossRef]
6. Fabio, A.; Giannatiempo, S.; Semino, M. Longitudinal cognitive rehabilitation applied with eye-tracker for patients with Rett Syndrome. *Res. Dev. Disabil.* **2021**, *111*, 103891. [CrossRef]
7. Chugh, S.; Brousseau, B.; Rose, J.; Eizenman, M. Detection and Correspondence Matching of Corneal Reflections for Eye Tracking Using Deep Learning. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021.
8. Pauszek Joseph, R. An introduction to eye tracking in human factors healthcare research and medical device testing. *Hum. Factors Healthc.* **2023**, *3*, 100031. [CrossRef]
9. Yu, B.; Zhang, X.; Zhang, Y.; Wang, L.; Chen, X.; Lin, X. The Application of Sensors in Medical Fields. *China Med. Device Inf.* **2017**, *23*, 17–18.
10. Li, Z. Indoor human body localization method based on IMU and infrared sensor network. *Transducer Microsyst. Technol.* **2018**, *37*, 24–27.
11. Atallah, L.; Lo, B.; King, R.; Yang, G.Z. Sensor positioning for activity recognition using wearable accelerometers. *IEEE Trans. Biomed. Circuits Syst.* **2011**, *5*, 23–33. [CrossRef]
12. Gabela, J.; Kealy, A.; Li, S.; Hedley, M.; Moran, W.; Ni, W.; Williams, S. The Effect of Linear Approximation and Gaussian Noise Assumption in Multi-Sensor Positioning Through Experimental Evaluation. *IEEE Sens. J.* **2019**, *19*, 10719–10727. [CrossRef]
13. Zhang, Z. Microsoft Kinect Sensor and Its Effect. *IEEE MultiMedia* **2012**, *19*, 4–10. [CrossRef]
14. Smisek, J.; Jancosek, M.; Pajdla, T. 3D with Kinect. In Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops) Computer Vision Workshops (ICCV Workshops), Barcelona, Spain, 6–13 November 2011; pp. 1154–1160.
15. Khoshelh, K. Accuracy Analysis of Kinect Depth Data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2012**, *XXXVIII–5/W12*, 133–138. [CrossRef]
16. Zhang, R.H.; Walshe, C.; Liu, Z.; Guan, L.; Muller, K.; Whritner, J.; Zhang, L.; Hayhoe, M.; Ballard, D. Atari-head: Atari human eye-tracking and demonstration dataset. In Proceedings of the 34th AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 6811–6820.
17. Li, Z.; Zheng, X.; Liu, K. Research Progress in Mobile Device Sight Tracking Technology. *Comput. Eng. Appl.* **2018**, *54*, 6–11.
18. Zhang, K.; Zhao, X.; Ma, Z.; Man, Y. A simplified 3d gaze tracking technology with stereo vision. In Proceedings of the 2010 International Conference on Optoelectronics and Image Processing, Hainan China, 11–12 November 2011; pp. 131–134.
19. Li, S.; Liu, Z.; Sun, M.T. Real time gaze estimation with a consumer depth camera. *Inf. Sci.* **2015**, *320*, 346–360.
20. Zanetti, M.; De Cecco, M.; Fornaser, A.; Leuci, M.; Conci, N. The Use of INTER-EYE for 3D Eye-Tracking Systematic Error Compensation. In Proceedings of the International Symposium ELMAR-2016, Zadar, Croatia, 12–14 September 2016; pp. 173–176.
21. Elmadjian, C.; Shukla, P.; Tula, A.D.; Morimoto, C.H. 3D gaze estimation in the scene volume with a head-mounted eye tracker. In Proceedings of the Workshop on Communication by Gaze Interaction, Warsaw, Poland, 15 June 2018; pp. 1–9.
22. González-Ortega, D.; Díaz-Pernas, F.J.; Martínez-Zarzuela, M.; Antón-Rodríguez, M. Comparative analysis of Kinect-based and oculus-based gaze region estimation methods in a driving simulator. *Sensors* **2020**, *21*, 26. [CrossRef]
23. Han, E. Integrating mobile eye-tracking and VSLAM for recording spatial gaze in works of art and architecture. *Technol. Archit. Des.* **2021**, *5*, 177–187. [CrossRef]

24. Liu, J.; Chi, J.; Hu, W.; Wang, Z. 3D Model-Based Gaze Tracking via Iris Features with a Single Camera and a Single Light Source. *IEEE Trans. Hum.-Mach. Syst.* **2021**, *51*, 75–86. [CrossRef]
25. Chen, Q. Research of Attention Analysis System Based on Gaze Estimation Neural Networks. Ph.D. Thesis, University of Electronic Science and Technology of China, Chengdu, China, 2022; pp. 29–35.
26. Chi, J.N.; Xie, L.H.; Zhang, P.Y.; Lu, Y.F.; Zhang, G.S. Hybrid Particle and Kalman Filtering for Pupil Tracking in Active IR Illumination Gaze Tracking System. *Math. Probl. Eng.* **2014**, *2014*, 426234. [CrossRef]
27. Janabi-Sharifi, F.; Marey, M. A kalman-filter-based method for pose estimation in visual servoing. *IEEE Trans. Robot.* **2010**, *26*, 939–947. [CrossRef]
28. Sultan, M.S.; Chen, X.; Ma, G.; Xue, J.; Ni, W.; Zhang, T.; Zhang, W. Hand-eye 3D pose estimation for a drawing robot. In Proceedings of the 2013 IEEE International Conference on Mechatronics & Automation, Takamatsu, Japan, 4–7 August 2013.
29. Munir, F.; Jalil, A.; Jeon, M. Real time eye tracking using Kalman extended spatiotemporal context learning. In Proceedings of the Second International Workshop on Pattern Recognition, SPIE, Singapore, 1–3 May 2017; Volume 10443, pp. 254–258.
30. Pan, Z.; Liu, R.; Zhang, M. *Human Eye Tracking Based on CNN and Kalman Filtering*; Transactions on Edutainment XV; Springer: Berlin/Heidelberg, Germany, 2019; pp. 265–273.
31. Vaishnavi, B.; Rao, S.K.; Jahan, K. Underwater bearings-only tracking using particle filter. *Int. J. Innov. Technol. Explor. Eng.* **2019**, *8*, 451–455.
32. Bagherzadeh, S.; Toosizadeh, S. Eye tracking algorithm based on multi model Kalman filter. *HighTech Innov. J.* **2022**, *3*, 15–27. [CrossRef]
33. Feng, Z.; Yang, B.; Li, Y.; Wang, Z.; Zheng, Y. Research on Human Hand Tracking Aiming at Improving Its Accurateness. *J. Comput. Res. Dev.* **2008**, *45*, 1239–1248.
34. Zhang, Z.; Zhang, J. A new real-time eye tracking based on nonlinear unscented Kalman filter for monitoring driver fatigue. *J. Control. Theory Appl.* **2010**, *8*, 181–188. [CrossRef]
35. Hannuksela, J. *Facial Feature Based Head Tracking and Pose Estimation*; Department of Electrical & Information Engineering, University of Oulu: Oulu, Finland, 2003.
36. Bankar, R.; Salankar, S. Improvement of Head Gesture Recognition Using Camshift Based Face Tracking with UKF. In Proceedings of the 2019 9th International Conference on Emerging Trends in Engineering and Technology—Signal and Information Processing (ICETET-SIP-19), Nagpur, India, 1–2 November 2019.
37. Du, X.; Chen, D.; Liu, H.; Ma, Z.; Yang, Q. Real-time hand tracking based on YOLOv4 model and Kalman filter. *J. China Univ. Posts Telecommun.* **2021**, *28*, 86.
38. Tian, L.; Xu, Y.; Xue, W.; Cheng, L. Consistent Extended Kalman Filter Design for Maneuvering Target Tracking and Its Application on Hand Position Tracking. *Guid. Navig. Control.* **2022**, *2*, 26. [CrossRef]
39. Li, P.; Liang, L.; Hui, D.; Wang, G. Head Pose Estimation of Patients with Monocular Vision for Surgery Robot Based on Deep Learning. *Chin. J. Biomed. Eng.* **2022**, *41*, 537–546.
40. Li, H.; Chutatape, O. Boundary detection of optic disk by a modified ASM method. *Pattern Recognit.* **2003**, *36*, 2093–2104. [CrossRef]
41. Putriany, D.; Rachmawati, E.; Sthevanie, F. Indonesian ethnicity recognition based on face image using gray level co-occurrence matrix and color histogram. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1077*, 012040. [CrossRef]
42. Singh, U.K.; Singh, A.K.; Bhatia, V.; Mishra, A.K. EKF-and UKF-based estimators for radar system. *Front. Signal Process.* **2021**, *1*, 704382. [CrossRef]
43. Papakon, G.; Amir, M.; Warn, G. A scaled spherical simplex filter (S3F) with a decreased n + 2 sigma points set size and equivalent 2n + 1 Unscented Kalman Filter (UKF) accuracy. *Mech. Syst. Signal Process.* **2022**, *163*, 107433.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

X-ray Security Inspection Image Dangerous Goods Detection Algorithm Based on Improved YOLOv4

Xiaoyu Yu [1,2], Wenjun Yuan [2] and Aili Wang [2,*]

[1] College of Electron and Information, University of Electronic Science and Technology of China, Zhongshan Institute, Zhongshan 528402, China
[2] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; 2020610085@stu.hrbust.edu.cn
* Correspondence: aili925@hrbust.edu.cn

Abstract: Aiming at the problems of multi-scale and serious overlap of dangerous goods in X-ray security-inspection-image samples, an X-ray dangerous-goods-detection algorithm with high detection accuracy is designed based on the improvement of YOLOv4. Using deformable convolution to redesign YOLOv4's path-aggregation-network (PANet) module, deformable convolution can flexibly change its receptive field based on the shape of the detected object. When the high-level information and low-level information are fused in the PANet module, deformable convolution is used to align features, which can effectively improve the detection accuracy. Then, the Focal-EIOU loss function is introduced, which can solve the problem of the CIOU loss function being prone to causing severe loss-value oscillation when dealing with low-quality samples. During training, the network can converge more quickly and the detection accuracy can be slightly improved. Finally, Soft-NMS was used to improve the non-maximum suppression of YOLOv4, effectively solving the problem of the high overlap rate of hazardous materials in the X-ray security-inspection dataset and improving accuracy. On the SIXRay dataset, this model detected 95.73%, 83.00%, 82.95%, 85.13%, and 80.74% AP for guns, knives, wrenches, pliers, and scissors, respectively, and the detected mAP reached 85.51%. The proposed model can effectively reduce the false-detection rate of dangerous goods in X-ray security images and improve the detection ability of small targets.

Keywords: X-ray security image; YOLOv4; deformable convolution; path aggregation network; Soft-NMS

1. Introduction

X-rays can penetrate substances and interact with them to produce high-resolution images with rich internal details, which is conducive to the detection of high-density contraband hidden inside objects [1]. Different materials have different degrees of X-ray absorption and scattering attenuation, and the corresponding X-ray images generated by the goods have different colors. Because of the advantages of using X-ray to detect dangerous goods in baggage, such as the little damage to goods, non-necessity of unpacking, its safety and reliability, and its easy operation, it is widely used in various places requiring security inspection.

At present, the X-ray safety inspection for dangerous goods is still manual monitoring. The security inspector needs to observe the X-ray-scanned image on the screen with the naked eye and judge whether there are dangerous goods based on their own experience. The accuracy of the inspection of prohibited goods depends on the proficiency and mental state of the security inspector. In addition, there are many uncertainties in the number of items in the X-ray security-inspection images, and not all luggage contains dangerous items, which greatly affects the alertness of security inspectors while increasing the difficulty of detection, resulting in an increase in the rate of missed detection.

Compared with natural images, X-ray images are characterized by low contrast, limited color range, poor texture, and serious overlap [2]. Therefore, experts have conducted deep research on the characteristics of designing, labeling, and selecting X-ray-image data. Bastan et al. compared the performance of various feature detectors and a combination of different descriptors, proving the feasibility and potential of traditional methods of manual features in X-ray-image detection [3]. Mikolaj et al. found that the methods of a visual word bag yielded great differences in terms of the feature detector, feature descriptor, word size, and final classification method, and further proved that the use of feature-point density as a simple measure of image complexity is a component of the overall classifier [4].

In 2017, Mery et al. proposed separating the target and background of an X-ray image using the adaptive sparse-representation method to improve the performance of the detection of dangerous goods [5]. In 2018, Xing Xiaolan et al. used the method of median filtering to de-noise X-ray security images and used two gray-projection algorithms to find the area with the smallest gray value in the image to carry out the detection of the X-ray security image [6]. Russo et al. used the approximate-median-filter algorithm to remove the background from the input image and then used the shape-based-filtering method to obtain the region of interest, calculated the local binary pattern (LBP) and histogram based on the pixels of the region of interest to form the feature vector, and finally used a support vector machine (SVM) to classify [7]. In 2019, Santos et al. implemented bilateral filtering-preprocessing technology before the detection phase to improve the accuracy and used color-threshold processing and Hough transform in the HSV color space to effectively segment the region of interest. In the detection phase, the directional-gradient histogram was extracted from the image as the key feature of classification [8]. Li Hai et al. used X-ray images for bilateral filtering processing to select an image with rich edge information, and each channel was subject to homomorphic filtering processing, which could effectively assist with the detection of dangerous goods in X-ray images [9].

Although the method of detecting dangerous goods based on traditional X-ray security images has strong interpretability, the efficiency of manual feature extraction is low, and the performance of traditional methods in processing large amounts of data and rapid detection is not ideal.

Due to the significant improvement in the parallel-computing ability of computer systems and the emergence of a large number of X-ray security-image data, the X-ray security-image technology for detecting dangerous goods based on the deep-learning method has gradually become the preferred method of most researchers, as it can realize automatic extraction of multiple features of the image [10], thus avoiding the traditional image-feature-extraction operation, and has good invariance features, such as displacement and scaling, as well as good scalability, which are great advantages in X-ray security-image detection [11].

In 2017, Akcay et al. used transfer learning to conduct pre-training on an ImageNet dataset with Faster RCNN, and the mAP reached 88.3% [12]. Zhu et al. improved the Faster RCNN detection model through appropriate anchor selection and the non-maximum-suppression (NMS) algorithm, and achieved excellent detection performance [13]. In 2018, Singh et al. proposed the R-FCN300 model to improve the detection speed by solving the problem of repeated calculation of ROI in Fast RCNN and decoupling the classification branch [14]. Zhang et al. improved on the basis of the SDD network, proposed the Detection with Enriched Semantics (DES) model, used the segmentation module to increase the semantic information of low-level feature maps, and used the global-activation module to enhance the semantic information of high-level feature maps [15].

In 2019, Guo et al. used ResNet101 to replace the backbone network of the basic network SSD to obtain stronger anti-degradation performance to build the SSD-Resnet101 structure. On this basis, the shallow features are used to fuse with the deep features to increase the receptive field of the shallow-feature map. Full use is made of context information to improve the detection accuracy of small and medium-sized target dangerous goods in safety inspections [16]. In order to explore the transferability brought by different

shapes, different image resolutions, and different colors in X-ray images, Gaus et al. used the transfer-learning method to evaluate the network structures of Faster R-CNN, Mask R-CNN, and RetinaNet, among which Faster R-CNN had the best detection performance [17].

In 2020, Tang Haoyang et al. used deformable convolution to reconstruct the feature-pyramid structure of the SSD network to improve the detection accuracy of the network in order to extract the deeper semantic features of dangerous goods when the feature pyramid is fused [18]. Yu et al. proposed an SSD-X detection network. Considering the position uncertainty and overlap of the target, multiple data enhancement is used to effectively improve the accuracy and over-fitting phenomenon. The focus loss is introduced into the confidence-loss function to accelerate the convergence rate of the model [19].

In 2021, Guo Shouxiang et al. believed that the composite-backbone-network structure has a stronger ability to extract features. On the basis of YOLOv3, the composite-backbone network was used to build the YOLO-C network to improve the detection accuracy of the network [20]. On the basis of CenterNet, Tang et al. used ResNet-50 to improve its backbone network to improve detection speed and added a sampling layer to the feature-processing network to improve detection accuracy [21].

In 2022, Kumar et al. proposed a multi-channel region-recommendation network (MCRPN) to solve the scale difference of dangerous goods in X-ray-image recognition and achieve a faster RCNN network, which uses different levels of convolution features in visual semantics and integrates the richer semantic information at the upper level of VGG16 and the shallower edge features at the lower level to map the multi-scale candidate-target area to the corresponding feature map [22]. Jiang et al. proposed the AM-YOLO model, adding the SE-attention module to the backbone network of YOLOv4 to distinguish the importance of feature-map channels, and proposed a new path-aggregation network to achieve the fusion of shallow and deep features, thus improving the network model-detection ability [23].

We proposed a combination of deformable convolution and the path-aggregation-network (PANet) module of the YOLOv4 network, and designed and implemented a dangerous-goods-detection algorithm for X-ray security-inspection images. The main contribution of the proposed method is summarized as follows:

1. This paper proposes combining deformable convolution with the PANet module of the YOLOv4 network and using the more flexible receptive field of deformable convolution to solve the problem of feature misalignment in the feature-fusion module of YOLOv4 in the process of high- and low-level feature fusion.

2. Based on the backbone network, a channel-pruning algorithm is designed to remove redundant channels in the network, reduce the amount of computation, and improve the reasoning speed of the network model. Experiments show that this method can effectively improve the inference speed of network models and meet the requirements of real-time security checks in terms of speed.

2. Related Work

X-ray security-inspection images mainly have the following characteristics:

(1) Serious loss of detail and color features: Due to the X-ray-imaging method and the material of the object itself, the original color information and detailed information on the contour of the object are lost during imaging.

(2) Background interference: When the background material is the same as the object, contour information similar to the object color is generated, which interferes with the model's learning of object-feature information during training.

(3) Serious-overlap phenomenon: The shape of an object undergoes significant changes under ray projection, and the random placement of positions and the overlapping placement of multiple objects results in complex, overlapping, and occluding contours of the object formed by X-rays passing through the object, increasing the difficulty of extracting effective features of various categories.

(4) The scale of prohibited items is diverse: Even within a single X-ray security image, the size of prohibited items is diverse, and the same object may even exhibit different sizes due to issues such as angle, compression, and image size.

Based on the characteristics of X-ray images, Mery et al. [11] evaluated 10 X-ray contraband-detection methods based on visual-pool-bag models, sparse representations, deep learning, and classical pattern-recognition schemes, and found that deep-learning methods performed the best. Miao et al. [24] proposed a class-balance hierarchical-refinement model that uses different scale features to filter irrelevant information and identify and classify prohibited items. Wei et al. [25] proposed a deblocking attention module that utilizes edge information and material information of prohibited items to generate attention maps and feature maps for detection. Gu et al. [26] proposed using feature-enhancement modules to improve feature-extraction capabilities while utilizing multi-scale fusion to obtain more accurate regions of interest, improving the accuracy and robustness of prohibited-item detection in X-ray security images.

The above methods have greatly improved the detection accuracy and laid the foundation for the development of X-ray-security prohibited-item detection. However, in real scenes, the variable shape and scale of targets, severe overlapping occlusion, and complex background interference are still key issues that need to be addressed in current research, especially the challenges brought on by the characteristics of X-ray-security images of prohibited items themselves. The current detection accuracy still does not meet the requirements of practical applications. On the one hand, the scale and shape of contraband vary greatly, the distribution of context information is uneven, and conventional convolution cannot adapt to the receptive field of the actual target and cannot flexibly handle the height change of context distribution due to its fixed sampling location, which ultimately causes some important context information to be ignored, weakening the ability of feature extraction. On the other hand, items of the same material present the same or similar colors in the image, which can easily cause confusion between the target and the background when overlapping and obstructing, and can even not be distinguished. When these items are processed through convolutional layers, they receive similar feature responses, resulting in a decrease in recognition and positioning accuracy.

3. Improved YOLOv4 Model

The YOLO network is a target-detection network based on regression. It can complete classification and location tasks of targets in a network directly and simultaneously by fully extracting the features of the detected targets, and truly realizes a simple and efficient end-to-end design idea. The YOLOv4 algorithm has a more complex network structure but has higher accuracy and faster speed, and shows a significant improvement in detecting small targets and occluded targets.

There will be many targets of different sizes in the X-ray-detection task, and different targets have different characteristics, resulting in a low accuracy rate of judgment only through target characterization when detecting dangerous goods. YOLOv4's neck adopts the PAN structure for fusion, uses shallow features to distinguish simple targets, uses deep features to distinguish complex targets, transfers high-level strong semantic features, enhances the whole pyramid, enhances semantic information, adds a bottom-up pyramid, transfers low-level location features, combines semantic information, and has location information. However, PAN ignores the problem of feature alignment. The direct splicing between up-sampling and local features cause the feature map to have an unaligned context, which turns into errors in prediction, especially on the object boundary.

This improved design proposes a feature-selection module that can adaptively learn the bottom-up feature map containing more spatial details to achieve accurate positioning. A feature-alignment module is proposed that aligns the up-sampling feature with a set of reference features by adjusting each sampling position in the convolution kernel using the learned offset. The two modules are integrated into the PAN structure, and the PANv2 structure is proposed as shown in Figure 1.

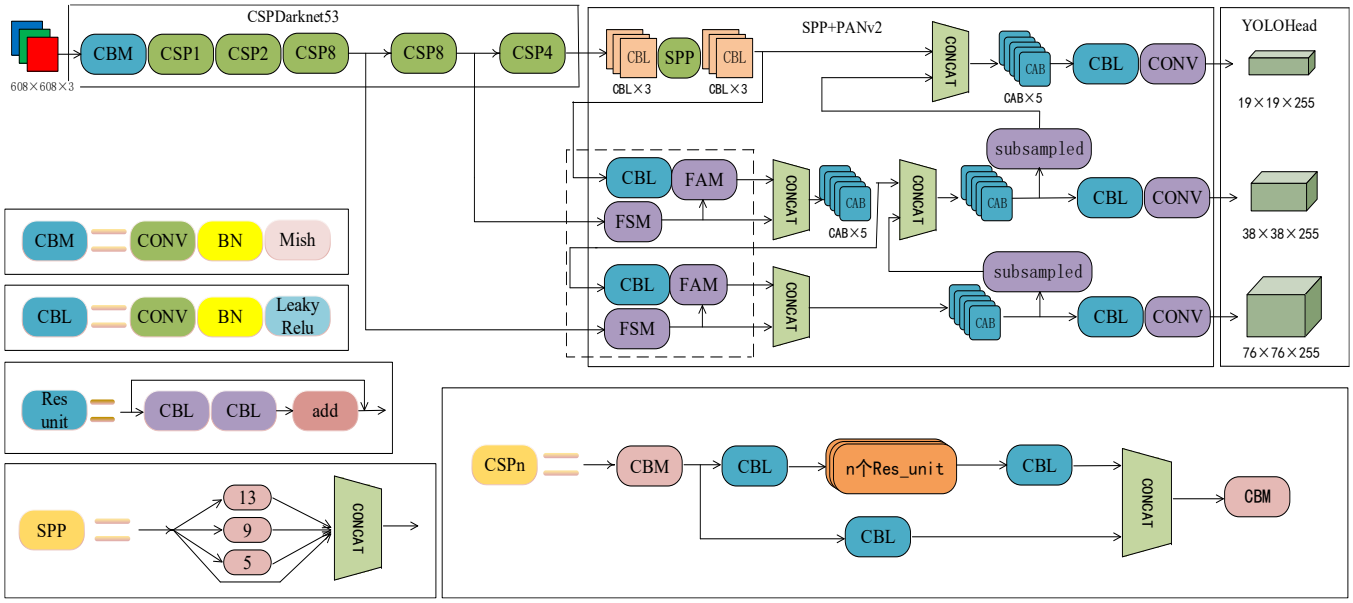


Figure 1. Improved YOLOv4 network for dangerous-goods detection.

3.1. The Deformable Convolution

The neck of YOLOv4 uses the PANet module to fuse the high- and low-level features and combines the semantic information and location information. However, the direct splicing between the upper sampling and local features may cause the feature map to contain misaligned context information, which can be converted into errors in prediction, especially on the object boundary, resulting in poor accuracy when detecting dangerous goods. Deformable convolution [27] introduces offset in the receptive field, which is different from the square receptive field of traditional convolution. This feature can be used to achieve more accurate feature alignment and improve the accuracy of detection.

The standard convolution layer usually uses sliding-window and scale-invariant feature transformation to deal with the geometric changes of the object. It can effectively expand the receptive field by stacking more convolution layers, but the corresponding calculation cost increases exponentially with the increase of the receptive field. With the increase in depth, the receptive field goes far beyond the area of interest, resulting in the extracted features being affected by redundant context information. The deformable convolution can adaptively determine the size of the receptive field and flexibly adjust the receptive field by learning the additional offset of the convolution network.

Traditional convolution uses a regular grid as the convolution kernel, slides on the input-characteristic graph step by step, and calculates the point product sum of the matrix of the input characteristic graph one by one, and finally adds the variation. The size of the convolution kernel determines the size of the receptive field. The corresponding value of each position in the output-characteristic diagram is calculated as shown in Formula (1):

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n) \tag{1}$$

where y represents the output-feature map, x represents the input-feature map, w represents the weight coefficient, b represents the deviation, p_0 is the 0th point in the grid, and p_n is the nth point in the grid.

The deformable convolution adds a learnable offset parameter on the basis of ordinary convolution, which can be used to adjust the receptive field to better extract the features of complex-shaped objects. In the deformable convolution, $\{\Delta p_n | n = 1, \cdots, N\}$ is considered

an offset. The offsets of each position are enumerated on the convolutional kernel when $N = |R|$, as shown in Formula (2):

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n + \Delta p_n) \tag{2}$$

The diagram of the deformable convolutional receptive field is shown in Figure 2. The green dot represents the original receptive-field range, and the blue dot represents the receptive-field range after increasing the offset. The offset in deformable convolution is learned through back propagation, which makes the receptive field more flexible and able to adaptively match various shapes.

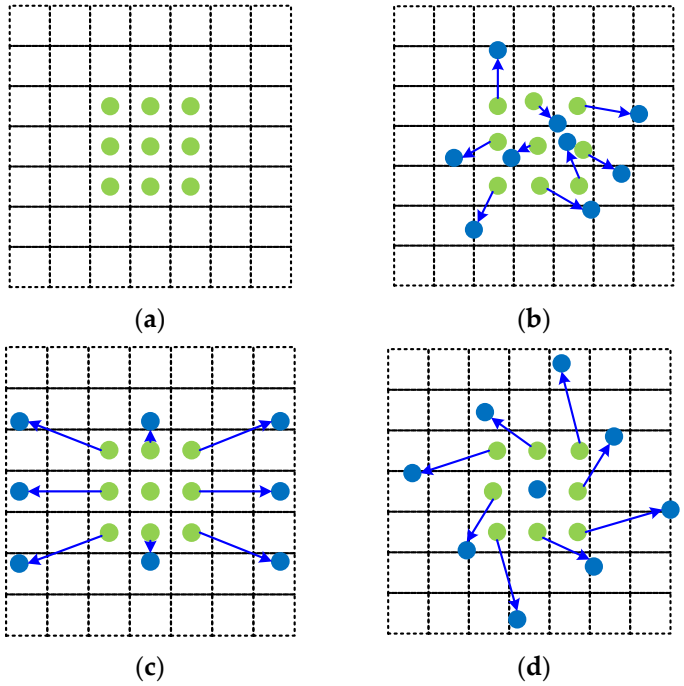


Figure 2. Schematic diagram of the deformable convolution's receptive field. (**a**) Original receptive field, (**b**) target-movement receptive field, (**c**) target-size-scaling receptive field, and (**d**) target-rotation receptive field.

Since the coordinate position after adding the offset is usually not an integer, it does not correspond to the feature points on the actual feature map. Bilinear interpolation can be used to obtain the feature value after adding the offset.

$$x(p) = \sum_{q} G(q, p) \cdot x(q) \tag{3}$$

p corresponds to any sampling point at any position, $x(q)$ is the convolutional-feature map, and $G(\cdot)$ is a bilinear insertion kernel with one linear insertion in the axis and the axis directions.

$$G(q, p) = \sum_{q} g(q_x, p_x) \cdot g(q_y, p_y) \tag{4}$$

$$g(q_x, p_x) = \sum_{q} \max(0, 1 - |q_x - p_x|) \tag{5}$$

q_x and q_y are the horizontal and vertical coordinates, respectively, of the feature-map coordinate point q, and p_x are p_y the horizontal and vertical coordinates, respectively, of the feature-map coordinate point p.

The schematic diagram of deformable convolutional sampling is shown in Figure 3. For the input-feature map, it is assumed that the traditional convolutional network uses a convolutional-kernel size of 3 × 3 to obtain the convolutional-feature map, and the deformable convolution is obtained by an additional kernel of 3 × 3 to learn the offset domain. The offset domain has the same size as the input-feature map, and the number of channels is 2*N*, which corresponds to N two-dimensional offset. Then, the output is the result of the joint action of the input-feature map and the offset domain.

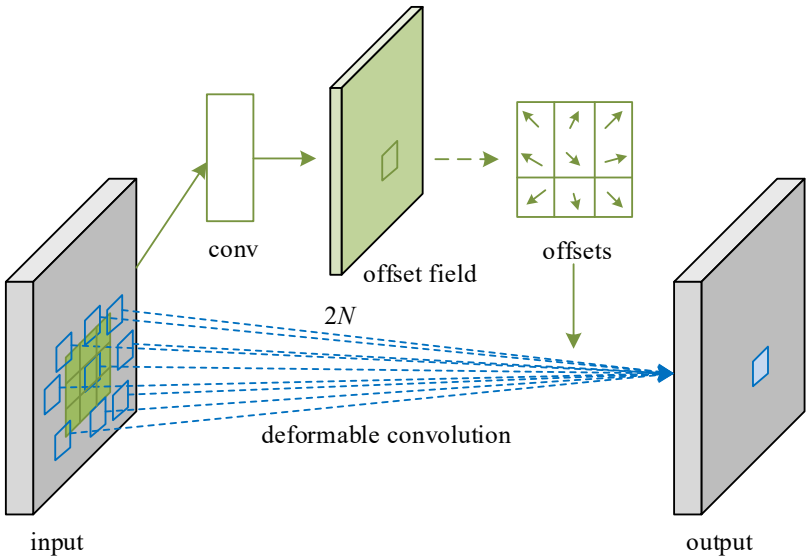


Figure 3. Schematic diagram of deformable convolution sampling.

3.2. Improved PANet Module

Before performing channel reduction on features, a feature-selection module is proposed to adaptively adjust feature maps that contain excessive spatial details, thereby achieving precise localization and suppressing redundant feature maps. The schematic diagram of the feature-selection module (FSM) is shown in Figure 4. The module first adopts a dual-branch structure, using global-maximum pooling and global-average pooling to extract the information of its feature maps. Then, the output-feature maps are inputted into the MLP structure separately. Then, the two feature maps outputted from the MLP are added and subjected to the sigmoid operation. Finally, the output obtained by multiplying the output-feature maps by the input-feature maps is placed in the input-feature maps and then subjected to the add operation.

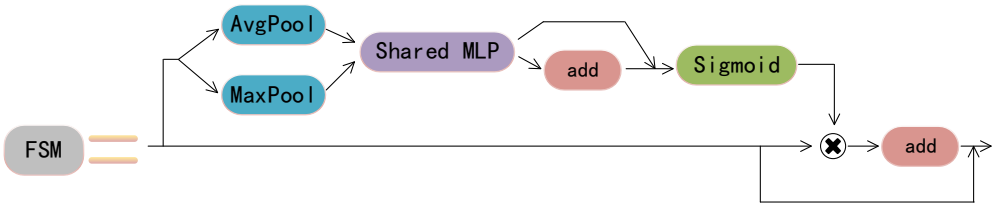


Figure 4. Feature-selection module (FSM).

The feature-selection module introduces additional skip connections between input- and scaled-feature maps. Using skip connections for scaled features can avoid any specific channel response being excessively amplified or suppressed and can adaptively recalibrate the channel response through channel attention.

The recursive use of down-sampling and up-sampling operations in deep neural networks may result in spatial misalignment between feature maps, and feature fusion directly through element-level addition or channel-level stitching can affect the detection of

object-bounding boxes [28]. A feature-alignment module is proposed to address the above issues. This module can calibrate the underlying features by combining high-level features before feature fusion. The feature-alignment module is shown in Figure 5.

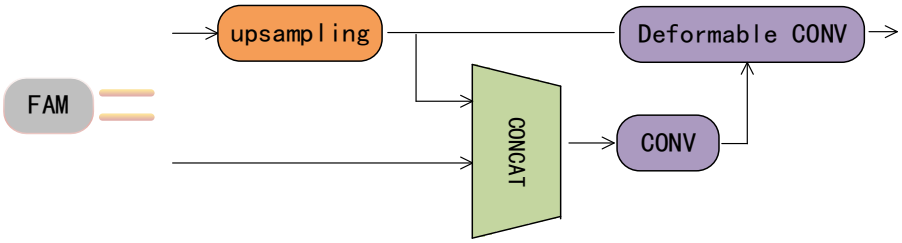


Figure 5. Feature-alignment module (FAM).

The feature-alignment module up-samples low-level information to ensure consistent feature-map size and then concatenates the feature map with high-level feature maps. The offset is learned through convolution and input into deformable convolution to perform feature alignment on the up-sampled feature map. Spatial-position information is represented through two-dimensional feature maps, where each offset includes the offset distance and corresponding point of each point, as shown in Formulas (6) and (7):

$$\hat{P}_i^u = f_a(P_i^u, \Delta_i) \tag{6}$$

$$\Delta_i = f_o([\hat{C}_{i-1}, P_i^u]) \tag{7}$$

$\hat{C}_{i-1}$ Represents the $(i-1)$th layer feature map, P_i^u represents the result of up-sampling the ith layer feature, and $[\hat{C}_{i-1}, P_i^u]$ is a concatenation of $\hat{C}_{i-1}$ and P_i^u, which provides the spatial difference between up-sampling and corresponding bottom-up features. $f_a(\cdot)$ and $f_o(\cdot)$ represent the learning offset Δ_i derived from spatial differences and aligned features with the learning offset, respectively.

Using deformable convolution for feature alignment, first, an input-feature map and a convolutional layer are defined, and then, the output features at any position $\hat{x}_P$ are obtained after the convolutional kernel.

$$\hat{x}_P = \sum_{n=1}^{N} W_n \cdot x_P + P_n \tag{8}$$

N is the size of the convolutional layer, W_n is the weight of the nth convolution, and P_n is the pre-specified offset.

In order to adaptively adjust for different sample positions, in addition to pre-specified offsets, other offsets $\{\Delta p_1, \Delta p_2, \cdots, \Delta p_n\}$ need to be learned.

$$\hat{x}_P = \sum_{n=1}^{N} W_n \cdot x_P + P_n + \Delta P_n \tag{9}$$

Each ΔP_n is a tuple of (h, w), where $h \in (-H_i, H_i)$, $w \in (-W_i, W_i)$.

The improved YOLOv4 network is shown in Figure 1 and combines the feature-alignment module and feature-selection module to improve the PANet module of YOLOv4. Before the fusion of low-level and high-level features, the up-sampled features of the low-level features and the high-level features are removed from redundant information through the feature-selection module and then input into the feature-alignment module for feature alignment.

3.3. Focal-EIoU Loss Function

The target detection uses bounding-box regression to locate the target in the image, and the early target detection uses IOU as the loss function of the location. However,

when the prediction boxing does not overlap the real boxing, the gradient of the IOU loss function disappears, resulting in a slower convergence speed and an inaccurate detector. This inspired several improved loss-function designs based on IOU, including GIOU, DIOU, and CIOU. Whereas GIOU introduces a penalty term in the IOU loss function to alleviate the problem of gradient disappearance, YOLOv4 uses the CIOU loss function. The loss function considers the center-point distance and width–height ratio between the prediction box and the real box in the penalty term, but the relative ratio of width and height is not a very direct indicator, so it is proposed to use the side length as a more direct penalty term. In addition, in order to solve the problem of severe oscillation of loss value caused by low-quality samples, the combination of EIOU loss and focal loss forms the Focal-EIOU loss function.

3.4. Soft-NMS

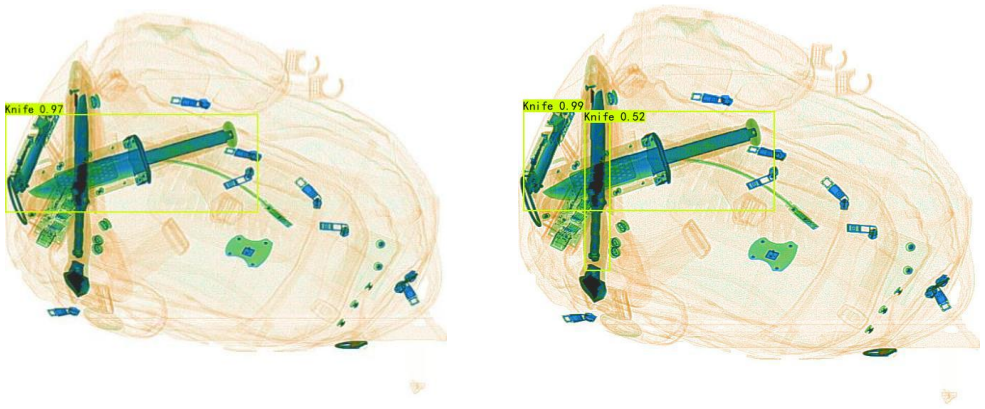
After the YOLOv4 network detects images, a target may generate a large number of bounding boxes, and the final output is only the optimal bounding box corresponding to the target, which requires a non-maximum-suppression algorithm. This algorithm sorts the confidence levels of all bounding boxes and deletes excess bounding boxes in descending order of confidence levels. The approach of the non-maximum-suppression algorithm in YOLOv4 is to select the bounding box with the highest confidence, calculate the DOIU between the remaining bounding boxes, delete them if they are greater than the set threshold and retain them if they are less than the set threshold, and then select the bounding box with the second highest confidence to perform this operation, and so on until all filtering is completed. The non-maximum-suppression algorithm mainly uses this iterative form to continuously perform DIOU operations with other boxes with the highest confidence and filter out boxes with a larger DIOU.

The specific operation of Soft-NMS has three inputs: detection-box set B, confidence set S, threshold N, and a set D used to store the final detection box. When set B is not empty, the maximum confidence in set S is found. Assuming the subscript of this maximum confidence is m, there is also a corresponding detection box b_m in set B, which is the corresponding detection box for this confidence. Then, detection box b_m in stored in M, and it is merged and delete from the B set. Each detection box b_m is looped through in set B and the DIOU is calculated, and the combined effect of the DIOU value and confidence is used as the final score of the detection box. Formula (10) provides the definition of Soft-NMS as follows in order to change the practice of directly deleting detection boxes with high overlap in NMS and follow the principle that the larger the DIOU, the lower the score:

$$s_i = \begin{cases} s_i, & DIOU(\mathrm{M}, b_i) < N_t \\ s_i(1 - DIOU(\mathrm{M}, b_i)), & DIOU(\mathrm{M}, b_i) \geq N_t \end{cases} \tag{10}$$

s_i is the confidence level of the object, $\mathrm{DIOU}(\mathrm{M}, b_i)$ represents the size of the DIOU between detection boxes, and N_t is the set threshold.

The processing results using two different NMS algorithms are shown in Figure 6. Figure 6a shows the results of the traditional NMS method, which may miss detection when two objects overlap. This is because when two objects overlap, the DIOU threshold is directly used to determine whether to delete the bounding box, which may delete the exact bounding box of the other object. The Soft-NMS algorithm no longer only uses the DIOU to determine whether to delete the bounding box but also considers both the DIOU and the confidence size when deleting the bounding box and gives the correct detection results, as shown in Figure 6b.



(**a**) NMS (**b**) Soft-NMS

Figure 6. The processing results using two different NMS algorithms.

3.5. Improved Channel Pruning

There is a large number of redundant parameters in the operation process of convolutional neural networks, which have little effect on improving the detection accuracy of the model and may even lead to a decrease in accuracy. Effectively removing these redundant parameters can improve the detection speed of the network. Channel pruning can delete entire redundant channels while preserving the original convolutional structure, improving network-detection performance without losing accuracy. This design aims to create a channel-pruning method for the YOLOv4-PANv2 network, which considers channel pruning as a search problem, and the legitimate pruning network in the search space is called a subnet. The adaptive BN method is used to evaluate the subnet, which can accurately and quickly find the optimal subnet. The overall pruning process is shown in Figure 7. Firstly, n subnets are generated through the pruning strategy, and each subnet is evaluated by adaptive BN to select the one with the highest score. After several epochs, the pruned model is determined.

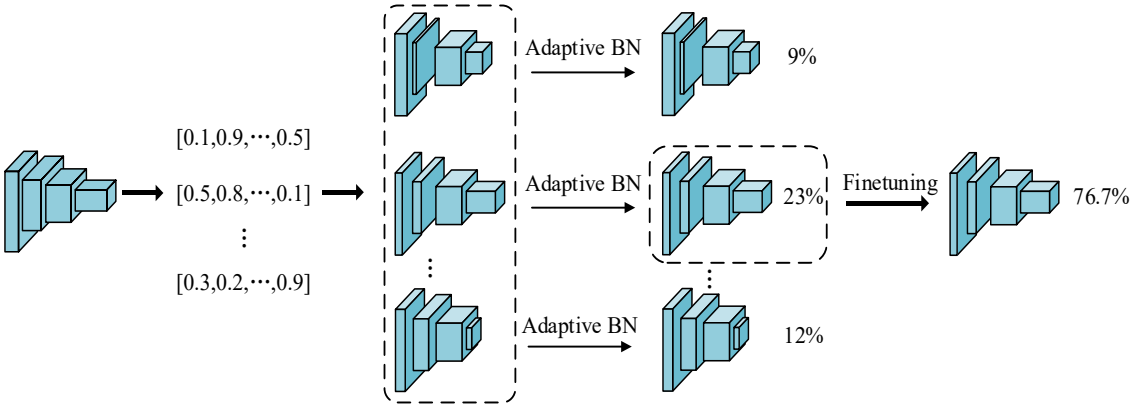


Figure 7. The flow chart of channel pruning.

The detection accuracy of the subnet generated by the pruning strategy can only be tested after the training process converges. However, each subnet is evaluated by this method, which requires not only fine hyperparameter adjustment but also an extra time-consuming training process. This design uses an adaptive BN-evaluation method that can quickly and accurately select subnets with good final-testing performance. The original BN formula is written as follows:

$$y = \gamma \frac{x - u}{\sqrt{\sigma^2 + \varepsilon}} + \beta \tag{11}$$

where β and γ represent the scaling factors and offsets learned through training, respectively, and μ and σ^2 represent the mean and variance of the current batch for small batch size of N, respectively.

$$u_B = E[x_B] = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{12}$$

$$\sigma_B^2 = V_{ar}[x_B] = \frac{1}{N-1}\sum_{i=1}^{N} (x_i - u_B)^2 \tag{13}$$

If the same method is used to normalize a batch of samples that need to be predicted during testing, uncertainty occurs in the prediction results. Therefore, during the testing phase, the global-BN statistics are used to calculate μ_T and σ_T^2 as follows:

$$u_t = mu_{t-1} + (1-m)u_B \tag{14}$$

$$\sigma_t^2 = m\sigma_{t-1}^2 + (1-m)\sigma_B^2 \tag{15}$$

where m is the momentum coefficient, and the subscript t represents the number of training iterations. The statistical information of BN is not learned through training but rather is obtained through data statistics. During testing, global-BN statistical information is required to stabilize performance. However, there may be a mismatch between the global-BN statistical information of the pruned subnet and the global-BN statistical information before pruning, leading to unstable detection performance in direct testing. The general method is to train the subnet for several epochs before accurately testing its subnet performance. Adaptive BN inputs data into the network for several epochs of inference, which resamples the subnets and solves the problem of statistical-data mismatch. The use of the adaptive-BN method can quickly select subnets with excellent performance.

4. Results

All experiments in this paper were implemented under the Linux operating system using the Python language and Pytorch framework and a GPU-accelerated NVIDIA A100-PCIe graphics card, and the CPU was Intel(R) Xeon(R) Gold 6330 CPU @ 2.00 GHz.

This paper selected the public dataset Security Inspection X-ray Benchmark (SIXray) for the experiments [24], which was released by the University of Chinese Academy of Sciences. The SIXray dataset contains more than one million security-screening images from subway stations, which provided enough training and testing data. Moreover, the images in the dataset have high clarity and resolution, covering many of the most common items in security tasks. Each image provides detailed labeling information, including the type, location, and size of the items. In this experiment, 8929 positive sample images were used, including five categories of guns, knives, wrenches, pliers, and scissors. The quantity distribution of each category is shown in Table 1.

Table 1. The distribution of the number of categories in the dataset.

Category	Guns	Knives	Wrenches	Pliers	Scissors
Number	3131	1943	2199	3961	983

The dataset was divided into two parts, with 80% of the samples as the training set and 20% of the samples as the test set. The maximum learning rate was 1×10^{-3}, and the minimum learning rate was 1×10^{-6}. Using the label-smoothing strategy and cosine-annealing optimization algorithm, we learned to first simulate the rapid decline of the function and then linearly increase and repeat the process continuously. The epoch was set to 100. First, the influence of the Focal-EIOU loss function on the network was verified. The original CIOU loss function and the Focal-EIOU loss function were used to train the YOLOv4 network. The visualization results of the loss function are shown in Figure 8. The network using the CIOU loss function began to converge to a stable level at the 30th

epoch, whereas the network using the Focal-EIOU loss function began to converge at the 19th epoch, indicating that it had a faster convergence rate.

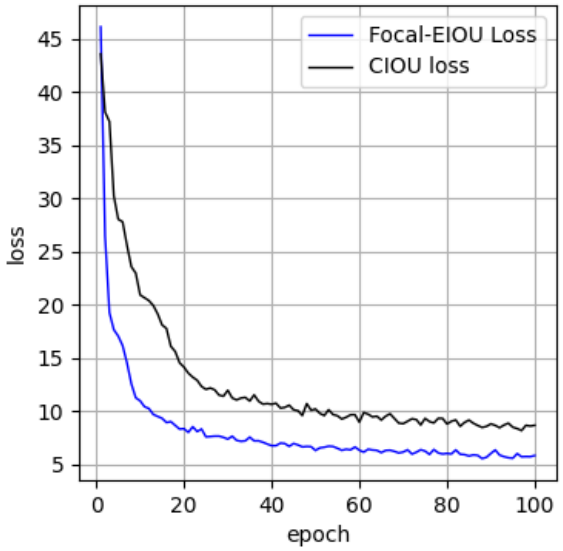


Figure 8. Diagram of the loss-visualization results.

In order to verify the effectiveness of the proposed method in this paper in improving the accuracy of the YOLOv4 network, we compared it with other object-detection models, including YOLOv3, M2Det [29], SSD [30], and YOLOv4, as shown in Table 2.

Table 2. Detection results with different models.

Method	Guns (%)	Knives (%)	Wrenches (%)	Pliers (%)	Scissors (%)	mAP (%)
YOLOv3	93.18	78.00	68.55	79.69	76.97	79.28
M2Det	95.49	75.70	70.17	83.00	82.96	81.47
SSD	94.91	77.87	74.82	84.51	82.69	82.96
YOLOv4	94.40	81.69	77.38	84.50	77.55	83.11
YOLOv4-PANv2	95.73	83.00	82.95	85.13	80.74	85.51

The mAP of the improved model was the highest, reaching 85.51%, which was 6.23%, 4.04%, 2.55%, and 2.4% higher than that of the YOLOv3, M2Det, SSD, and YOLOv4 models, respectively. Compared with YOLOv4, the AP value of guns increased by 1.33%, that of knives increased by 1.31%, that of wrenches increased by 5.57%, that of pliers increased by 0.63%, and that of scissors increased by 3.19%.

In order to further analyze the performance of the different categories, the AP, precision, recall and F1 measure of each category of the YOLOv4-PANv2 network were analyzed, as shown in Table 3. It can be seen from the table that the AP, precision, recall, and F1 measure of guns were the highest, because the number of gun samples in the dataset was the largest. The recall of knives was relatively low. Because knives are usually thin in shape and can easily overlap with other dangerous goods, there is more missed detection. The precision of the five types of dangerous goods was maintained at a good level, indicating that the model has strong generalization ability and a low false-detection rate.

Table 3. Improved model-performance analysis for each category.

Category	AP (%)	Precision (%)	Recall (%)	F1 Measure
Guns	95.73	98.59	84.92	0.91
Knives	83.00	91.43	67.20	0.77
Wrenches	82.95	83.66	71.95	0.77
Pliers	85.13	92.95	74.23	0.82
Scissors	80.74	86.88	75.13	0.81

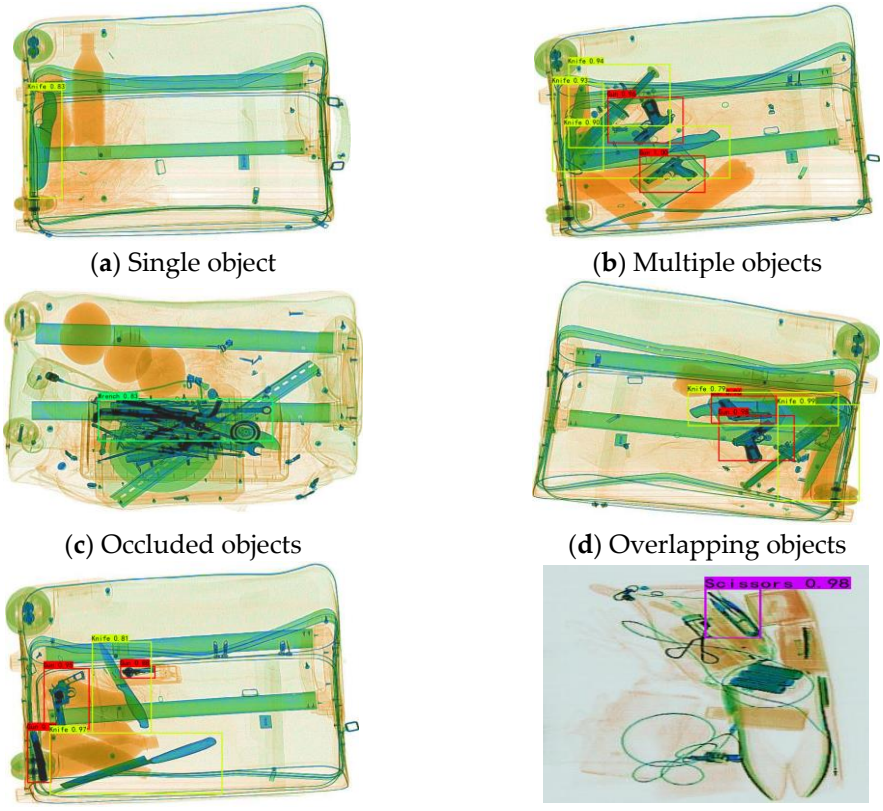
The average value of various objects was taken as the overall performance-evaluation index and compared with the original YOLOv4, as shown in Table 4.

Table 4. Comparison of YOLOv4 performance before and after improvement.

Method	mAP (%)	Precision (%)	Recall (%)	F1 Measure
YOLOv4	83.11	90.35	73.00	0.80
YOLOv4-PANv2	85.51	90.70	74.69	0.82

As can be seen from the table, compared with the original YOLOv4 model, the mAP, accuracy, and recall rate of the YOLOv4-PANv2 model increased by 2.4%, 0.35%, and 1.69%, respectively, which means that all indicators of our detection algorithm were improved and the improved model had better detection performance.

The detection performance of the model in the test set is shown in Figure 9, where (a) is a single-target image, (b) is a multi-target image, (c) is an occluded-target image, (d) is an overlapping image, (e) is a placement-difference image, and (f) is a small-target image. It can be seen that the improved model could accurately detect each target with our proposed method.



(**a**) Single object (**b**) Multiple objects

(**c**) Occluded objects (**d**) Overlapping objects

(**e**) Objects with placement difference (**f**) Small objects

Figure 9. Detection results for dangerous goods.

5. Discussion

5.1. Ablation Experiment

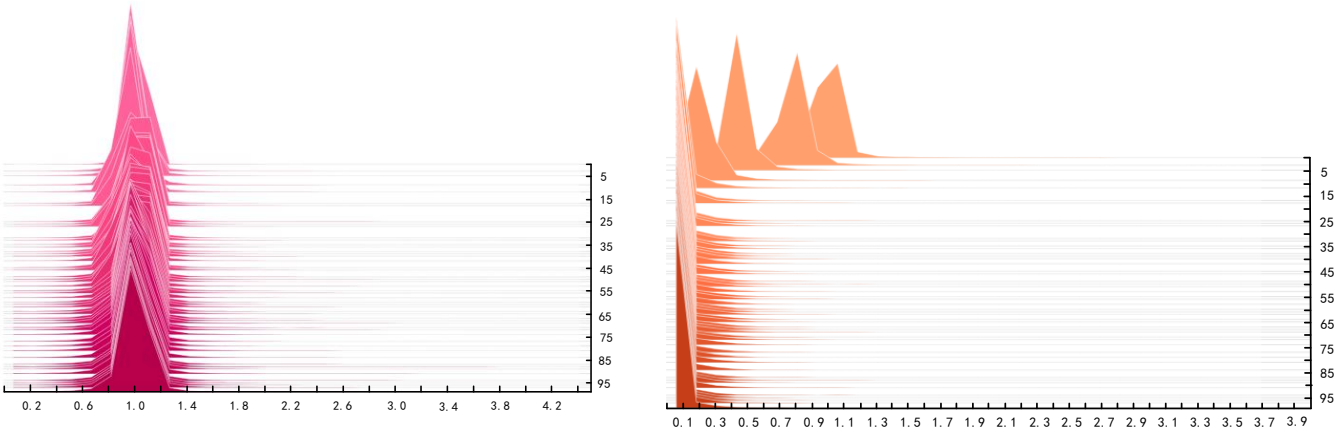
In order to verify the model-detection effect after adding different improvement points, ablation experiments and analysis were conducted on the improved PANet module, Focal-EIOU loss function, and Soft-NMS. The comparison results of the ablation experiments are shown in Table 5. It can be seen from the table that mAP was significantly improved after the PANet module was improved, indicating that the feature-selection module and feature-alignment module were used in the process of low-level and high-level feature fusion, which significantly improved the detection performance. After using the Focal-EIOU loss function, the model mAP was slightly improved. It can be seen that the Focal-EIOU loss function not only accelerated the convergence of the model but also improved the detection performance of the model. Finally, Soft-NMS also increased the mAP of the model by 0.9%, which more accurately deleted redundant boundary boxes in the processing of images with a high overlap rate.

Table 5. Comparison results of ablation experiment.

Method	Improved PAN	Focal-EIOU Loss	Soft-NMS	mAP (%)
YOLOv4	×	×	×	83.11
YOLOv4-Improved PAN	√	×	×	85.06
YOLOv4-Focal-EIOU Loss	×	√	×	83.96
YOLOv4-Soft-NMS	×	×	√	84.01
YOLOv4-PANv2	√	√	√	85.51

5.2. Analysis of Channel-Pruning Performance

Although the YOLOv4-PANv2 network improved detection accuracy compared to the YOLOv4 network, the detection speed is not ideal and it would be difficult to meet the detection speed required for X-ray security when there are many bags. To address the above issues, a channel-pruning algorithm was designed to reduce the redundant convolutional channels of the model while not reducing accuracy in order to reduce the computational complexity and improve the detection speed of the model. The distribution of scaling factors before and after sparse training is shown in Figure 10.



(**a**) Base model (**b**) Sparse model

Figure 10. Distribution of scaling factors for sparse training.

Figure 10a shows the distribution of scaling factors in the BN layer during normal training, and Figure 10b shows the distribution of scaling factors in the BN layer during sparse training. The x-axis coordinate represents the size of the scaling factor, the y-axis coordinate represents the epoch of the training, and the z-axis coordinate represents the number of scaling factors. The use of L1 regularization terms effectively suppressed the

scaling factor corresponding to channels that made little contribution to the model, making it approach 0. Channels with relatively large contributions also had their scaling factor suppressed, but the final value was also relatively large.

The statistical distribution of the scaling factor is shown in Figure 11. The horizontal axis represents the current epoch, whereas the vertical axis represents the numerical value of the scaling factor. A higher value indicates a higher quantity. In the end, all scaling factors were basically compressed to 0, and sparse training had a significant effect.

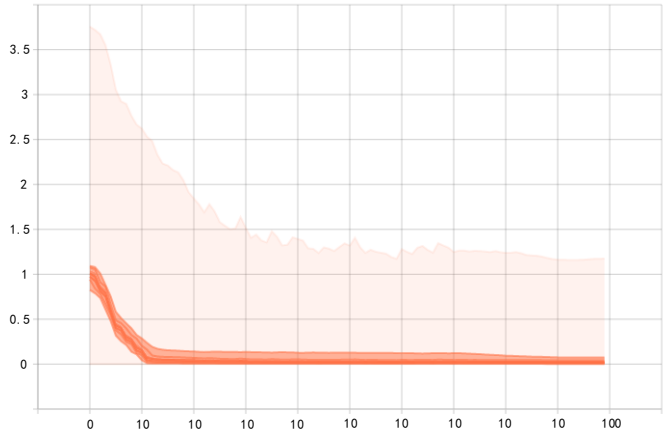


Figure 11. Statistical distribution of scaling factors.

To verify the effectiveness of channel pruning, the detection performance of the YOLOv4-PANv2 network before and after pruning was compared. The detection results are shown in Table 6. From the table, it can be seen that after channel pruning, the model's FLOPs decreased by 20.38 G, FPS increased by 23.70 f/s, and mAP only decreased by 0.73%, which was acceptable. After pruning, the computational complexity of the model was significantly reduced, and FPS was significantly improved. This is because the channel-pruning algorithm removed many redundant channels, reducing the calculation parameters and the number of calculations, which can effectively improve the detection speed of the model for X-ray security-inspection images.

Table 6. Detection results after model pruning.

Method	FLOPs (G)	FPS (f/s)	mAP (%)
YOLOv4-PANv2	60.52	26.71	85.51
Prune-YOLOv4	40.14	50.41	84.78

6. Conclusions

This paper proposes a combination of deformable convolution and the PANet module of the YOLOv4 network, and designs and implements a dangerous-goods-detection algorithm for X-ray security-inspection images. The deformable convolution with a more flexible receptive field is used to solve feature misalignment in the fusion of high- and low-level features of YOLOv4. The pruning strategy uses adaptive BN to evaluate subnets, which can quickly select subnets with good detection results. While ensuring the accuracy of the detection model, it greatly reduces the FLOPs of the model and improves the FPS of the detection model. Experimental results show that our improved YOLOv4 method can effectively improve the accuracy of the detection network and meet the requirements of X-ray security detection in terms of accuracy and speed. Future work will use lightweight methods to reduce the number of parameters and computation while ensuring detection accuracy, making it easier for embedded implementation.

Author Contributions: Conceptualization, X.Y., A.W. and W.Y.; methodology, X.Y., A.W. and W.Y.; software, W.Y.; validation W.Y.; writing—review and editing X.Y., W.Y. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the High-End Foreign Experts Introduction Program (G202201-2010L) and the Major Science and Technology Projects of Zhongshan City in 2022 (2022A1020).

Data Availability Statement: https://hyper.ai/datasets/18691.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, Y.; Xu, T.; Zhao, B.; Li, T.; Wang, D. X-ray and Infrared Image Fusion in Security Field. In Proceedings of the 2019 IEEE International Conference of Intelligent Applied Systems on Engineering (ICIASE), Fuzhou, China, 26–29 April 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 16–19.
2. Chang, A.; Zhang, Y.; Zhang, S.; Zhong, L.; Zhang, L. Detecting Prohibited Objects with Physical Size Constraint from Cluttered X-ray Baggage Images. *Knowl. Based Syst.* **2022**, *237*, 107916. [CrossRef]
3. Bastan, M. Multi-View Object Detection in Dual-Energy X-ray Images. *Mach. Vis. Appl.* **2015**, *26*, 1045–1060. [CrossRef]
4. Kundegorski, M.E.; Akçay, S.; Devereux, M.; Mouton, A.; Breckon, T.P. On Using Feature Descriptors as Visual Words for Object Detection within X-ray Baggage Security Screening. In Proceedings of the 7th International Conference on Imaging for Crime Detection and Prevention (ICDP 2016), Madrid, Spain, 23–25 November 2016; pp. 1–6.
5. Mery, D.; Katsaggelos, A.K. A Logarithmic X-ray Imaging Model for Baggage Inspection: Simulation and Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–27 July 2017; pp. 251–259.
6. Xing, X.; Hu, H.; Xu, Y. Security image interpretation based on gray projection algorithm and FPGA implementation method. *Technol. Rev.* **2018**, *18*, 42–44.
7. Russo, A.U.; Deb, K.; Tista, S.C.; Islam, A. Smoke Detection Method Based on LBP and SVM from Surveillance Camera. In Proceedings of the 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2), Rajshahi, Bangladesh, 8–9 February 2018; pp. 1–4.
8. Santos, A.; Abu, P.A.; Oppus, C.; Reyes, R. Traffic Sign Detection and Recognition for Assistive Driving. In Proceedings of the 2019 International Symposium on Multimedia and Communication Technology (ISMAC), Quezon City, Philippines, 19–21 August 2019; pp. 1–6.
9. Hai, L.; Xiaoyun, Y.; Sihai, F. Fusion of Bilateral filtering and homomorphic filtering based on X-ray image recognition. *Comput. Digit. Eng.* **2019**, *47*, 1120–1124+1130.
10. Liang, K.J.; Heilmann, G.; Gregory, C.; Diallo, S.O.; Carlson, D.; Spell, G.P.; Carin, L. Automatic Threat Recognition of Prohibited Items at Aviation Checkpoint with X-ray Imaging: A Deep Learning Approach. In Proceedings of the Anomaly Detection and Imaging with X-rays (ADIX) III, Orlando, FL, USA, 17–18 April 2018; SPIE: Bellingham, WA, USA, 2018; Volume 10632, p. 1063203.
11. Mery, D.; Svec, E.; Arias, M.; Riffo, V.; Saavedra, J.M.; Banerjee, S. Modern Computer Vision Techniques for X-ray Testing in Baggage Inspection. *IEEE Trans. Syst. Manag.* **2017**, *47*, 682–692. [CrossRef]
12. Akcay, S.; Breckon, T.P. An Evaluation of Region Based Object Detection Strategies within X-ray Baggage Security Imagery. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 1337–1341.
13. Zhu, X.; Zhang, C.; Xie, W.; Zhang, D. Server Monitoring System Using an Improved Faster RCNN Approach. In Proceedings of the 2017 11th IEEE International Conference on Anti-Counterfeiting, Security, and Identification (ASID), Xiamen, China, 27–29 October 2017; pp. 50–53.
14. Singh, B.; Li, H.; Sharma, A.; Davis, L.S. R-FCN-3000 at 30fps: Decoupling Detection and Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 1081–1090.
15. Zhang, Z.; Qiao, S.; Xie, C.; Shen, W.; Wang, B.; Yuille, A.L. Single-Shot Object Detection with Enriched Semantics. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 5813–5821.
16. Guo, R.; Zhang, L.; Ying, Y.; Sun, H.; Han, Y.; Tan, H. Automatic Detection and Identification of Controlled Knives Based on Improved SSD Model. In Proceedings of the 2019 Chinese Automation Congress (CAC), Hangzhou, China, 22–24 November 2019; pp. 5120–5125.
17. Gaus YF, A.; Bhowmik, N.; Akcay, S.; Breckon, T. Evaluating the Transferability and Adversarial Discrimination of Convolutional Neural Networks for Threat Object Detection and Classification within X-ray Security Imagery. In Proceedings of the 2019 18th IEEE International Conference on Machine Learning And Applications (ICMLA), Boca Raton, FL, USA, 16–19 December 2019; pp. 420–425.
18. Haoyang, T.; Yan, W.; Xiaoyuan, Z. Dangerous goods detection algorithm by X-ray machine based on feature pyramid. *Xian Univ. Posts Telecommun.* **2020**, *25*, 58–63.

19. Shi, Y.; Xu, Y.; Wei, L.; Gao, H.; Xu, X. X-DOG: An Intelligent X-ray-based Dangerous Goods Detection and Automatic Alarm System. In Proceedings of the 2020 International Conferences on Internet of Things (iThings), Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), Rhodes, Greece, 2–6 November 2020; pp. 576–582.
20. Shouxiang, G.; Liang, Z. Yolo C: One Stage Network for Prohibited Items Detection within X-ray lmages. *Laser Optoelectron. Prog.* **2021**, *58*, 75–84.
21. Tang, C.; Wu, Z.; Wang, S.; Deng, C.; Luo, L. Industrial Object Detection Method Based on Improved CenterNet. In Proceedings of the 2021 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI), Shanghai, China, 27–29 August 2021; pp. 121–125.
22. Kumar, R.S.; Balaji, A.; Singh, G.; Kumar, A.; Manikandaprabu, P. Recursive CNN Model to Detect Anomaly Detection in X-ray Security Image. In Proceedings of the 2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM), Pradesh, India, 23–25 February 2022; pp. 742–747.
23. Jiang, C.; Zhang, H.; Yue, Y.; Hu, X. AM-YOLO: Improved YOLOV4 based on attention mechanism and multi-feature fusion. In Proceedings of the 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 15–17 September 2022; pp. 1403–1407.
24. Miao, C.; Xie, L.; Wan, F.; Su, C.; Liu, H.; Jiao, J.; Ye, Q. SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2119–2128.
25. Wei, Y.; Tao, R.; Wu, Z.; Ma, Y.; Zhang, L.; Liu, X. Occluded Prohibited Items Detection: An X-ray Security Inspection Benchmark and Deocclusion Attention Module. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 138–146.
26. Gu, B.; Ge, R.; Chen, Y.; Luo, L.; Coatrieux, G. Automatic and Robust Object Detection in X-ray Baggage Inspection Using Deep Convolutional Neural Networks. *IEEE Trans. Ind. Electron.* **2021**, *68*, 10248–10257. [CrossRef]
27. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable Convolutional Networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, ltaly, 22–29 October 2017; pp. 764–773.
28. Huang, S.; Lu, Z.; Cheng, R.; He, C. FaPN: Feature-aligned Pyramid Network for Dense Image Prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021; pp. 864–873.
29. Zhao, Q.; Sheng, T.; Wang, Y.; Tang, Z.; Chen, Y.; Cai, L.; Ling, H. M2Det: A Single-Shot Object Detector Based on Multi-Level Feature Pyramid Network. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 9259–9266. [CrossRef]
30. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision (CVPR), Amsterdam, The Netherlands, 11–14 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 21–37.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Crowd Counting by Multi-Scale Dilated Convolution Networks

Jingwei Dong [1,*], Ziqi Zhao [1] and Tongxin Wang [2]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China; 1605040126@stu.hrbust.edu.cn
[2] Technical Department, Beijing Duyou Information Technology Co., Ltd., Beijing 100085, China; wangtongxin@baidu.com
* Correspondence: djw@hrbust.edu.cn

Abstract: The number of people in a crowd is crucial information in public safety, intelligent monitoring, traffic management, architectural design, and other fields. At present, the counting accuracy in public spaces remains compromised by some unavoidable situations, such as the uneven distribution of a crowd and the difference in head scale caused by people's differing distances from the camera. To solve these problems, we propose a deep learning crowd counting model, multi-scale dilated convolution networks (MSDCNet), based on crowd density map estimation. MSDCNet consists of three parts. The front-end network uses the truncated VGG16 to obtain preliminary features of the input image, with a proposed spatial pyramid pooling (SPP) module replacing the max-pooling layer to extract features with scale invariance. The core network is our proposed multi-scale feature extraction network (MFENet) for extracting features in three different scales. The back-end network consists of consecutive dilation convolution layers instead of traditional alternate convolution and pooling to expand the receptive field, extract high-level semantic information and avoid the spatial feature loss of small-scale heads. The experimental results on three public datasets show that the proposed model solved the above problems satisfactorily and obtained better counting accuracy than representative models in terms of mean absolute error (MAE) and mean square error (MSE).

Keywords: deep learning; crowd counting; density map estimation; spatial pyramid pooling (SPP); multi-scale feature extraction; dilated convolution

1. Introduction

The rapid development of monitoring equipment has led to a continuous increase in the amount of image and video data, while the demand for video analysis is also increasing. Crowd counting is an important aspect of video analysis, especially in crowded public places where avoiding public safety accidents through early warning is crucial [1–3].

There are three types of approaches to counting the number of people in a crowd. The most direct method is to detect each person in an image one by one through object detection methods and to accumulate the objects to obtain the final count [4]. The detection-based approaches can achieve satisfactory results in sparse crowds without serious occlusion [5]. The second type of crowd counting method is to extract the foreground features of an image and build the regression model to learn the mapping from the features to people number [6]. Thus, the regression-based approaches avoid object detection in the counting and simplify the task [7]. However, this type of method ignores the spatial information of the crowd distribution in the real scene and only focuses on the number of targets in the image. The third type of crowd counting, density estimation-based methods, does not detect or locate objects in an image, but maps the local image features and density distribution on the corresponding area, estimates the density map, and integrates the density map over the area to obtain the number of people in that region [8,9]. It turns the counting problem into a problem of estimating the density of the image.

With the advent of the convolutional neural network (CNN), deep learning models have been introduced into crowd counting and density estimating, and have become the mainstream research direction [10,11]. In 2012, AlexNet [12] won the ImageNet Challenge, using ReLU as the activation function, and using dropout to randomly kill some neurons during training to avoid over-fit. Because of the powerful parallel processing capability of GPU, training on the deep convolution network is possible. After AlexNet achieved good results in target classification and counting, more and more deep networks emerged, such as the well-known Inception (GoogLeNet) [13] and VGG series [14]. In surveillance videos, people inevitably appear in different scales because of their differing distances from the surveillance camera, which is often installed at a high altitude. Without scale invariance, the CNN-based models could not obtain accurate counting results until Zhang et al. proposed a multi-column convolutional neural network (MCNN) in 2016 [15]. In this classical model, they designed three convolution network columns with different scales and cascaded feature maps extracted to obtain head feature maps at large, medium, and small scales. The crowd density map was then obtained by fusing it to the back-end network. To fully cover all challenging scenarios, a new large-scale dataset called Shanghai_Tech was collected and labeled. Since then, how to extract features with scale invariance from images and improve the accuracy of density estimation has become a lively point of discussion in crowd counting research.

In order to solve the problem of scale variance and crowd uneven distribution in crowd counting, we propose a deep learning model to learn the non-linear relationship between the input crowd image and the density map. By integrating the density map, the crowd in the input image is then statistically counted. The contributions of this paper can be summarized as follows:

1. We propose a multi-scale dilated convolution network (MSDCNet) to estimate the density map of an input crowd image. A multi-scale feature extraction network is designed as the core component of MSDCNet.
2. In the front-end network of MSDCNet, we propose SPP modules to replace max-pooling layers of VGG16 that can extract features with scale invariance. In the back-end network of MSDCNet, we use the dilated convolution layers to replace the traditional alternate convolution and pooling to expand the receptive field, extract high-level semantic information and avoid the loss of spatial information at the same time.
3. The effectiveness of our approach is validated on three public datasets, Shanghai_Tech, UCF_CC_50, and UCF_QNRF. Compared with other representative crowd counting models, our model has a better counting performance.

In the rest of this paper, Section 2 includes the recent work related to density estimation. The framework and principle of MSDCNet are described in Section 3. The datasets, experimental results and related analysis are given in Section 4. Conclusions and suggestions for future work are given in Section 5.

2. Related Work

2.1. Density Map

A crowd counting sample generally consists of a crowd image and its label matrix which is the same size as the image. The label matrix stores numbers 0 and 1 that represent the heads' center-point positions. A visualized label matrix and the corresponding crowd image are shown in Figure 1. The main issue of crowd counting is to find the mapping between input images and density maps through extensive training. The training process of crowd counting models is shown in Figure 2. The preprocessing of training samples includes two objectives: generating ground truth density maps and data augmentation. A sample in datasets provides an image and a set of coordinates of each head-center-point instead of the label matrix mentioned above. The coordinates need to be converted into a label matrix and then into a smooth and continuous density map to serve as the ground truth density map of the training sample (see Section 2.2 for details). Due to the differences

in the number of samples, image size, and crowd density among different datasets, the data augmentation for different datasets also varies (see Section 4.1 for details).

Figure 1. Visualization of a label matrix.

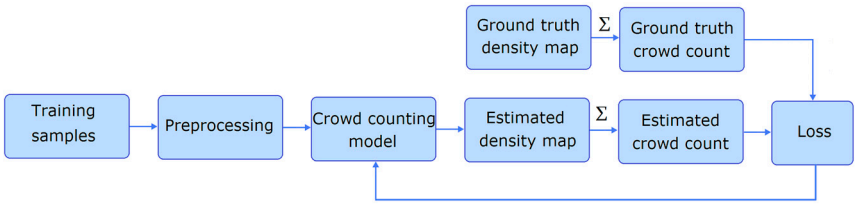


Figure 2. The training diagram of crowd counting models.

2.2. Generating Density Map Based on Gaussian Kernel

First, we initialized a zero matrix with the same size as the input image. Based on the known position coordinates of each head, we set the corresponding elements to 1, and obtained a label matrix composed of 0 and 1. Using the Gaussian kernel and the label matrix for convolution, we obtained a smooth and continuous density map. Due to the variance in the heads' scale in an image, we adopted an adaptive Gaussian kernel whose size varies with the heads' scale. After convolution, the 1 at a certain position in the label matrix was replaced with multiple weights between 0 and 1 around it, and the sum of these weights was equal to 1. This not only did not affect the counting of the total number of heads through the density map, but also obtained the spatial information of each head. A ground truth density map obtained via the above method is visualized as a pseudo-color map shown in Figure 3.

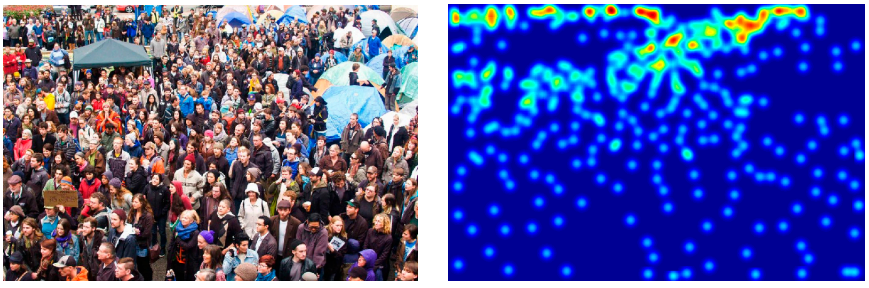


Figure 3. Generated Gaussian kernel density map by adaptive Gaussian kernel convolution.

2.3. Crowd Counting Models

In 2018, a single-column network, named the congested scene recognition network (CSRNet) [16], was proposed to extract more scale information by using dilated convolutions and expanding the receptive field. CSRNet obtained encouraging counting results in highly congested spaces. In 2020, Jiang et al. [17] proposed trellis encoder–decoder networks (TEDnet) to generate high-quality density estimation maps. They designed multiple decoding columns to hierarchically aggregate spatially endowed features at different

encoding stages. Some researchers found out different ways leading to the same target. In order to eliminate the distortion of the crowd image, a reverse perspective network was proposed by Yang et al. [18]. In 2021, a locality-aware crowd counting model was proposed to avoid serious deviation in counting results [19]. A Tencent YouTu team proposed a point-based approach [20]. A set of point proposals is given to represent heads. In [21], an unbalance-based generalized loss function was proposed to improve the training process of the crowd counting model. Introducing an attention mechanism into training is one of the solutions to scale differences in crowd counting [22]. The emergence of various innovative models and learning strategies is continuously improving the accuracy of crowd counting.

3. Proposed Approach

The proposed MSDCNet consists of three parts, a front-end network, a multi-scale feature extraction network (MFENet), and a back-end network, as shown in Figure 4. The crowd-counting task is to obtain the density map instead of classification, so we design it as fully convolutional networks (FCNs). The feature of FCN is that both the input and output are 2D images. The input can be images of any size, while the output size is the same as the input size and they have a corresponding spatial structure. The FCN consists of two parts: a full convolution part (the front-end network and MFENet) which is used to extract features, and a deconvolution part (the back-end network) which is used to obtain the density map estimation by up-sampling.

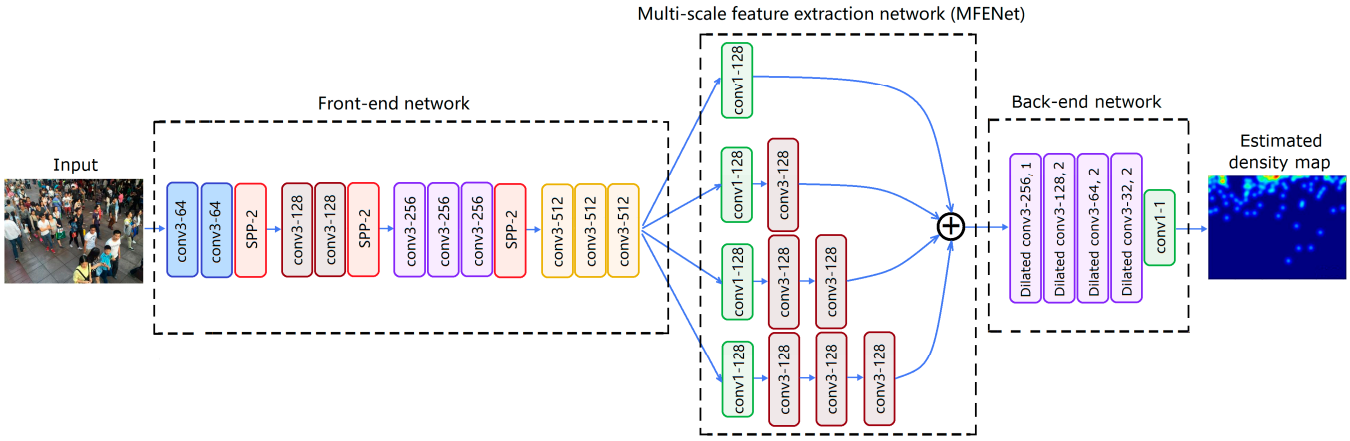


Figure 4. The framework of the MSDCNet. The convolution layers are denoted as "conv (size)-(number of filters)", the dilated convolution layers are denoted as "Dilated conv (size)-(number of filters), dilation rate".

3.1. Front-end Network

The front-end network is used to extract preliminary features from input images. Because VGG16 [14] performs well in object detection and classification, and has strong transfer learning ability, we used it as the backbone of the front end. VGG16 consists of 13 conv layers, 5 pooling layers, 3 fully connected (FC) layers and a soft-max output layer, as shown in Figure 5. The pooling layer is 2×2 max-pooling. After each pooling layer, the width and height of the feature map are reduced by half. With the increase in the network depth, the size of the feature map gradually decreases. The small feature size is not conducive to crowd counting especially in a converged scene. Therefore, to refer to CSRNet [16], the front-end network uses the first 10 conv layers of VGG16 and the 3 pooling layers between them. In order to learn features of multi-scale heads in crowd images, spatial feature fusion is performed in pooling layers in our model. We propose a pooling module based on the SPP idea (denoted as the SPP module) to replace the max-pooling layer.

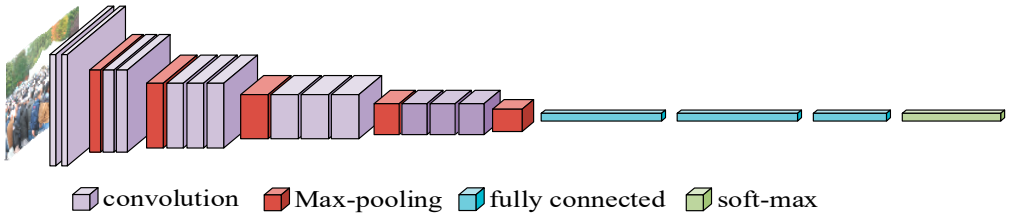


Figure 5. The framework of VGG16.

3.1.1. SPP Module

In 2015, He et al. [23] proposed spatial pyramid pooling (SPP) to address the input size limitation of the standard CNN. As shown in Figure 6, the SPP layer is used after the last conv layer (conv5, with 256 filters) and before the FC layers (fc6, fc7). The SPP extracts the features from the output of conv5 through pooling bins (windows) with different sizes. Each bin obtains one output, so the size of the SPP output features is determined by the number of bins instead of the size of the input. Compared to the breaking fixed size limitation, the most important advantage of the SPP is found by researchers to be that different sizes of bins correspond to different regions of the original image, i.e., to objects of different scales. This greatly improves the robustness of the network to object scaling. The SPP is therefore an effective way to handle multi-scale networks.

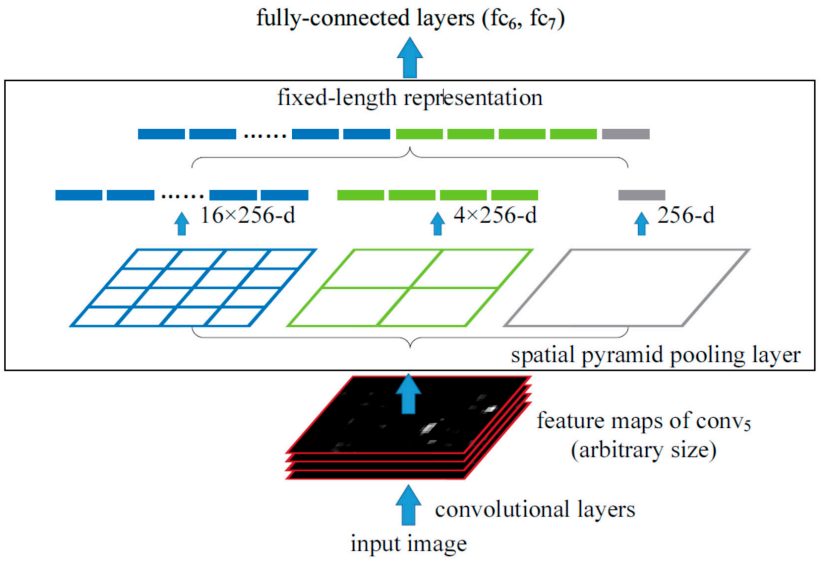


Figure 6. The network structure with a SPP layer.

Inspired by the different sizes of bins in the SPP, we proposed a SPP module with different pooling kernel sizes to replace the 2×2 max pooling layers in VGG16 and to further extract the features with scale invariance. Identical to the original 2×2 max-pooling, our SPP module reduces the width and height of the feature map by half, too. Therefore, it is denoted as "SPP-2" in Figure 4. As shown in Figure 7, the SPP module is a stack of three max-pooling layers. The size of the input feature maps is $W \times H$, and the sizes of the three pooling kernels we used are 2×2, 2×2, and 3×3, and the strides are 2, 1, and 1, respectively. Finally, on every channel, the obtained three feature maps of size $W/2 \times H/2$ are summed up. With three SPP modules fusing the spatial features, the front-end network can expand the receptive field and extract features in different scales without increasing the parameters.

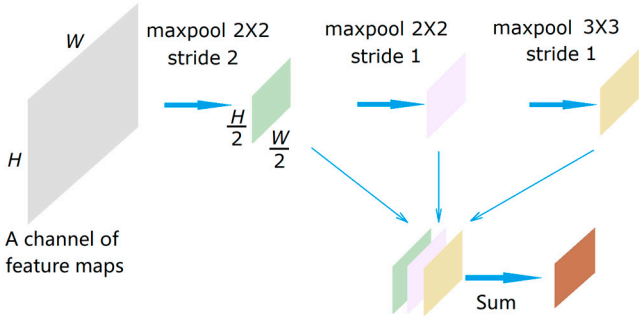


Figure 7. The framework of the proposed SPP module.

3.1.2. Architecture of Front-End Network

The front-end network consists of 10 conv layers and 3 SPP modules, and its architecture is shown in Table 1. After three SPP modules, the size of the feature map output by the front-end network becomes $W/8 \times H/8$.

Table 1. The architecture of the front-end network. The conv layers are denoted as "conv (size)-(number of filters)". The ReLU activation function is not shown for brevity.

Layers	Kernel	Stride
Conv1	conv3-64	1
Conv2	conv3-64	1
SPP modules	maxpool 2×2	2
	maxpool 2×2	1
	maxpool 3×3	1
Conv3	conv3-128	1
Conv4	conv3-128	1
SPP modules	maxpool 2×2	2
	maxpool 2×2	1
	maxpool 3×3	1
Conv5	conv3-256	1
Conv6	conv3-256	1
Conv7	conv3-256	1
SPP modules	maxpool 2×2	2
	maxpool 2×2	1
	maxpool 3×3	1
Conv8	conv3-512	1
Conv9	conv3-512	1
Conv10	conv3-512	1

3.2. Multi-Scale Feature Extraction Network (MFENet)

In order to perceive the dramatic changes in the scale of the image and extract the multi-scale features of heads, MFENet is designed as the core of MSDCNet in addition to the SPP module in the front-end network.

In order to improve the accuracy of a deep learning model, researchers improve the network structure constantly. The development of a structure design is mainly along two lines; one is the Inception series (i.e., complexity), from GoogLeNet to Inception V2, V3, V4. The other is the VGG series (i.e., depth), which uses a simple structure to make the network as deep as possible, from VGG to ResNet, DenseNet, etc. The core idea of the Inception

block is to use parallel sub-networks (conv kernels of different sizes) to achieve perception at different scales, and finally fuse them to obtain better representation of the image. But the complexity of multi-column networks is too great; for example, GoogLeNet consists of 9 Inception blocks and 100+ conv layers. Therefore, we cut off the latter half of VGG16 and replaced it with a multi-column subnet (MFENet) to achieve a balance between complexity and accuracy.

MFENet contains four columns with different perception fields to extract different scales of head features, as shown in Table 2. The first layer of each column is a bottleneck layer, 128 1×1 convolutions with a depth of 512. The number of channels of each column is reduced from 512 to 128 due to the bottleneck layer. At the same time, the scale of the feature map extracted by the front-end network is still preserved to cover small targets. One column is only a bottleneck layer and the other columns use stacked 3×3 convolutions in one, two, and three layers to implement perceptive fields in sizes of 3×3, 5×5, and 7×7. Thus, MFENet can extract heads' features in the three scales, large, medium, and small. Multiple small kernels are used consecutively instead of one large kernel, as the latter can significantly reduce the network parameters [14]. For example, to obtain the same 7×7 perceptive field, a 7×7 kernel has 49 parameters and three 3×3 kernels only have 27 parameters. On the other hand, compared to a 7×7 conv layer, three 3×3 conv layers are two layers deeper and have a stronger ability to fit nonlinearity because the nonlinear activation function ReLU is adopted on each conv layer.

Table 2. The architecture of MFENet. The convolutional layers are denoted as "conv (size)-(number of filters)". Padding = 1, stride = 1.

Layers	Kernel
Column1	conv1-128
Column2	conv1-128
	conv3-128
Column3	conv1-128
	conv3-128
	conv3-128
Column4	conv1-128
	conv3-128
	conv3-128
	conv3-128

3.3. Back-End Network

The back-end network is designed to expand the receptive field, extract higher-level semantic information, and then conduct up-sampling to obtain a high-quality density map. After the front-end network and MFENet, to achieve these goals we can continue to deepen the network, using alternate conv and pooling layers. However, pooling will further reduce the feature size and decrease the spatial resolution, which is not conducive to obtaining high-quality density maps. Therefore, we use a dilated convolution instead of the combination of conv and pooling to achieve the above goals [16].

A dilated convolution uses sparse kernels to increase the receptive field [24], as shown in Figure 8. In our model, the size of the dilated convolution kernel is 3×3. When the dilation rate is 2, the receptive field is expanded to 5×5. But the number of parameters is only 9, which is only 36% of that of 5×5 standard convolution. Padding is set to 1, so that overfitting can be prevented to avoid the loss of heads' features while keeping the feature map size unchanged, i.e., without loss of spatial resolution.

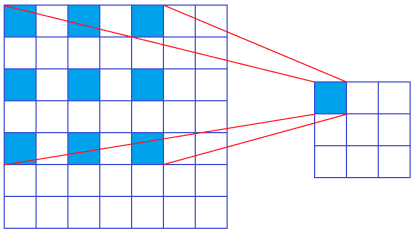


Figure 8. Dilated convolution. The kernel size is 3 × 3, dilation rate = 2.

Four consecutive dilated convolutional layers are deployed in the back-end network, whose architecture is shown in Table 3. After dilated conv layers, a 1 × 1 conv is deployed to output the density map, whose size remains unchanged (W/8 × H/8). In order to obtain a density map of the same size as the input image, up-sampling is conducted by bilinear interpolation to resize the output density map.

Table 3. The architecture of the back-end network. The dilated convolutional layers are denoted as "Dilated conv (size)-(number of filters)". Padding = 1, stride = 1.

Layers	Kernel	Dilation Rate
Dilated conv1	Dilated conv3-256	1
Dilated conv2	Dilated conv3-128	2
Dilated conv3	Dilated conv3-64	2
Dilated conv4	Dilated conv3-32	2
Conv	conv1-1	-

3.4. Discussion on Number of Parameters

In order to evaluate the cost of every component of MSDCNet, the number of parameters is calculated without model compression, as shown in Table 4. The total number of parameters is 9.560 M, and the main cost comes from the front-end network, i.e.,the 10 conv layers of VGG16. Three SPP modules are pooling operations without parameters. Due to the bottleneck layers in every column, MFENet only has 0.360 M parameters, about 3.8% of the total number of parameters. Compared to standard convolution with the same kernel size, dilated convolution in the back-end network does not increase the parameters together with the receptive field expanding.

Table 4. The number of parameters in MSDCNet.

Component		Parameters (M)	
Front-end network			7.633
MFENet	Column1	65,536	0.360
	Column2	81,920	
	Column3	98,304	
	Column4	114,688	
Back-end network			1.567
Total			9.560

4. Experimental Results and Discussions

The experiments were carried out on a Linux, Ubuntu16.04 operating system, and the model was implemented using the Python 3.8 and PyTorch 1.8.0 library. The experimental environment is an Intel (R) Core (TM) i7-8750H CPU @ 2.60 GHz, DDR4 16 GB memory, and a NVDIA GeForce GTX 1650Ti graphics card.

4.1. Datasets and Data Augmentation

Three public datasets, ShanghaiTech, UCF_CC_50, and UCF-QNRF, were used in our experiments. The statistics of samples and annotations in the datasets are shown in Table 5. ShanghaiTech includes samples of different scenes and crowd density levels, which are divided into two parts, Part_A and Part_B. The resolution of the images in Part_B is the same. The UCF_CC_50 dataset contains 50 grayscale images with different resolutions and scenes, including sports events, political processions, religious events, etc. The number of people in the images varies significantly, and some images are of extremely dense scenes. The UCF-QNRF dataset contains high-resolution images of outdoor surroundings with green plantation, buildings, streets, the sky and so on. Therefore, UCF-QNRF is of great significance to the dense crowd counting model against background interference.

Table 5. The statistics of samples and annotations in datasets.

Dataset	Number of Samples			Average Resolution		Annotations		
	Total	Training Set	Test Set	Training Set	Test Set	Total	Ave	Max
ShanghaiTech_Part_A	482	300	182	872 × 598	861 × 574	241,677	501	3139
ShanghaiTech_Part_B	716	400	316	1024 × 768		88,488	123	578
UCF_CC_50	50	-	-	902 × 653		63,974	1279	4633
UCF-QNRF	1535	1201	334	2896 × 2006	2910 × 2038	1,251,642	815	12,865

In ShanghaiTech and UCF-QNRF, samples were divided into a training set and a test set, but the size of three training sets was not large enough. The training sets and UCF_CC_50 were expanded. In order to compare them with other counting models under the same conditions, we used the general data enhancement method [16]. Nine patches of 1/4 size were cropped from an original image; four of them were nonoverlapping areas (top left, bottom left, top right, and bottom right), and the other five patches were randomly cropped. All patches and the original image were then mirrored. Finally, the sample size was expanded to 20 times the original ones. After expansion, UCF_ CC_ 50 contained only 1000 samples, so we randomly cropped the very dense and high-resolution samples to expand the dataset to 7500 samples in total. A fivefold cross-validation was conducted, in which 6000 images were used for training and 1500 images were used for testing.

4.2. Evaluation Metrics

We used the commonly used measures of crowd counting [25], the mean absolute error (MAE) and mean squared error (MSE), which are defined as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|C_i - \hat{C}_i\right|,\ MSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|C_i - \hat{C}_i\right|^2} \tag{1}$$

where N is the number of input images, C_i is the actual number of people in the i th image, and $\hat{C}_i$ is the estimated number of people in the i th image. The lower the MAE is, the higher is counting accuracy; the lower the MSE is, the better the network's robustness is.

4.3. Network Training

The MSDCNet consists of three parts, and we train them separately from easy to difficult as follows:

1. Training the back-end network. The dimensions of MFENet's input and output are same (H/8 × W/8 × 512), so we can remove it first and connect the back end to the front end directly. The front-end network adopts VGG16's pretrained model on

ImageNet. We then freeze the front end and set the back end to trainable. We then obtain a preliminary back end.

2. Fine-tuning the front-end network. We freeze the back end, and train the front end from the pretrained weights of VGG16.
3. Training MFENet column to column. We put MFENet back into our model. At first, we froze the front end, back end, and column 2, 3, and 4 of MFENet, and only set column1 to trainable. After column1 was trained, we trained column 2 starting at part of the weights coming from column1. After four columns had been trained one by one, we trained them together.
4. Fine-tuning the model in order from back end to front end and then to MFENet.

Our model was trained using the Adam optimizer, the ReLU activation function and the Euclidean distance loss. The initial learning rate was set to 0.002 with a decay factor of 0.005 and a momentum value of 0.9.

4.4. Experimental Results on the ShanghaiTech Dataset

After data augmentation, the training set of ShanghaiTech Part_A and Part_ B were expanded to 6000 and 8000 images. Some density maps generated by MSDCNet are shown in Figure 9. The comparison of the counting errors between MSDCNet and other models is shown in Table 6. Besides the accuracy, the parameters and time complexity of every model are compared, too. The time complexity is related to the size of the input image; here, we use the average resolution of training set samples (as shown in Table 5). Only the multiply-accumulate operations (MACs) of conv and FC layers were considered; pooling, BN and ReLU were conventionally ignored.

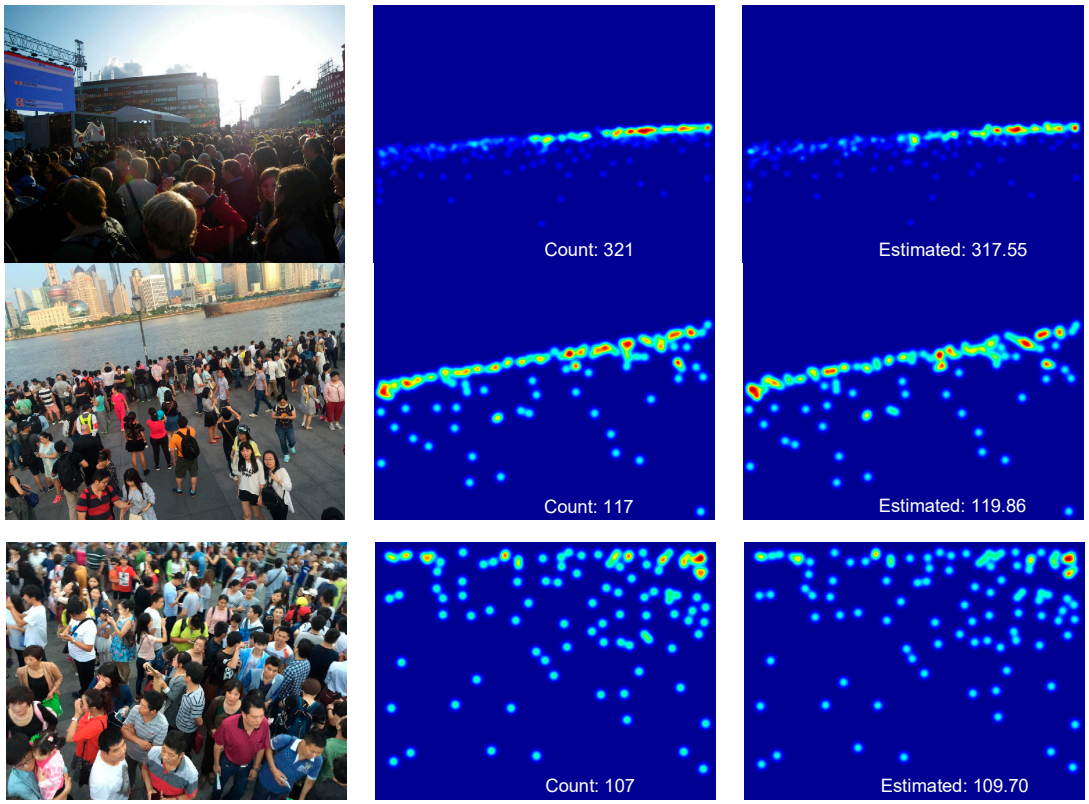


Figure 9. Some density maps generated by MSDCNet from ShanghaiTech dataset. The ground truth density maps are shown in the medium column, and the estimated ones are shown in the right column.

Table 6. Counting errors on ShanghaiTech dataset.

Method	Part_A				Part_B		
	MAE↓	MSE	Parameters (M)	MACs (G)	MAE	MSE	MACs (G)
MCNN [15]	110.2	173.2	0.133	13.989	26.4	41.3	21.097
SwitchCNN [26]	90.4	135.0	15.108	164.182	21.6	33.4	247.610
MSCNN [27]	83.8	127.4	3.084	288.706	17.7	30.2	435.412
CP-CNN [28]	73.6	106.4	17.137	254.311	20.1	30.1	383.537
CSRNet [16]	68.2	115.0	16.259	215.405	10.6	16.0	324.862
SANet [29]	67.0	104.5	1.146	49.334	8.4	13.6	74.403
TEDNet [17]	64.2	109.1	8.863	367.193	8.2	12.8	553.780
DUBNet [30]	64.6	106.8	18.827	52.172	7.7	12.5	78.683
MSDCNet (Ours)	60.9	97.2	9.560	160.804	6.9	11.2	242.516

The experimental results show that our model achieves the best counting accuracy. On the Part_A dataset, the MAE of MSDCNet decreased by 3.7 and the MSE decreased by 9.6 compared to the best-performing model, DUBNet. On the Part_B dataset, the MAE decreased by 0.8 and the MSE decreased by 1.3. The backbone of DUBNet is ResNet50 and its MACs are much lower than ours because a large number of residual blocks is used. This also leads to its parameters being almost twice the size of ours. From the perspective of model volume, the MCNN has the lowest parameters and time complexity. Due to the stacked nine multi-scale encoders with a four-column structure, TEDNet has significant MACs. TEDNet used nine multi-scale encoders with multi-column structures, resulting in a significant increase in the model complexity. Compared with SwitchCNN whose MACs are similar to ours, our parameters were 37% less and the accuracy was greatly improved.

4.5. Experimental Results on the UCF_CC_50 Dataset

The fivefold cross-validation result of our model on the UCF_CC_50 dataset is shown in Figure 10. Some density maps generated by MSDCNet are shown in Figure 11. The comparison of the counting errors between MSDCNet and other models on the UCF_CC_50 dataset is shown in Table 7. The experimental results show that our model achieves excellent counting accuracy on UCF_CC_50. The MAE of MSDCNet decreased by 36.9 and the MSE decreased by 58.0 compared to the DUBNet with the best performance. The average crowd number of samples in UCF_CC_50 was 1279, which is much larger than that of other datasets. Therefore, it can be seen that our model performs well in extremely dense crowds. MSCNN has only about three M parameters because of its concise structure, but has the maximum MACs because it has fewer pooling layers. In our model, both complexity and parameters are taken into account. There are three pooling modules in the front end, so that the subsequent computation can be controlled in a reasonable range.

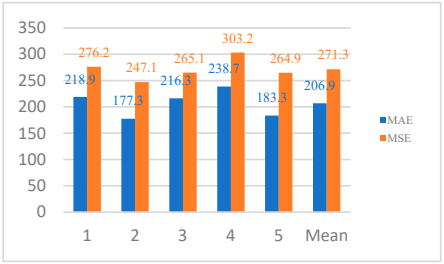


Figure 10. Bar chart of counting errors of the fivefold cross-validation.

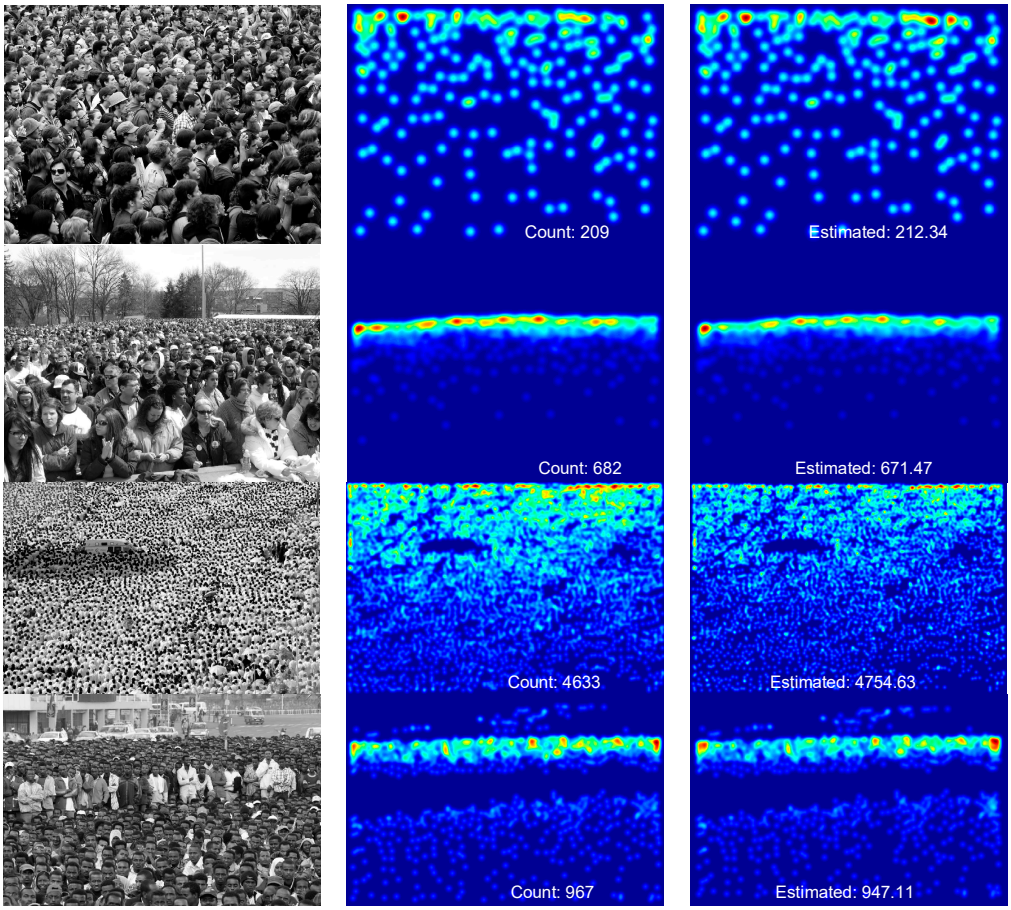


Figure 11. Density maps on UCF_CC_50 test set. The ground truth density maps are shown in the medium column and the estimated ones are shown in the right column.

Table 7. Counting errors on UCF_CC_50 dataset.

Method	MAE↓	MSE	Parameters (M)	MACs (G)
MCNN [15]	377.6	509.1	0.133	15.801
MSCNN [27]	363.7	468.4	3.084	326.106
SwitchCNN [26]	318.1	439.2	15.108	185.450
CP-CNN [28]	295.8	320.9	17.137	287.254
CSRNet [16]	266.1	397.5	16.259	243.308
SANet [29]	258.4	334.9	1.146	55.725
DUBNet [30]	243.8	329.3	18.827	58.930
MSDCNet (ours)	206.9	271.3	9.560	181.635

4.6. Experimental Results on the UCF-QNRF Dataset

The UCF-QNRF dataset contains dense crowd images with multiple scenes, views, and illuminations, and is important for verifying the robustness of counting models. Some density maps generated by MSDCNet from the UCF-QNRF dataset are shown in Figure 12. The comparison of the counting errors between MSDCNet and other models on the UCF-QNRF dataset is shown in Table 8. The experimental results show that our model achieves better counting accuracy on UCF-QNRF than other models. The MAE of MSDCNet decreased by 4.2 and the MSE decreased by 9.6 compared to DUBNet with the best performance. And MSDCNet shows excellent robustness on the UCF-QNRF dataset.

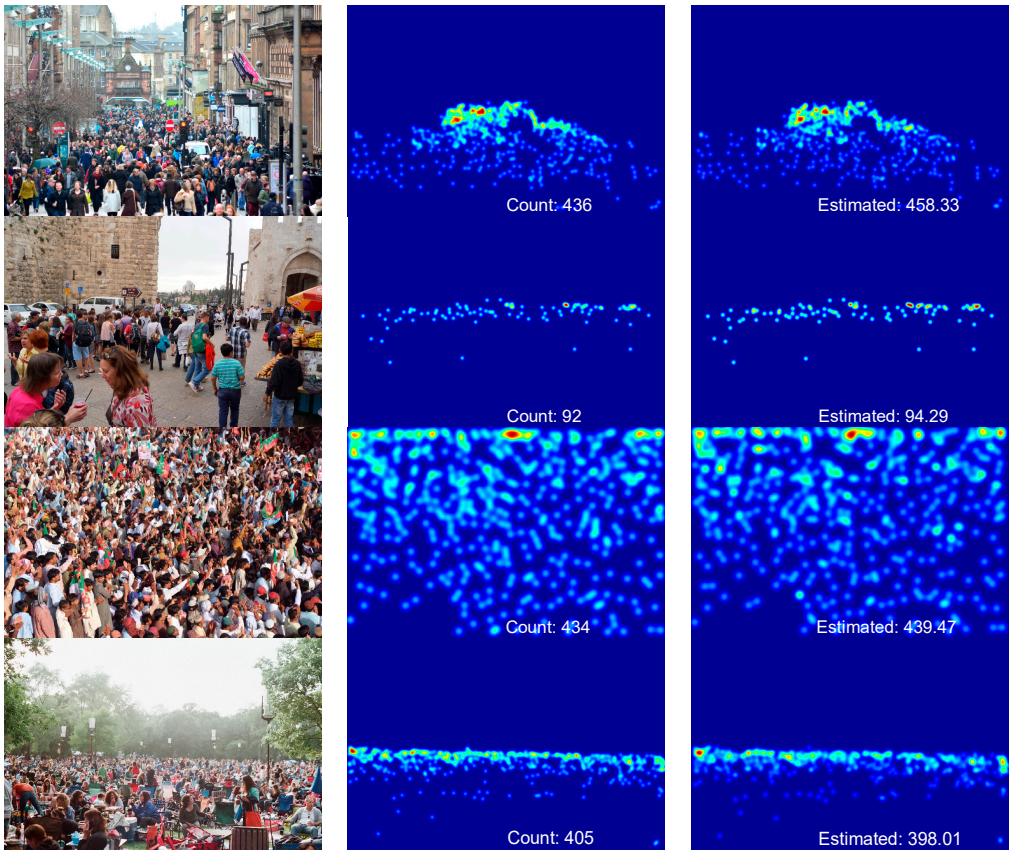


Figure 12. Some density maps generated by MSDCNet from the UCF-QNRF test dataset.

Table 8. Counting errors on the UCF-QNRF dataset.

Method	MAE↓	MSE	Parameters (M)	MACs (G)
SwitchCNN [26]	228	445	15.108	1829.095
RAZ-Net [31]	116	195	24.465	4848.993
TEDNet [17]	113	188	8.863	4090.777
DUBNet [30]	105.6	180.5	18.827	581.231
MSDCNet (ours)	101.4	170.9	9.560	1791.465

4.7. Ablation Experiments

To verify the effectiveness of the proposed MFENet, the SPP module and dilated conv layers, three ablation experiments were conducted. In the first experiment, MFENet in MSDCNet was removed and the model only consisted of a front-end and back-end network. It is denoted as model-A, as shown in Figure 13a. In the second experiment, MFENet was preserved, three SPP modules in the front end were removed, and the original 2×2 max-pooling layers in VGG16 were used. This is denoted as model-B, as shown in Figure 13b. In the third experiment, four dilated conv layers of the back end were replaced with four standard conv layers and a pooling layer. This is denoted as model-C, as shown in Figure 13c. The models are trained and tested on the UCF-QNRF dataset and the experimental results are shown in Table 9.

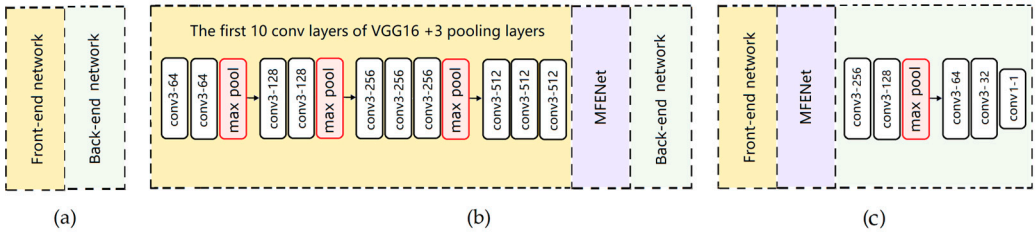


Figure 13. The frameworks of models in ablation experiments. (**a**) model-A: model w/o MFENet; (**b**) model-B: model w/o SPP modules; (**c**) model-C: model w/o dilated conv.

Table 9. Ablation experimental results on UCF-QNRF dataset.

Model	MAE	MSE
model-A (MSDCNet without MFENet)	185.6	268.1
model-B (MSDCNet without SPP module)	163.3	238.7
model-C (MSDCNet without dilated conv)	117.1	185.5
MSDCNet	101.4	170.9

The experimental results indicate that:

1. From the counting errors of the two experimental models and MSDCNet, it can be seen that after removing MFENet and the SPP, the counting accuracy of MSDCNet significantly decreased. Without MFENet, the MAE of model-A increased by 84.2 (about 45% of model-A's MAE), and the MSE increased by 97.2 (about 36% of model-A's MSE). Without the SPP module, the MAE of model-B increased by 61.9 (about 38% of the MAE of model-B), and the MSE increased by 67.8 (about 28% of the MSE of model-B). Without dilated conv layers, the MAE of model-C increased by 15.7 (about 13% of the MAE of model-C), and the MSE increased by 14.6 (about 8% of the MSE of model-C). These fully demonstrate the effectiveness of MFENet, the SPP modules and the dilated conv layers in our model.
2. From the counting errors of the three experimental models, it can be seen that the counting error of model-A is higher than that of model-B and model-C, indicating that MFENet makes a greater contribution to the counting accuracy of the entire network than the SPP modules and dilated conv layers.

5. Conclusions

The proposed crowd counting model solved the problems of uneven crowd distribution and variance of heads scale, achieved high counting accuracy and good robustness on three public datasets. The one-column framework of the backbone simplified the model's structure and kept a certain depth to extract high-level features. The proposed SPP modules in the front end fused the spatial features in pooling and helped the front end to extract features in a different scale without increasing the parameters. By connecting a multi-column subnet (MEFNet) to the one-column backbone, we achieved a compromise result between precision and complexity. Only by increasing the parameters by 3.8% did MEFNet perceive the dramatic changes in the scale of the image and extract the multi-scale features of the heads. The consecutive dilated convolution layers used as the back end prevented overfitting effectively to avoid the loss of head features, expanded the receptive field, and improved the quality of the output density maps.

In future research, we will investigate the impact of the background and image edge, and establish a dense crowd counting model with higher accuracy and better robustness.

Author Contributions: Conceptualization, J.D.; methodology, J.D., Z.Z. and T.W.; software, validation, Z.Z. and T.W.; Writing—original draft, Z.Z.; writing—review and editing, J.D.; supervision, J.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China under Grant NSFC-61671190.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: We thank Jinyang Li and Yushun Zhang for their valuable comments and discussion.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, J.; Liu, J.; Wang, Z. Convolutional Neural Network for Crowd Counting on Metro Platforms. *Symmetry* **2021**, *13*, 703. [CrossRef]
2. de Silva, E.M.K.; Kumarasinghe, P.; Indrajith, K.K.D.A.K.; Pushpakumara, T.V.; Vimukthi, R.D.Y.; de Zoysa, K.; Gunawardana, K.; de Silva, S. Feasibility of using convolutional neural networks for individual-identification of wild Asian elephants. *Mamm. Biol.* **2022**, *102*, 931–941. [CrossRef]
3. Lu, W.G.; Tan, Z. Research on Crowded Trampling Accident Prevention and Disposal in Urban Public Places: The Case of Itaewon Trampling Accident in Korea. *China Emerg. Rescue* **2023**, *1*, 4–10.
4. Gao, G.; Gao, J.; Liu, Q.; Wang, Q.; Wang, Y. CNN-based Density Estimation and Crowd Counting: A Survey. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020.
5. Zhao, R.; Dong, D.; Wang, Y.; Li, C.; Ma, Y.; Enriquez, V.F. Image-Based Crowd Stability Analysis Using Improved Multi-Column Convolutional Neural Network. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 5480–5489. [CrossRef]
6. Chan, A.B.; Liang, Z.S.; Vasconcelos, N. Privacy preserving crowd monitoring: Counting people without people models or tracking. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, USA, 23–28 June 2008.
7. Chan, A.B.; Vasconcelos, N. Bayesian Poisson regression for crowd counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 6 May 2010.
8. Wan, H.L.; Wang, X.M.; Peng, Z.W.; Bai, Z.; Yang, X.; Sun, J. A dense crowd counting algorithm based on a novel multi-scale attention mechanism. *J. Electron. Imaging* **2022**, *44*, 1129–1136.
9. Jiang, N.; Zhou, O.; Yu, F.H. A review of computer vision-based target counting methods. *Laser Optoelectron. Prog.* **2021**, *58*, 43–59.
10. Meng, Y.B.; Chen, X.R.; Liu, G.H.; Xu, S.J. Crowd density estimation method based on multi-feature information fusion. *Laser Optoelectron. Prog.* **2021**, *58*, 276–287.
11. Wang, Y.; Zhang, W.; Huang, D.; Liu, Y.; Zhu, J. Multi-scale features fused network with multi-level supervised path for crowd counting. *Expert Syst. Appl.* **2022**, *59*, 200–212. [CrossRef]
12. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–8 December 2012.
13. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014.
14. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
15. Zhang, Y.; Zhou, D.; Chen, S.; Gao, S.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.
16. Li, Y.; Zhang, X.; Chen, D. CSRNet: Dilated Convolutional Neural Networks for Understanding the Highly Congested Scenes. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1091–1100.
17. Jiang, X.L.; Xiao, Z.H.; Zhang, B.C.; Zhen, X.; Cao, X.; Doermann, D.; Shao, L. Crowd counting and density estimation by trellis encoder-decoder networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020.
18. Yang, Y.; Li, G.; Wu, Z.; Su, L.; Huang, Q.; Sebe, N. Reverse Perspective Network for Perspective-Aware Object Counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020.
19. Zhou, J.T.; Zhang, L.; Du, J.; Peng, X.; Fang, Z.; Xiao, Z.; Zhu, H. Locality-Aware Crowd Counting. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 3602–3613. [CrossRef] [PubMed]
20. Song, Q.; Wang, C.; Jiang, Z.; Wang, Y.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; Wu, Y. Rethinking Counting and Localization in Crowds: A Purely Point-Based Framework. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021.
21. Wan, J.; Liu, Z.; Chan, A.B. A Generalized Loss Function for Crowd Counting and Localization. In Proceedings of the Computer Vision and Pattern Recognition, Online; 2021.
22. Lin, H.; Ma, Z.H.; Ji, R.R. Boosting Crowd Counting via Multifaceted Attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19–24 June 2022.

23. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
24. 24. Fisher, Y.; Koltun, V. Muti-scale context aggregation by dilated convolutions. In Proceedings of the International Conference on Learning Representations (ICLR), San Juan, Puerto Rico, 2–5 May 2016.
25. Liu, W.; Salzmann, M.; Fua, P. Context-aware crowd counting. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
26. Sam, D.B.; Surya, S.; Babu, R.V. Switching convolutional neural network for crowd counting. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
27. Zeng, L.; Xu, X.; Cai, B.; Qiu, S.; Zhang, T. Multi-scale convolutional neural networks for crowd counting. In Proceedings of the IEEE International Conference on Image Proceeding (ICIP), Beijing, China, 17–20 September 2017.
28. Sindagi, V.A.; Patel, V.M. Generating high-quality crowd density maps using contextual pyramid CNNs. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
29. Cao, X.; Wang, Z.; Zhao, Y.; Su, F. Scale Aggregation Network for Accurate and Efficient Crowd Counting. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
30. Oh, M.H.; Olsen, P.; Ramamurthy, K.N. Crowd Counting with Decomposed Uncertainty. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020.
31. Liu, C.; Weng, X.; Mu, Y. Recurrent Attentive Zooming for Joint Crowd Counting and Precise Localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

A Multi-View Face Expression Recognition Method Based on DenseNet and GAN

Jingwei Dong *, Yushun Zhang and Lingye Fan

School of Measurement and Communication Engineering, Harbin University of Science and Technology, Harbin 150080, China; 1605030129@stu.hrbust.edu.cn (Y.Z.); 1605030304@stu.hrbust.edu.cn (L.F.)
* Correspondence: djw@hrbust.edu.cn

Abstract: Facial expression recognition (FER) techniques can be widely used in human-computer interaction, intelligent robots, intelligent monitoring, and other domains. Currently, FER methods based on deep learning have become the mainstream schemes. However, these methods have some problems, such as a large number of parameters, difficulty in being applied to embedded processors, and the fact that recognition accuracy is affected by facial deflection. To solve the problem of a large number of parameters, we propose a DSC-DenseNet model, which improves the standard convolution in DenseNet to depthwise separable convolution (DSC). To solve the problem wherein face deflection affects the recognition effect, we propose a posture normalization model based on GAN: a GAN with two local discriminators (LD-GAN) that strengthen the discriminatory abilities of the expression-related local parts, such as the parts related to the eyes, eyebrows, mouth, and nose. These discriminators improve the model's ability to retain facial expressions and evidently benefits FER. Quantitative and qualitative experimental results on the Fer2013 and KDEF datasets have consistently shown the superiority of our FER method when working with multi-pose face images.

Keywords: facial expression recognition (FER); DenseNet; depthwise separable convolution (DSC); posture normalization; generative adversarial network (GAN)

1. Introduction

With the rapid development of computer technology and artificial intelligence technology, the demand for human–computer interaction is increasingly strong. The realization of the understanding and recognizing of human facial expressions by computers is valuable in the domains of intelligent robotics, intelligent monitoring, virtual reality, medical assisted diagnosis, and so on. Benefiting from the improvement of computer performance, algorithms based on deep learning have become the mainstream scheme of FER.

In the Large Scale Visual Recognition Challenge (ILSVRC) 2012, the AlxNet model [1] based on a convolution neural network (CNN) greatly improved the accuracy of FER. Since then, deeper CNNs have been proposed, such as VGGNet [2], GoogleNet [3], and ResNet [4]. Girshick et al. used Region CNN (R-CNN) to learn facial expression features and achieved good results [5]. Ren et al. used Faster R-CNN to generate high-quality features for FER [6]. Yao et al. used the CReLU activation function in ResNet, increased the network depth while ensuring the recognition rate, and designed a residual block, so that the FER system could capture expression changes by learning features of different scales [7]. However, these deep learning network models were complex and had a large number of parameters, making them unsuitable for embedded computers and mobile devices.

Researchers began to study lightweight CNN networks. In 2017, the densely connected convolution network (DenseNet) was proposed by Huang et al. [8]. DenseNets have several compelling advantages: they alleviate the vanishing gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters. SqueezeNet, a lightweight model proposed by Han, uses a lot of 1×1 convolution kernels

and achieves accuracy similar to that of AlexNet with fewer parameters [9]. Aiming at running on mobile terminals, ShuffleNet, a lightweight model proposed in [10], adds group convolution to the networks, and this makes the model smaller and faster. In the past two years, researchers have proposed many lightweight models that continuously improve FER accuracy [11–15].

The FER algorithms mentioned above focus on the front face image, but since the facial deflection at various angles cannot be avoided in natural environments, the accuracy of FER is more or less lowered [16]. Therefore, pose normalization is performed before FER; i.e., the face is corrected to the front view in the case of deflection.

In order to correct face deflection, early researchers have proposed some 3D modeling methods. However, when the facial deflection angle is too large, the face normalization results of this type of methods are unsatisfactory. In 2014, Goodfellow et al. proposed the generative adversarial network (GAN) [17], which provides a new solution to the problem of missing features caused by face deflection. Up to now, a series of GAN variants have been developed to correct facial deflection. These GAN models for face normalization focus on preserving contour features in the process of synthesizing face images in order to facilitate identification. If the downstream task is FER, the synthesized frontal face does not meet the requirements very well because it does not focus on the preservation of local features related to facial expressions.

In this paper, we present an expression recognition method which combines the DenseNet FER model with the GAN-based posture normalization model. This method solves the problems of the large number of parameters in the FER model and low accuracy in multi-view face normalization. The contributions of this paper are summarized as follows:

1. A lightweight FER model, DSC-DenseNet, which reduces network parameters and computations by improving the standard convolution in DenseNet to DSC, is proposed. When the parameter is 0.16M, the FER rate of this model is 96.7% for frontal face input and 77.3% for profiles without posture normalization.
2. A posture normalization model, GAN, with two local discriminators (LD-GAN) based on the TP-GAN model, is proposed. The encoder–decoder structure implements a two-pathway generator, global pathway, and local pathway. In order to preserve more local features related to facial expressions in generated frontal faces, the discriminator was improved by adding two local discriminators besides the global discriminator to enhance its adversarial capability against the local pathway encoder. The loss functions are also improved to achieve better effects in network training.
3. The effectiveness of this method was verified on three public datasets. The validity of the lightweight FER model was verified on the CK+ and Fer2013 datasets, and the final effect of the combination of the posture normalization model and the FER model was verified on the KDEF dataset. Compared to the methods used in other representative models, this method effectively reduces the number of parameters of our model and has a higher FER rate (92.7%) under the condition of multi-angle deflection.

The remaining parts of this paper are organized as follows. Section 2 describes the previous related work. Section 3 describes the lightweight FER model and the posture normalization model that we propose in detail. Section 4 describes the experimental datasets, results, and related analysis. Section 5 gives conclusions and suggestions for future work.

2. Related Work

2.1. DenseNet

Although various CNN-based FER models have improved recognition rates, the consequent increase in the number of parameters has also resulted in more computational requirements. DenseNet is a model with a narrow network structure, as shown in Figure 1 [8].

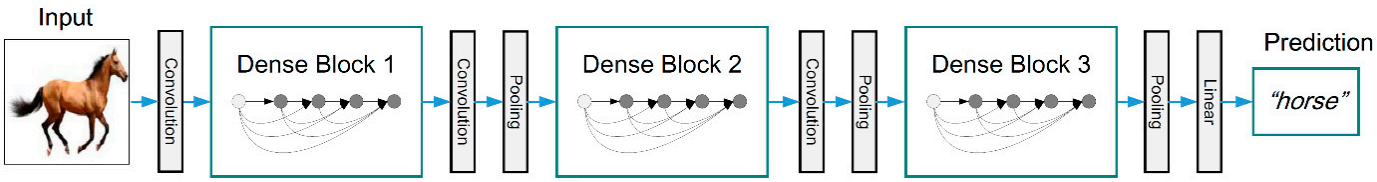


Figure 1. A deep DenseNet with three dense blocks.

DenseNet consists of dense blocks and transition layers, which lie between two adjacent blocks and change feature map sizes via convolution and pooling. The basic idea is that in a dense block, like in ResNet, direct connections from the preceding to the following layers are created. The difference in the dense block is that it establishes a dense connection between all the preceding layers to the followed layer; i.e., each layer takes all preceding feature-maps as input. The feature reuse of DenseNet improves the transmission ability of information throughout the entire network and reduces the number of parameters. To achieve the same accuracy as ResNet, DenseNet only needs about half of ResNet's parameters and half of its FLOPs (floating-point operations).

2.2. Depthwise Separable Convolution (DSC)

Compared to standard convolution, DSC has much lower parameters and computational complexity. Thus, it has been successfully applied to two well-known models, Xception [18] and MobileNet [19], by the Google team. DSC splits the computation of standard convolution into two steps: depthwise convolution, which applies a single convolutional filter per each input channel, and, pointwise convolution, which creates a linear combination of the output of the depthwise convolution. For example, the depthwise conv applies N convolution kernels of size $M \times M \times 1$ to N input channels of size $W \times H$, achieves N feature maps of size $W \times H \times 1$, concatenates N feature maps, and achieves one feature map of size $W \times H \times N$. In other words, the depthwise conv has the same number of channels for input and output feature maps. However, there has been no connection between the different channels in the process so far. Then, the pointwise conv, by applying K standard convolutions of size $1 \times 1 \times N$, solves this problem. It weights the feature map depthwise to generate a feature map of size $W \times H \times K$; i.e., it has the ability to fuse channels. The ratio of DSC to standard convolution is $1/K + 1/M^2$.

2.3. GAN and Its Variants

The GAN is a deep learning model, and its framework is shown in Figure 2a [20]. This framework corresponds to a minimax two-player game. The GAN consists of two models: a generative model *G* that captures the data distribution and a discriminative model *D* that estimates the probability that a sample came from the training data rather than from G. The aim of the training procedure for *G* is to maximize the probability of *D* making a mistake.

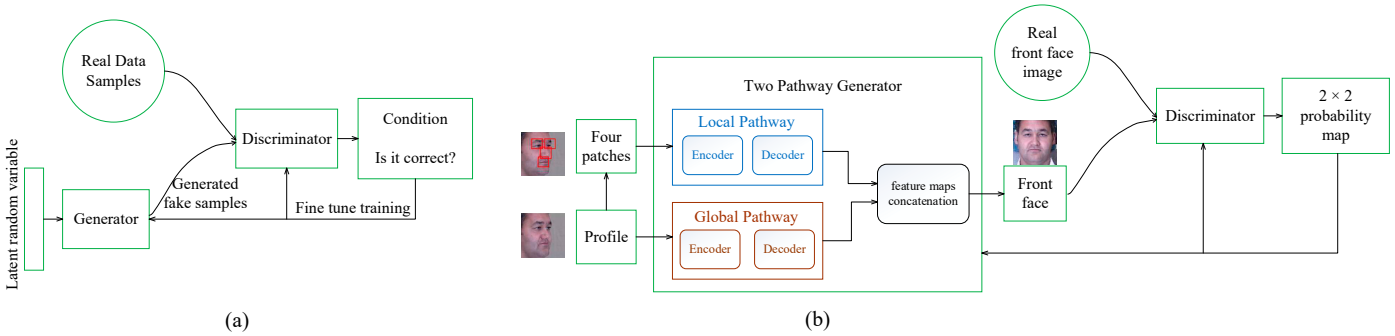


Figure 2. The frameworks of GAN and TP-GAN. (**a**) GAN framework; (**b**) TP-GAN framework.

Huang et al. proposed a two-pathway GAN (TP-GAN) [21] for photorealistic frontal view synthesis by simultaneously perceiving global structures and local details. As shown in Figure 2b, TP-GAN uses two pathways in *G* to perceive global structures and local details simultaneously. Four landmark-located patch networks, in addition to the commonly used global encoder–decoder network, are used to attend to local textures. Then, the positive synthesized image is used for the downstream task: identity recognition.

The training strategy and the loss function are challenging problems facing GANs. Combining a 3D-morphable model with a traditional GAN, FF-GAN [22] solves the problem of GANs being difficult to train by providing shape and appearance priors to guide the training on insufficient samples. Additionally, a new symmetry loss is introduced into the loss function. Similarly providing additional information to assist in training, the disentangled representation learning GAN (DR-GAN) [23] introduces a pose code to *G* and a pose estimation to D. Hu et al. proposed a couple-agent pose-guided GAN (CAPG-GAN) [24]. In the learning process for this network, the pose-guided *G* uses posture information provided by landmark heatmaps of input profile images and ground truth images. The couple-agent *D* essentially consists of two independent discriminators: one for rotation angle discriminating and the other for texture discriminating. Differing from the approaches above, Hardy et al. proposed a learning procedure for distributed GANs, MD-GAN [25], which can be trained over datasets that are spread across multiple servers.

3. Proposed Approach

3.1. Lightweight FER Model: DSC-DenseNet

3.1.1. The Framework of Dense Block

The lightweight FER model in this paper is based on DenseNet's feature reuse strategy, which is shown in Figure 3. In a dense block, the original feature x_0 is inputted into the layer, h_1, and x_1 is the output. The input of the layer h_2 includes not only x_1, the output from the layer h_1, but also the original feature x_0. The input of the layer h_3 includes not only x_2, the output of layer h_2, but also x_1 and x_0.

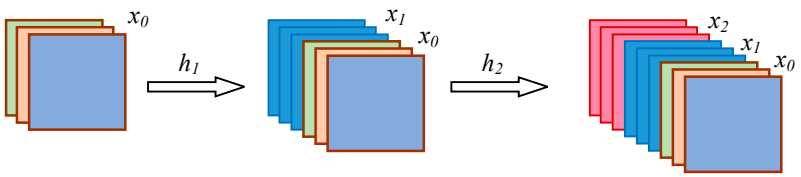


Figure 3. The sketch map of feature reuse in dense block.

To further lighten the FER model, we improve the feature map extracting in the dense block by using DSC instead of standard convolution to simplify the calculation.

The non-linear transformation function H(·) in dense block is defined as batch normalization(BN) + rectified linear unit (ReLU) +3 × 3 convolution. As the number of layers increases, the number of input channels increases dramatically with the number of overlapping feature maps. For this reason, a 1 × 1 conv is used before the 3 × 3 conv to limit the number of input channels. Then H(·) is defined as BN + ReLU + 1 × 1 Conv + BN + ReLU + 3 × 3 Conv. Assume that the input size of layer *i* in a dense block is 48 × 48, the feature maps of all the preceding layers are concatenated with *N* channels, the bottleneck layer reduces the number of channels to 128, and the growth rate *k* (the number of output channels per layer; i.e., the increase number of input channels in the next layer after concatenation) is 32. Then, the operation of layer *i* is as shown in Figure 4.

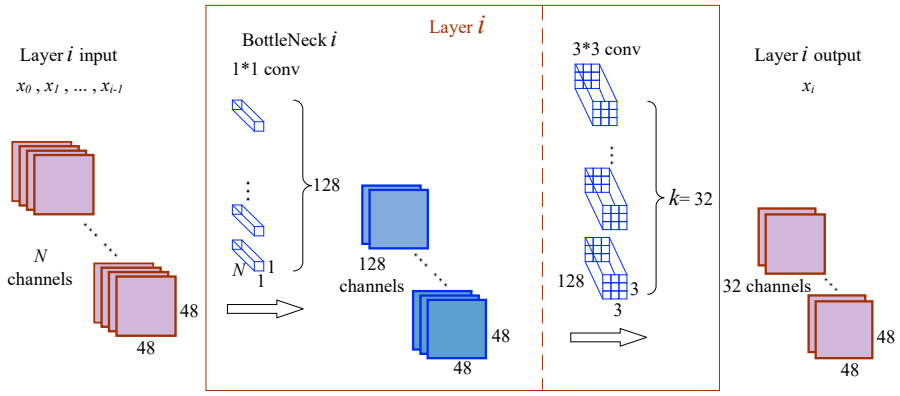


Figure 4. The framework of layer i in a dense block of DenseNet. 1×1 and 3×3 are the kernel sizes.

After replacing standard convolution with DSC, H(·) is defined as BN + ReLU + 3×3 DSC. Then, the operation of layer i is as shown in Figure 5. The number of pointwise conv, i.e., the number of channels of the feature map output by DSC, is the growth rate k of DenseNet.

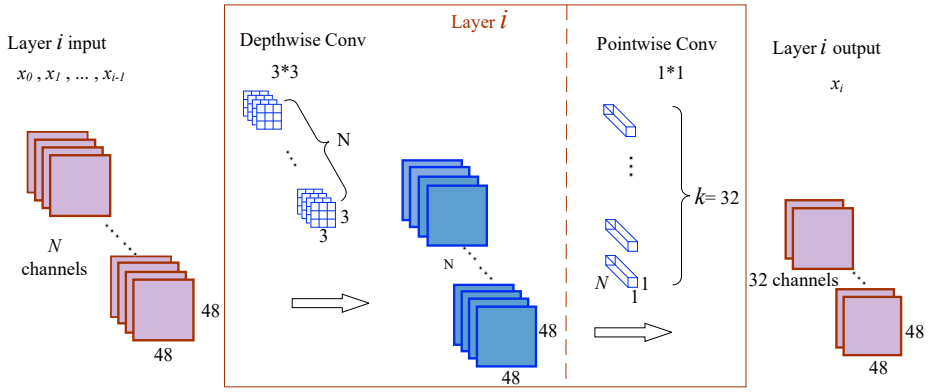


Figure 5. The framework of layer i in a dense block of DSC-DenseNet. 1×1 and 3×3 are the kernel sizes.

3.1.2. The Architecture of DSC-DenseNet

Because of its use of DSC, we refer to this network, which is shown in Figure 6, as DSC-DenseNet. The parameters of its components are given in Table 1.

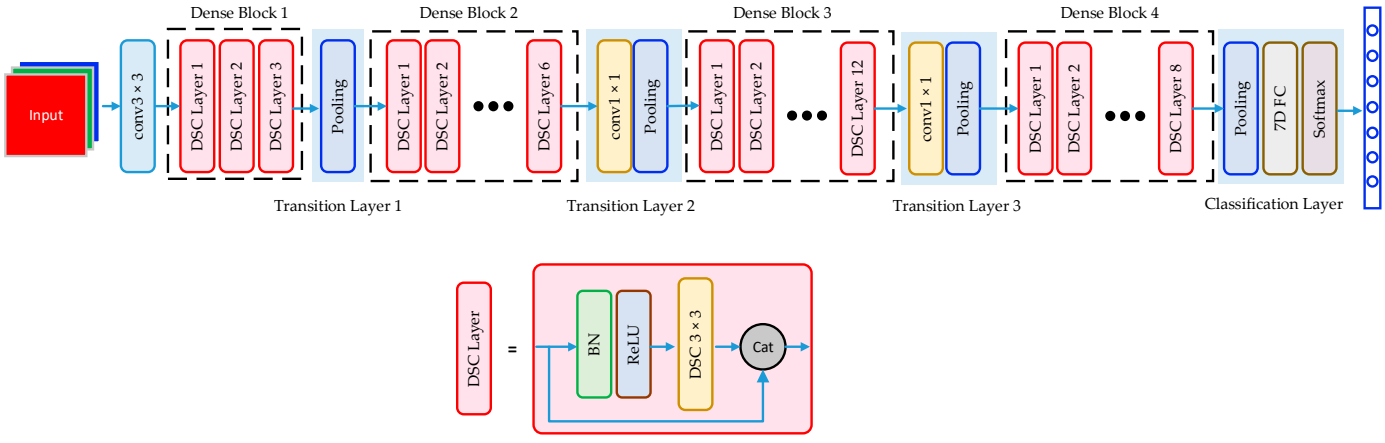


Figure 6. The framework of DSC-DenseNet.

Table 1. The architecture of DSC-DenseNet. The growth rate is k = 32.

Layers	Operator	Output Size	Output Channels
Convolution	3 × 3 conv, stride 1	48 × 48	32
Dense Block1	3 × 3 DSC × 3 layers	48 × 48	32 + 32 × 3 = 128
Transition Layer1	2 × 2 average pool, stride 2	24 × 24	128
Dense Block2	3 × 3 DSC × 6 layers	24 × 24	128 + 32 × 6 = 320
Transition Layer2	1 × 1 × 128 conv	24 × 24	128
	2 × 2 average pool, stride 2	12 × 12	128
Dense Block3	3 × 3 DSC × 12 layers	12 × 12	128 + 32 × 12 = 512
Transition Layer3	1 × 1 × 256 conv	12 × 12	256
	2 × 2 average pool, stride 2	6 × 6	256
Dense Block4	3 × 3 DSC × 8 layers	6 × 6	256 + 32 × 8 = 512
Classification Layer	6 × 6 global average pool	1 × 1	-
	7D fully-connected, softmax	-	-

DSC-DenseNet consists of four dense blocks, three transition layers, and a classification layer. The four dense blocks contain three, six, twelve, and eight DSC layers, respectively. ReLU is used as the activation function. The transition layer uses 2 × 2 average pooling. If too many channels are output by the previous dense block, a 1 × 1 conv is added to reduce the number of channels and thus simplify operations. Examples of this addition include the 1 × 1 × 128 conv in transition layer 2 and the 1 × 1 × 256 conv in transition layer 3. In this situation, the transition layer is BN + 1 × 1 Conv + 2 × 2 average-pooling. Finally, the classification layer realizes the recognition of seven major types of facial expressions through SoftMax multiple classifiers.

Assuming that the input feature map's size and the channel of the convolution layer are $H_i \times W_i \times C_i$, and those of the output feature map are $H_o \times W_o \times C_o$, the convolution kernel of depthwise conv is $H_k \times W_k \times C_i \times 1$ and FLOPs is $H_k \times W_k \times C_i \times H_i \times W_i$, while the convolution kernel of pointwise conv is $1 \times 1 \times C_i \times C_o$ and FLOPs is $C_i \times C_o \times H_i \times W_i$. After summing two computational quantities and dividing them by that of standard convolution, a fraction can be obtained: $1/C_o + 1/H_k W_k$. As the number of feature maps increases, $1/C_o$ can be ignored. The size of the depthwise convolution kernel determines the computational quantity. Due to the use of DSC with 3 × 3 convolution kernels, the computational complexity in a dense block can be reduced to $1/32 + 1/3^2 \approx 14.2\%$. Due to the other layers in DSC-DenseNet, the computational complexity can actually be reduced to about 30%.

3.2. Frontal Face Normalization Model: GAN with Two Local Discriminators (LD-GAN)

Facial pose variations still remain a great challenge for FER models, especially for lightweight ones that sacrifice some accuracy. Therefore, facial pose normalization is a commonly adopted step. Synthesizing a frontal face from a profile image is a highly non-linear transformation.

3.2.1. The Framework of LD-GAN

Based on the idea of a two-pathway generator for TP-GAN, we propose LD-GAN, whose framework is shown in Figure 7. A global pathway is used to process facial contour features and a local pathway is used to process facial expression features in G. Two local feature discriminators are added in D to enhance the adversarial operation between G and D.

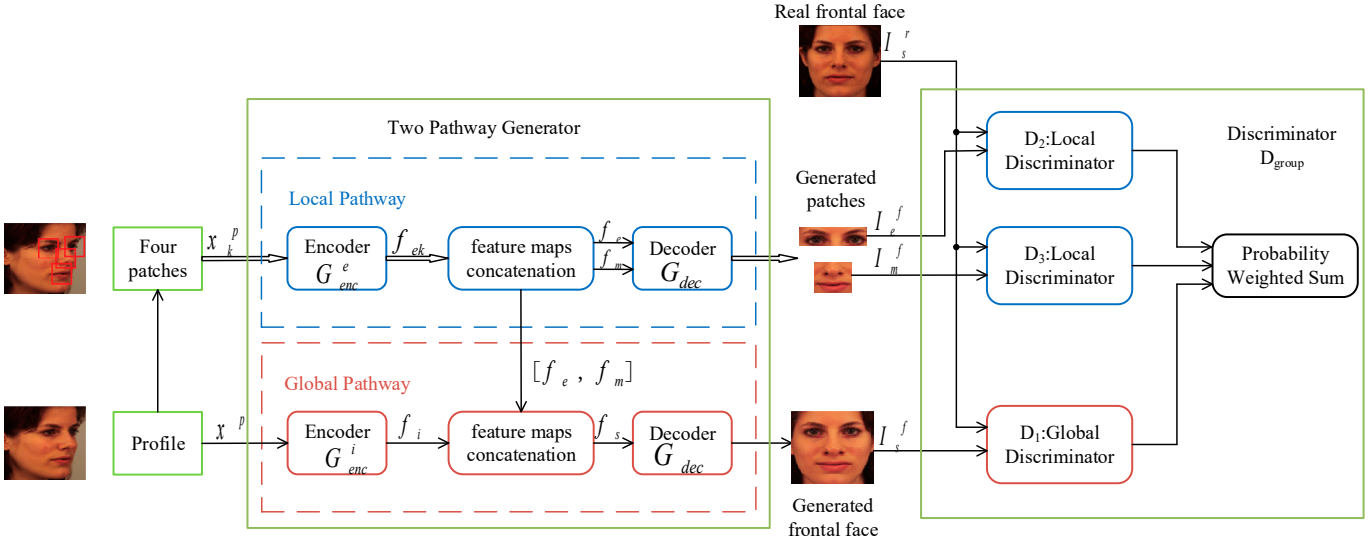


Figure 7. The framework of LD-GAN.

The profile x^p is input into the global pathway and the global encoder G^i_{enc} extracts the global contour features $f_i = G^i_{enc}(x^p)$. After landmarked allocating and cropping, four local patches of eyes, noses, and mouths x^p_k are input into the local pathway, and the local encoder G^e_{enc} extracts the facial expression features $f_{ek} = G^e_{enc}(x_k{}^p)$. The purpose of cropping out expression-related parts is to reduce image noise by eliminating pixels that are less associated with expressions and to force G^e_{enc} to focus on extracting expression features. This way, the generated image can preserve the original expression better. To fuse the information from all pathways, feature maps need to be concatenated together, but only if they have the same spatial resolution. Thus, we first fuse two feature maps from eye patches together, specifically according to their landmarks, and then fuse those of the nose and mouth. At last, we obtain f_s simply by concatenating the global pathway feature map f_i with the two fused feature maps, f_e and f_m. f_s is decoded by G_{dec} and a frontal face I^f_s is generated. In the local pathway, f_e and f_m are decoded by G_{dec} and two frontal patches, eye patch I^f_e and mouth-nose patch I^f_e, are achieved in order to feed D. The reason we use only two patches instead of four is to simplify the framework of D.

Inspired by multi-discriminator strategy [26,27], we propose a D_{group} including a global discriminator D_1 and two local discriminators, eye discriminator D_2 and mouth-nose discriminator D_3. The input of D_{group} includes four images: the real frontal face I^r_s, generated frontal face I^f_s, generated eye patch I^f_e, and generated mouth-nose patch I^f_m. The next step is to combine these four images into three image pairs, $\left(I^r_s, I^f_s\right)$, $\left(I^r_s, I^f_e\right)$, and $\left(I^r_s, I^f_m\right)$, then input them into D_1–D_3 respectively.

3.2.2. The Architecture of LD-GAN

As a key component for extracting features from facial images, the core of the encoder is a CNN. We use Light-CNNs as encoders in both global and local pathways because of their advantages, i.e., having fewer parameters and better robustness [28]. As shown in Table 2, G^i_{enc} and G^e_{enc} have the same architecture of Conv0, Conv1–Conv4, fc1, and fc2. The activation function of Conv1–Conv4 is Maxout. The size of the input RGB image is 128×128 and the size of the final feature map obtained is $1 \times 1 \times 256$. The decoder G_{dec} is a deconvolution neural network. The deconvolution process has no learning ability and can only visualize global contour or local expression features. The network architecture of G_{dec} is shown in Table 3.

Table 2. The architecture of the encoder.

Layer	Kernel/Stride	Output
Conv0	$7 \times 7/1$	$128 \times 128 \times 64$
Conv1	$5 \times 5/2$	$64 \times 64 \times 64$
Conv2	$3 \times 3/2$	$32 \times 32 \times 128$
Conv3	$3 \times 3/2$	$16 \times 16 \times 256$
Conv4	$3 \times 3/2$	$8 \times 8 \times 512$
fc1	-	512
fc2	-	256

Table 3. The architecture of the decoder.

Layers	Kernel	Output
fc reshape	-	$8 \times 8 \times 64$
Deconv0	3×3	$16 \times 16 \times 64$
Deconv1	3×3	$32 \times 32 \times 32$
Deconv2	3×3	$64 \times 64 \times 16$
Deconv3	3×3	$128 \times 128 \times 16$
Deconv4	3×3	$128 \times 128 \times 3$

The architecture of D_1 is shown in Table 4. The size of the generated frontal face input into D_1 is $128 \times 128 \times 3$, and it is changed to $64 \times 64 \times 64$ after the Conv0 layer and to $1 \times 1 \times 1024$ after the Conv1–Conv5 layers. The architectures of D_2 and D_3 are totally same to that of D_1 except for the sizes of input images. The size of the eye patch is 95×20, and that of the mouth-nose patch is 50×75.

Table 4. The architecture of the D_1.

Layer	Kernel/Stride	Output
Conv0	$4 \times 4/2$	$64 \times 64 \times 64$
Conv1	$4 \times 4/2$	$32 \times 32 \times 128$
Conv2	$4 \times 4/2$	$16 \times 16 \times 256$
Conv3	$4 \times 4/2$	$8 \times 8 \times 512$
Conv4	$4 \times 4/2$	$4 \times 4 \times 1024$
Conv5	$4 \times 4/1$	$1 \times 1 \times 1024$

3.2.3. The Loss Function Improved

We have improved the loss function of LD-GAN. Content loss L_{con} is added on the basis of adversarial loss L_{ck}. L_{con} consists of pixel loss L_P and symmetric loss L_S. Then, the loss function of G is:

$$L_G = L_{ck} + L_{con} = L_{ck} + L_P + L_S \tag{1}$$

L_P is the difference in pixels between a generated frontal face and an input profile image:

$$L_p = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left| (I_s^f)_{i,j,k} - x_{i,j,k}^p \right| \tag{2}$$

The smaller L_P is, the closer the quality of the generated image will be to that of the input image, and so, L_P should be minimized. L_S is the Manhattan distance between the left and right sides of the generated frontal face:

$$L_s = \frac{1}{W/2 \times H} \sum_{i=1}^{W/2} \sum_{j=1}^{H} \left| (I_s^f)_{i,j,k} - (I_s^f)_{W-i,j,k} \right| \tag{3}$$

Calculating L_S accelerates the convergence of G. The adversarial loss is:

$$L_{ck} = lnD_k[(I_s^r)_k] + ln[1 - D_k(I_s^f)_k] \tag{4}$$

As D's ability to discriminate between true and false improves via training, G needs to compete against it to minimize its probability of discrimination. Thus, D needs to minimize L_{ck} for G. To sum up, the loss function of LD-GAN generator is:

$$L_G = \beta_1 L_p + \beta_2 L_s + \beta_3 L_{ck} \tag{5}$$

where β_1, β_2, and β_3 are the weights that affect the loss, and can be adjusted during training to achieve the best training results.

The D of LD-GAN does not involve content loss, and its training process only includes adversarial loss, which requires a weighted sum of three adversarial losses. The loss function of D_{group} is:

$$L_{D_{group}} = \omega_1 L_{c1} + \omega_2 L_{c2} + \omega_3 L_{c3} \tag{6}$$

where ω_1, ω_2 , and ω_3 are weighing hyper parameters. D identifies the true or false images generated by G and obtains a probability that the images will be judged as false; thus, D needs to maximize the adversarial loss.

4. Experimental Results and Discussions

The experiments were carried out on the Windows 10 operating system and the recognition methods were implemented using the Python language and PyTorch library. The experimental environment included an Intel(R) Core (TM) i7-10750H CPU @ 2.60 GHz processor, 16 GB memory, and GeForce GTX 1650Ti graphics card.

The effectivenesses of the proposed DSC-DenseNet and LD-GAN were verified on three public datasets. The final results of the combination of DSC-DenseNet and LD-GAN are demonstrated below.

4.1. Datasets

In experiments, we used the following public datasets:

Extended Cohn-Kanade Dataset (CK+) [29]: It is one of the most widely used expression datasets, and was released in 2010. There are 593 sequences in it, and each sequence begins with a neutral expression and proceeds to a peak expression. FER based on a static image often takes the last frames as samples. The eight included expressions are disgust, happiness, surprise, fear, anger, contempt, sadness, and neutral, as shown in Figure 8.

Figure 8. Eight expressions in the CK+ dataset.

Kaggle FER challenge dataset (Fer2013) [30]: All 35,887 examples are 48 × 48 gray images. The training set consists of 28,709 examples, and the validation and test set consists of 3589 examples. When compared to CK+, there are seven of the same expressions in Fer2013, except for contempt, as shown in Figure 9. Moreover, the examples in Fer2013 have deflection at different angles.

Figure 9. Seven expressions in the Fer2013 dataset.

Karolinska directed emotional faces (KDEF) [31]: It includes 4900 GRB images of size 562 × 762 in seven expressions. When compared to Fer2013, every expression in this dataset is represented with five different views, −90°, −45°, 0°, +45°, and +90°, as shown in Figure 10.

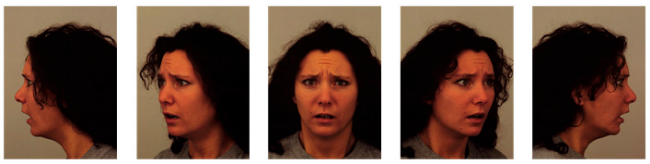


Figure 10. Samples at five views in the KDEF dataset.

4.2. Preprocessing

To ensure the effect of FER, we preprocessed the original images before experiments, including the data on face detection and alignment. We used the multitask cascaded convolutional networks (MTCNN) to hasten face detection and alignment [32,33].

We processed the last three frames in the labeled CK+ dataset expression sequences to 48 × 48 grayscale images. Since there were too few samples of contempt, and also in order to match the seven basic expressions across the datasets, contempt samples from CK+ were excluded from the experiments. Then, we expanded the number of samples to ten times their original number by using common methods for data augmentation such as scale augmentation, changing contrast and changing brightness, and flipping from left to right. To preserve an expression, the central area of the image in question should be maintained. Thus, random cropping or severe rescaling was not adopted by us. We found 10,500 samples from seven expression categories in total. In Fer2013, the numbers of most samples are much larger than those in CK+, but the former has insufficient disgust samples. Thus, we expanded the disgust class, and samples that did not contain faces or had severe facial occlusion were excluded. In KDEF, the sample size of each expression in every view is the same (140). We expanded this to 1400, and so the total number of utilized samples was 49,000.

4.3. Experimental Results of DSC-DenseNet

4.3.1. Experiments for Effectiveness of DSC-DenseNet

The network parameters during training were as follows: the epoch was 150 on CK+ and 250 on Fer2013, batch size was 32, initial learning rate was 0. 01, and learning rate decreased to 50% after each 8 epochs. We used 250 images from each expression category for testing on CK+ and 4000 images from all expression categories for testing on Fer2013. The confusion matrixes of FER percents are shown in Figures 11 and 12.

Figure 11. The confusion matrix of FER on the CK+ dataset.

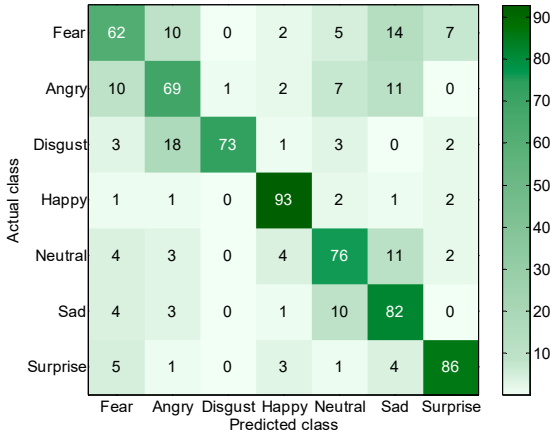


Figure 12. The confusion matrix of FER on the Fer2013 dataset.

The mean of the recognition rates on seven expressions, $R_{expression}$, was used as the final evaluation metric, and it was sometimes abbreviated as 'recognition rate' without causing confusion. It was defined thus:

$$Recognition\ rate = \frac{1}{7}\left(R_{Anger} + R_{Disgust} + \ldots + R_{Neutral}\right) \tag{7}$$

The experimental results showed that:

1. On the CK+ dataset, the recognition rate of our model was 96.7%. The percentages of recognition for the happiness, surprise, and disgust classes were the highest: 100%, 99%, and 98%, respectively. The reason for the good recognition results for the happiness class was that the features of happiness were more obvious than other emotions and thus it was not easily confused with other features. These results showed the same performance as other existing FER methods. The recognition rate for the sadness class was the lowest—92%—and 6% of sad expressions were misclassified as angry. The reason they were easily confused was that they both more or less involved frowning. Neutral and fear expressions were misclassified as sad in 4% and 3% of cases, respectively.
2. On the Fer2013 dataset, the recognition rate of our model was 77.3%. The percentages of recognition for the happiness and surprise classes were 93% and 86%, respectively. The recognition rates for the fear and anger classes were the lowest: 62% and 69%, respectively. The main classes that were confused with the fear class were sadness and anger, while the main classes that were confused with the anger class were fear and sadness.
3. For CK+, the frontal faces dataset, the recognition rate of our model could meet the practical requirements. For Fer2013, the dataset with profile faces, the recognition rate of our model was significantly reduced. Part of the reason for this was that facial occlusion affected recognition to some extent, although severely occluded samples were removed. Another reason was that there were multi-view images in the dataset, and facial deflection significantly affected the effectiveness of FER. This has also been the consensus among researchers, and it also indicates the necessity of studying facial pose normalization models in this paper.

4.3.2. Comparison of DSC-DenseNet with Other Lightweight Models

We performed comparison experiments on Fer2013 to compare our model to the classical lightweight classification models (SqueezeNet, ShuffleNet, ResNet, and MobileNet) and the state-of-art classification models (Separate-loss and RAN). The learning curves of some models are shown in Figure 13. We also compared our model to recently proposed FER models (Light-SE-ResNet and PGC-DenseNet). The FER recognition rate, params, and FLOPs of these models are shown in Table 5 (sorted by recognition rate).

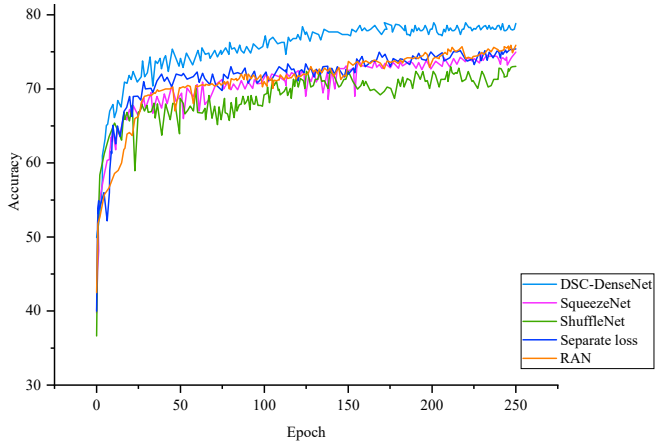


Figure 13. The learning curves of some lightweight models on the Fer2013 dataset.

Table 5. Comparison of lightweight models on the Fer2013 dataset.

Lightweight Model	Params (Million)	FLOPs (Billion)	Recognition Rate (95%CI)
MobileNetV3 (Large) [34]	4.2	0.02	63.9% ± 1.49%
Light-SE-ResNet [35]	5.1	0.06	67.4% ± 1.45%
ResNet18 [4]	11.2	0.09	69.3% ± 1.43%
ResNet50	23.5	0.21	70.1% ± 1.42%
ShuffleNet [10]	1.25	0.04	73.0% ± 1.38%
PGC-DenseNet [36]	0.26	0.12	73.3% ± 1.37%
SqueezeNet [9]	0.47	0.02	75.0% ± 1.34%
Separate-loss [37]	1.1	0.18	75.4% ± 1.33%
RAN [38]	1.2	0.14	76.0% ± 1.32%
DSC-DenseNet (Ours)	0.16	0.16	77.3% ± 1.30%

The experimental results showed that:

1. The FER recognition rate of our model on multi-view faces was 77.3% when the params value was 0.16M. Compared to the models with better accuracy (the lower half of Table 5), such as SqueezeNet and ShuffleNet, our model had higher FLOPs because of its concatenation of feature maps. However, its recognition rate was 2.3% and 4.3% higher than that of SqueezeNet and ShuffleNet, respectively. This could also be seen visually in their learning curves. On the other hand (the upper half of Table 5), it was shown that at the cost of accuracy, speed could be significantly improved. The FLOPs of MobileNetV3 and Light-SE-ResNet were extremely small.
2. Our model had a smaller params value to achieve approximate accuracy. Compared to Separate-loss and RAN, the two models with the closest accuracy to ours, the params value of DSC-DenseNet was equal to only about 15% of their params values, and the FLOPs value of DSC-DenseNet was between theirs. Therefore, our model achieved a practical recognition rate, meaning that the lightweight FER model proposed in this paper achieved a balance between the accuracy and performance requirements of the hardware platform.

4.4. Experimental Results of LD-GAN

4.4.1. Training Strategy

Compared to the true or false discrimination task in G, *D*'s true or false discrimination task was more difficult to train. In order to achieve a dynamic balance between the performances of *G* and D, the update frequency ratio between *G* and *D* was 1:2 in training. At the same time, a small learning rate given to *D* slowed down its convergence and avoided

the ability unbalance between *D* and G, effectively preventing G's loss from increasing continuously and keeping its internal parameters from improving in the desired direction.

During the training process, we could not know when the performance of GAN would be optimal. Too much training may have produced negative effects which could have made the parameters unstable and damaged the effectiveness of the original model. Therefore, it was necessary to store the parameters when the model achieved excellent performance during training. After a certain number of epochs in the learning process, 10 random face images in the test set were compared with the generated faces. If the difference between two images increased abruptly during the process, the training was terminated because this would have indicated that the previously stable parameters had been destroyed.

During the training process, *G* and *D* were optimized using Adam optimizer. the learning rates for *G* and *D* were set to 0.001 and 0.0005. For each epoch, the learning rate was adjusted as an exponential decay with a decay parameter of 0.999. The maximum number of epochs was set to 500 and the batch size was set to 128. In order to generalize the network better, label smoothing was used; i.e., the labels 0 and 1 were replaced by random numbers in the range of 0–0.1 and 0.8–1 when the true or false judgment was made.

4.4.2. Experiments for Effectiveness of LD-GAN

We performed an experiment on the KDEF dataset, and the loss curve for this is shown in Figure 14. From the trend of the loss curve, it is clear that the training strategy used on LD-GAN was effective. The result of the posture normalization is shown in Figure 15.

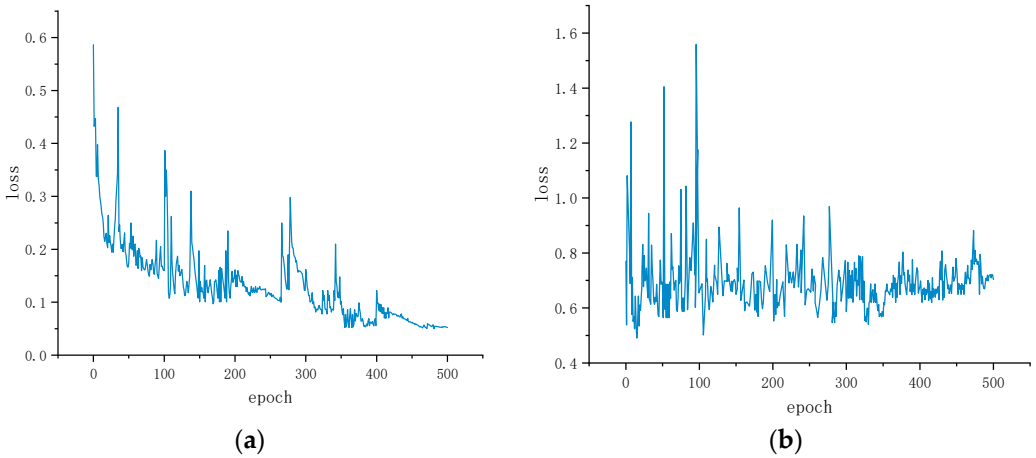


Figure 14. Training loss curves on the KDEF dataset. (**a**) Content loss L_{con}; (**b**) adversarial loss L_{ck}.

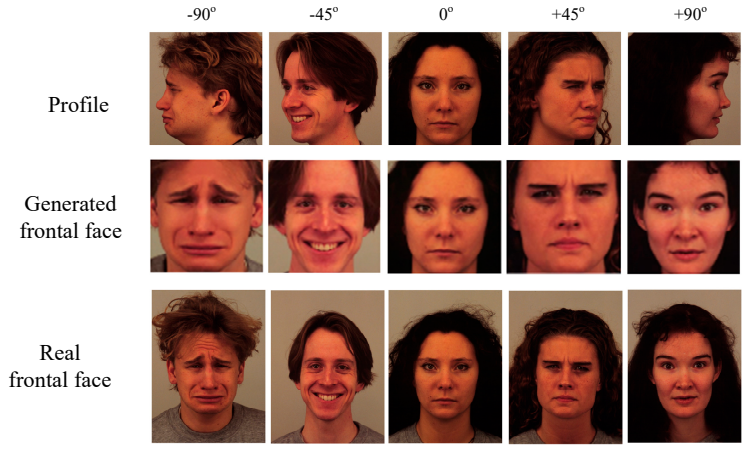


Figure 15. The result of the posture normalization on KDEF dataset.

In the study, the frontal images generated by LD-GAN were recognized by DSC-DenseNet. The mean FER recognition rate of each expression, at all views, was calculated, and the confusion matrix of seven facial expressions was derived, as is shown in Figure 16.

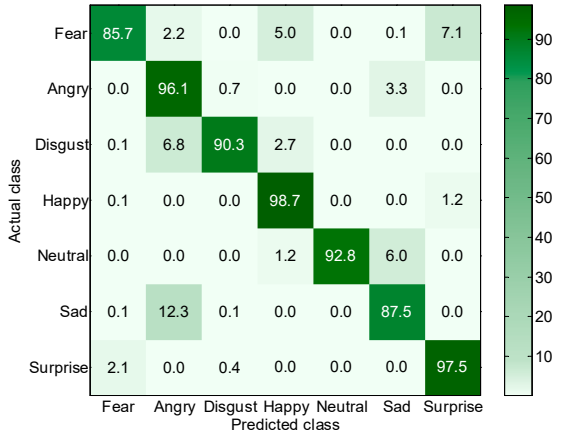


Figure 16. The confusion matrix of recognition rates.

The experimental results showed that:

1. The recognition rate for happy expressions was the highest: 98.7%. The rates of recognition for the surprise and anger classes were the second highest; these amounted to more than 96%. The recognition rates for the fear and sadness classes were lower: 85.7% and 87.5%, respectively. The mean recognition rate for the seven expressions was 92.7%.
2. When compared to the results without pose normalization (Figure 12), the method proposed significantly improved the recognition rate, and could also reduce the misclassification rate between different expressions to a lower level, thus meeting the needs of practical application.

4.4.3. Comparison of Our Model with Others

We performed comparison experiments on the KDEF dataset. The profiles in the $\pm 45^\circ$ and $\pm 90^\circ$ views were corrected to frontal faces by three GAN variants (FF-GAN, TP-GAN, and DR-GAN) and our model, and then FER was performed using DSC-DenseNet. We used 5-fold cross validation, and the results of this are shown in Table 6.

Table 6. Comparison of FER recognition rates, with 95% Cis, of the four models on the KDEF dataset.

Model \ View	$\pm 90^\circ$	$\pm 45^\circ$
FF-GAN [22]	83.5% ± 1.16%	89.6% ± 0.96%
TP-GAN [21]	83.6% ± 1.16%	88.9% ± 0.98%
DR-GAN [23]	85.4% ± 1.11%	90.7% ± 0.91%
LD-GAN (ours)	88.8% ± 0.99%	93.2% ± 0.79%

The experimental results showed that the FER rate of our method was 5.3%, 5.2%, and 3.4% higher than that of FF-GAN, TP-GAN, and DR-GAN, respectively, at $\pm 90^\circ$ views. At $\pm 45^\circ$ views, the FER rate of our method was 3.6%, 4.3%, and 2.5% higher than that of each of the three models, respectively.

4.4.4. Ablation Study

In order to generate facial images with more expression features, we used two pathways in *G* to generate local features that were closely related to the selected expressions. Two Local discriminators were added to *D* to preserve the local details and to improve

the accuracy of expression classification. To verify the validity of our model, an ablation experiment was performed on the KDEF dataset.

To compare our model to LD-GAN, two Local discriminators were removed from the model. The normalization effects of the two models are illustrated in Figure 17. Figure 17a shows a sad-faced man's profile at a -90° view and a neutral-faced woman's profile at a $+45^\circ$ view. Compared to the real frontal faces in Figure 17d, the man's mouth does not show the drop it should have, and the woman's mouth in Figure 17b does show an excessive rise in the images generated by GAN without the Local discriminators. These errors increase the probability that the man's sad expression would be misclassified as neutral and the woman's neutral expression would be misclassified as happy in subsequent FER steps. In Figure 17c, which shows the faces generated by LD-GAN, the corners of the subjects' mouths are closer to those in the real frontal faces.

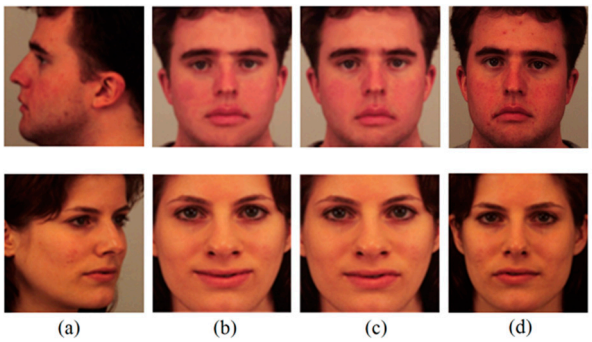


Figure 17. Comparison of face normalizations between two models. (**a**) Input profiles; (**b**) faces generated by GAN without Local discriminator; (**c**) faces generated by LD-GAN; (**d**) real frontal faces.

5. Conclusions

The lightweight DSC-DenseNet FER model proposed in this paper minimizes the number of parameters and the complexity of computation and achieves a useful FER rate. The LD-GAN face posture normalization model proposed improves the ability to reserve local features related to expression and can generate a face that is more conducive to FER. Experiments on multiple datasets show that the recognition rate for faces at multi-view is higher than 92% when combining LD-GAN and DSC-DenseNet.

In future research, we will investigate the impact of different lighting environments and local occlusion on our model and establish a lightweight FER method for natural scenes.

Author Contributions: Conceptualization, J.D. and Y.Z.; methodology, Y.Z., J.D. and L.F.; software, validation, Y.Z. and L.F.; writing—review and editing, J.D. and Y.Z.; supervision, J.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China under Grant NSFC-61671190.

Data Availability Statement: The code and data presented in this study are available at https://gitee.com/djw_Hrbust/fer.git (accessed on 1 June 2023).

Acknowledgments: We thank for Jinyang Li and Ziqi Zhao for their valuable comments and discussion.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
2. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-scale Image Recognition. *Comput. Sci.* **2014**, *56*, 1–14.
3. Szegedy, C.; Liu, W.; Jia, Y. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.

4. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
5. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
6. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
7. Yao, A.; Cai, D.; Hu, P. HoloNet: Towards Robust Emotion Recognition in the Wild. In Proceedings of the 18th ACM International Conference on Multimodal Interaction, Tokyo, Japan, 12–16 November 2016.
8. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
9. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level Accuracy with 50× Fewer Parameters and <0.5 MB Model Size. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
10. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. Shufflenet V2: Practical Guidelines for Efficient CNN Architecture Design. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
11. Shen, H.; Meng, Q.; Liu, Y. Facial Expression Recognition by Merging Multilayer Features of Lightweight Convolution Networks. *Laser Optoelectron. Prog.* **2021**, *58*, 148–155.
12. Li, C.; Lu, Y. Facial expression recognition based on depthwise separable convolution. *Comput. Eng. Des.* **2021**, *42*, 1448–1454.
13. Gao, J.; Cai, Y.; He, Z. TP-FER: Facial expression recognition method of tri-path networks based on optimal convolutional neural network. *Appl. Res. Comput.* **2021**, *38*, 7.
14. Liang, H.; Lei, Y. Expression Recognition with Separable Convolution Channel Enhancement Features. *Comput. Eng. Appl.* **2022**, *58*, 184–192.
15. Zhang, P.; Kong, W.; Teng, J. Facial Expression Recognition Based on Multi-scale Feature Attention Mechanism. *Comput. Eng. Appl.* **2022**, *58*, 182–189.
16. Han, Z. Facial Expression Recognition under Various Facial Postures. Master's Thesis, Soochow University, Suzhou, China, 2020.
17. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 8–11 December 2014.
18. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
19. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
20. Generative Adversarial Network (GAN). Available online: https://www.geeksforgeeks.org/generative-adversarial-network-gan/ (accessed on 31 March 2023).
21. Huang, R.; Zhang, S.; Li, T.; He, R. Beyond Face Rotation: Global and Local Perception GAN for Photorealistic and Identity Preserving Frontal View Synthesis. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
22. Yin, X.; Yu, X.; Sohn, K.; Liu, X.; Chandraker, M. Towards Large-Pose Face Frontalization in the Wild. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
23. Tran, L.; Yin, X.; Liu, X. Disentangled Representation Learning GAN for Pose-invariant Face Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
24. Hu, Y.; Wu, X.; Yu, B.; He, R.; Sun, Z. Pose-guided Photorealistic Face Rotation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
25. Hardy, C.; Le Merrer, E.; Sericola, B. MD-GAN: Multi-Discriminator Generative Adversarial Networks for Distributed Datasets. In Proceedings of the IEEE International Parallel and Distributed Processing Symposium, Rio de Janeiro, Brazil, 20–24 May 2019.
26. Lin, H.; Ma, H.; Gong, W.; Wang, C. Non-frontal Face Recognition Method with a Side-face-correction Generative Adversarial Networks. In Proceedings of the 3rd International Conference on Computer Vision, Image and Deep Learning, Changchun, China, 20–22 May 2022.
27. Iizuka, S.; Simo-serra, E.; Ishikawa, H. Globally ang Locally Consistent Image Completion. *ACM Trans. Graph.* **2017**, *36*, 1–14. [CrossRef]
28. Sharma, A.K.; Foroosh, H. Slim-CNN: A Light-Weight CNN for Face Attribute Prediction. In Proceedings of the IEEE International Conference on Automatic Face & Gesture Recognition, Buenos Aires, Argentina, 18–22 May 2020.
29. Extended Cohn-Kanade Dataset. Available online: http://www.pitt.edu/~emotion/ck-spread.htm (accessed on 19 April 2023).
30. Kaggle FER Challenge Dataset. Available online: https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/overview (accessed on 19 April 2023).
31. Karolinska Directed Emotional Faces. Available online: http://www.emotionlab.se/kdef (accessed on 19 April 2023).
32. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Process. Lett.* **2017**, *23*, 1499–1503. [CrossRef]

33. Xia, H.; Li, C. Face Recognition and Application of Film and Television Actors Based on Dlib. In Proceedings of the International Congress on Image and Signal Processing, Biomedical Engineering and Informatics, Suzhou, China, 19–21 October 2019.
34. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
35. Xie, Y. Research and Application of Facial Expression Recognition Method Based on Deep Learning Architecture. Master's Thesis, Wuhan Institute of Technology, Wuhan, China, 25 May 2022.
36. Yang, H. Facial Expression Recognition Method Based on Lightweight Convolutional Neural Network. Master's Thesis, Beijing University of Civil Engineering and Architecture, Beijing, China, 30 May 2020.
37. Li, Y.; Lu, Y.; Li, J.; Lu, G. Separate Loss for Basic and Compound Facial Expression Recognition in the Wild. In Proceedings of the Eleventh Asian Conference on Machine Learning, WINC AICHI, Nagoya, Japan, 17–19 November 2019.
38. Wang, K.; Peng, X.; Yang, J.; Meng, D.; Qiao, Y. Region Attention Networks for Pose and Occlusion Robust Facial Expression Recognition. *IEEE Trans Image Process.* **2020**, *29*, 4057–4069. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Speech Emotion Recognition Based on Deep Residual Shrinkage Network

Tian Han [1,2], Zhu Zhang [1,2,*], Mingyuan Ren [1], Changchun Dong [1], Xiaolin Jiang [1] and Quansheng Zhuang [2]

[1] Department of Artificial Intelligence, Jinhua Advanced Research Institute, Jinhua 321013, China; htopen@foxmail.com (T.H.); rmy2000@126.com (M.R.); hitdongcc@163.com (C.D.); jlynner@163.com (X.J.)
[2] School of Measurement and Communication Engineering, Harbin University of Science and Technology, Harbin 150080, China; izqshust@tom.com
* Correspondence: zhuzhang.zz@foxmail.com

Abstract: Speech emotion recognition (SER) technology is significant for human–computer interaction, and this paper studies the features and modeling of SER. Mel-spectrogram is introduced and utilized as the feature of speech, and the theory and extraction process of mel-spectrogram are presented in detail. A deep residual shrinkage network with bi-directional gated recurrent unit (DRSN-BiGRU) is proposed in this paper, which is composed of convolution network, residual shrinkage network, bi-directional recurrent unit, and fully-connected network. Through the self-attention mechanism, DRSN-BiGRU can automatically ignore noisy information and improve the ability to learn effective features. Network optimization, verification experiment is carried out in three emotional datasets (CASIA, IEMOCAP, and MELD), and the accuracy of DRSN-BiGRU are 86.03%, 86.07%, and 70.57%, respectively. The results are also analyzed and compared with DCNN-LSTM, CNN-BiLSTM, and DRN-BiGRU, which verified the superior performance of DRSN-BiGRU.

Keywords: speech emotion recognition; mel-spectrogram; DRSN; BiGRU

1. Introduction

Communication through language is the most fundamental and significant aspect of human interaction. Emotional communication in particular facilitates a deeper, more nuanced understanding between individuals, leading to more precise and genuine expressions of feelings. Regrettably, this emotional connection is often absent in human–computer interaction, limiting the level of engagement and satisfaction in such scenarios. Consequently, it becomes critical to imbue computers with the capacity to comprehend human emotions conveyed through vocal expression.

The origins of research on emotion recognition of speech date back to the 1980s [1,2]. The conventional method for emotion recognition of speech involves the classification of speech emotions, which commonly employs classification methods, such as KNN [3], SVM [4,5], HMM [6], GMM [7], Decision Tree [8], and ANN [9] to name a few. These classification methods employ feature extraction of speech, with commonly used features falling into three categories, prosodic features, sound quality features, and spectral features. Prosodic features include pitch, short-term energy, and duration [10,11], while spectral features include MFCC, LPCC, LFPC, GFCC, Formants, etc. [12]. Sound quality features include jitter, shimmer, and HNR [13–15]. Research on speech emotion recognition has traditionally focused on the use of these features or optimization of traditional models. In recent years, deep learning has emerged as a prominent alternative to the traditional models, showcasing superior performance across various fields, including the realm of speech emotion recognition. Recent applications of DNN, RNN, CNN, LSTM, and other network models have reaped fruitful outputs in voice emotion recognition [16,17].

Through speech emotion recognition technology, people can more conveniently interact with computers and achieve more intelligent services and management. At the

same time, voice emotion recognition technology can also help people better understand their emotional state, and timely detect and solve psychological problems. Therefore, the research and application of speech emotion recognition technology is of great significance for improving people's quality of life and work efficiency. At present, speech emotion recognition technology is not yet perfect, and there are two main areas that need to be improved which are searching for robust and universal emotional features of speech, and constructing strong models with high recognition accuracy. In this paper, a speech emotion recognition method is proposed which is composed of Mel-spectrogram feature extraction and Deep Residual Shrinkage Network with Bidirectional Gated Recurrent Unit (DRSN-BiGRU). In Section 3.1, the Mel-spectrogram is introduced in detail and adopted as a feature for speech emotion recognition, as it contains both time-domain and frequency-domain information of speech, and can be used as input for convolution layers. In Section 3.2, DRSN is proposed in detail which is used to recognize emotions from Mel-spectrogram. In this part, residual shrinkage building unit, soft thresholding and self-attention mechanism are introduced and analyzed, which are key components of the network. In Section 3.3, BiGRU is proposed as the last part of the model, which allows the network to not only obtain information from forward propagation, but also utilize information in the reverse direction, making more important features fully utilized. In order to verify the speech emotion recognition method, experiments on three emotional speech datasets (CASIA dataset, IEMOCAP dataset, and MELD dataset) are carried out and results are presented in Section 4. In Section 5, the speech emotion recognition method is discussed and compared with three other methods.

2. Related Work

Speech emotion recognition methods based on speech acoustic features typically extract some manually designed features from speech signals and feed these features into classifiers to complete recognition tasks. In 2018, Zhang used AlexNet designed for the ImageNet LSVRC-2010 to recognize emotion from speech, and in the research EMO-DB was used to verify the performance of the network. The accuracy of four emotions (angry, sad, happy, and neutral) are over 80%, which was about 20% higher than SVM. Then the same experiments were carried out in three other databases (RML, Enterface05, and Baum-1s), and AlexNet all got better results than SVM [18]. In 2019, Sun introduced a novel approach to recognize speech emotion utilizing a DNN-decision tree-SVM model. DNNs extract bottleneck features that are then employed to train each SVM in the decision tree. The findings of the experiment indicate that the proposed technique yields an average emotion recognition rate 6.25% higher than the traditional SVM classification and 2.91% better than the DNN-SVM classification [19]. In 2019, Zhao proposed a novel approach to emotion recognition from speech using two CNN and LSTM networks, one 1D CNN LSTM network and one 2D CNN LSTM network. These networks were specifically designed to learn both local and global emotional features from speech and log-Mel spectrogram data. The research found that the hybrid model could leverage the strengths of both types of networks while overcoming the limitations of both [17]. In 2020, Huang analyzed the differences in speech signal features of different emotional categories and ranked the contribution of features. The study found that the fundamental frequency (F0), sound intensity, and duration have a greater contribution to emotional categories [20]. In 2020, Atmaja used an 88-dimensional Ge MAPS feature set as the input feature of the Long Short-Term Memory (LSTM) network, and discussed the impact of different loss functions on the performance of dimensional emotion recognition [21]. In 2021, Cai proposed a multi task learning framework that simultaneously performs automatic speech recognition and speech emotion classification tasks. This method is an end-to-end deep neural network model for speech emotion recognition based on wav2vec [22]. In 2020, Yeh used end-to-end ASR to extract ASR-based representations for speech emotion recognition, and designed a decomposition domain adaptation method on a pre-trained ASR model to improve speech recognition rate and recognition accuracy of the target emotion corpus [23]. In 2020, Bakhshi used raw speech time-domain signals and frequency-domain information as

inputs to the deep Conv-RNN network, effectively extracting emotional representations of speech signals and achieving end-to-end dimensional emotion recognition [24]. In 2020, Sun used residual convolution neural networks to capture emotional information from raw speech signals and classify them. In addition, this method also incorporates speaker gender information to further improve recognition rate [25]. In 2020, a novel framework was introduced by Sajjad for detecting emotions from speech. The proposed methodology utilized key sequence segment selection based on a radial-based function network (RBFN) similarity measurement in clusters to arrive at the selected sequence. This sequence was converted into a spectrogram through the application of the STFT algorithm and subsequently fed into a CNN for feature extraction. To ensure precise recognition performance, normalization of CNN features was undertaken prior to their use in training a deep bi-directional long short-term memory (BiLSTM) model to learn temporal information for recognizing the final state of emotion [26]. In 2020, Issa introduced a novel architecture that utilizes a combination of Mel-frequency cepstral coefficients, chromagram, Mel-scale spectrogram, Tonnetz representation, and spectral contrast features to extract information from sound files for emotion recognition. The one-dimensional Convolution Neural Network is employed to classify the features, and an incremental method is employed to optimize the model for enhanced classification accuracy [27].

In recent studies, it has been further demonstrated that introducing attention mechanisms and using bidirectional long short-term memory networks can further improve recognition accuracy. In 2021, Wang used a multimodal transformer with shared weights to capture dependencies between modalities, which also achieved good results. Finally, the success of the pre-trained model in speech recognition tasks has led to its increasing attention in research on speech emotion recognition [28]. In 2022, Zou utilized the Wav2Vec pre-training model to extract deep features of speech and combined them with traditional acoustic features for emotion recognition [29]. The review of the above research is summarized in Table 1.

Currently, speech emotion recognition mainly relies on data-driven deep learning models. However, there are essential differences and inexplicability between the emotional representation space extracted by deep learning models and the human understandable space. The inexplicability of these deep learning models poses challenges for establishing secure and reliable speech emotion recognition tasks. In addition, emotion recognition involves knowledge from multiple disciplines, such as perception, cognition, and psychology. How to integrate these related knowledge into deep model algorithms is a major challenge in speech emotion recognition technology. Therefore, constructing a combination of knowledge driven and data driven speech emotion recognition will be another direction for the development of speech emotion recognition.

Table 1. Literature survey on techniques for speech emotion recognition.

Year and Author	Models	Datasets	Contributions
2018 Zhang [18]	DCNN	EMO-DB RML eNTERFACE BAUM	Application of AlexNet in SER
2019 Sun [19]	Decision Tree DNN SVM	CASEC	Bottlenect features extracted by DNNs
2019 Zhao [17]	LSTM CNN	EMO-DB IEMOCAP	1D and 2D DNN LSTM network
2020 Sun [25]	R-CNN	FAU eNTERFACE	End-to-end SER with RAW data

Table 1. *Cont.*

Year and Author	Models	Datasets	Contributions
2020 Huang [20]	NetVALD	IEMOCAP	Rank different features
2020 Atmaja [21]	LSTM	IEMOCAP	88-dimensional features
2021 Cai [22]	Wav2Vec	IEMOCAP	Multi task learning framework
2020 Yeh [23]	ASR model	IEMOCAP LibriSpeech	ASR based representation of speech
2020 Bakhshi [24]	DNN	RECOLA	End-to-end model of predicting continuous emotions
2020 Sajjad [26]	BiLSTM	EMO-DB IEMOCAP RAVDESS	Key sequence segment selection based on RBFN
2020 Issa [27]	CNN	EMO-DB IEMOCAP RAVDESS	A framework of features combination
2021 Wang [28]	Transformer	IEMOCAP	Multi modal transformer with sharing weights
2022 Zou [29]	CNN BiLSTM Wav2Vec2	IEMOCAP	Multi-level acoustic information with co-attention module

3. Materials and Methods

3.1. Mel-Spectrogram of Speech

Speech is a signal within the time domain that employs traditional features for recognition and classification. These features include short-term energy, zero-crossing rate, pitch, formant, MFCC, and LPCC, either in frequency or time domain. In order to produce superior results, it is necessary for the machine learning model to integrate both domain features and establish an appropriate trade-off. This presents significant challenges for model and feature selection. With the help of deep learning algorithms and their more intricate network structures, such as highlighting key features during the training process and filtering redundant information, more precise results can be produced. The spectrogram represents the speech signal in two dimensions of spectral information, which illustrates characteristics in both frequency and time domains. The Mel-filter function converts the energy spectrum to the Mel-frequency, which better imitates the human ear principle. By blending the spectrogram and Mel-filter advantages, this paper introduced the Mel-spectrogram feature for speech emotion recognition. Figure 1 depicts the feature extraction flow chart of Mel-spectrogram.

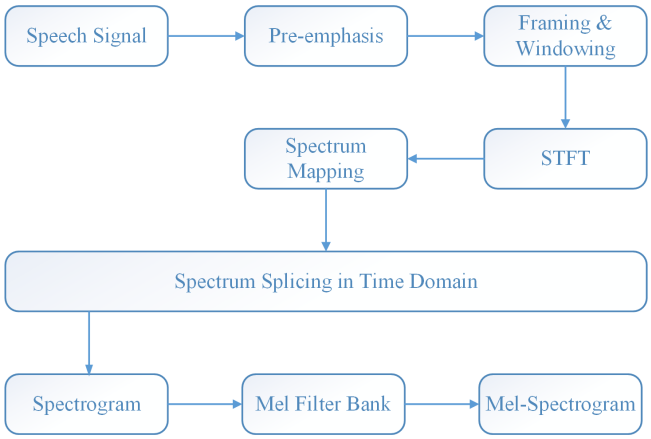


Figure 1. Flow chart of Mel-spectrogram extraction.

As shown in Figure 1, the feature extraction of Mel-spectrogram needs the following processes. (1) Pre-emphasis and frame-based windowing technology. The sampling frequency of voice signal is 16,000 Hz, the number of sampling points in each voice frame is N = 512, the pre-emphasis coefficient is 0.97, and the frame step is 200. (2) Fast Fourier transform. Each frame of speech signal divided by step (1) is very short, so it is considered that the frequency is in a stable state within a frame of time. The fast Fourier transform of each frame of speech signal $y_i(n)$ is calculated according to Equation (1) to obtain its spectrum. The multi-frame spectrum is spliced on the time dimension to form spectrogram.

$$F_{\mathrm{i}}(k) = \sum_{n=0}^{N-1} y_i^{'}(n) e^{\frac{-j2\pi}{N}kn} \tag{1}$$

The power spectrum of speech signal P is calculated by Equation (2).

$$P = \frac{|F_i(k)|^2}{N} \tag{2}$$

In which, $F_i(k)$ is the Fourier transform output of the ith frame of speech signal $y_i^{'}(n)$.

(3) Mel-filter bank. Mel-filter bank is used to convert frequency into Mel-frequency to simulate human perception. By passing the energy spectrum through multiple groups of Mel-filters bank, Mel-filters bank can approximate the human auditory system and have a certain choice of frequency, which can directly ignore the frequency signal that you do not want to perceive. The structure of Mel-filter is shown in Figure 2. Mel-spectrogram can be obtained from the spectrogram after passing through Mel-filter bank.

In this paper, the number of Mel-filter banks M = 40 is selected, and the response at the center frequency is 1, which gradually decreases to 0 at both ends, until the center frequency of two adjacent filters is reached, where the response is 0, and the interval between them increases with the increase of m value. Mel-filter banks can be constructed from Equation (3). The energy spectrum P of step (2) is passed through the Mel-filter bank $H_m(k)$ to obtain the Mel-spectrogram, which is shown in Figure 3.

$$H_m(k) = \begin{cases} 0 & k < f(m-1) \\ \frac{k-f(m-1)}{f(m)-f(m-1)} & f(m-1) \le k < f(m) \\ 1 & k = f(m) \\ \frac{f(m+1)-k}{f(m+1)-f(m)} & f(m) < k \le f(m+1) \\ 0 & k > f(m+1) \end{cases} \tag{3}$$

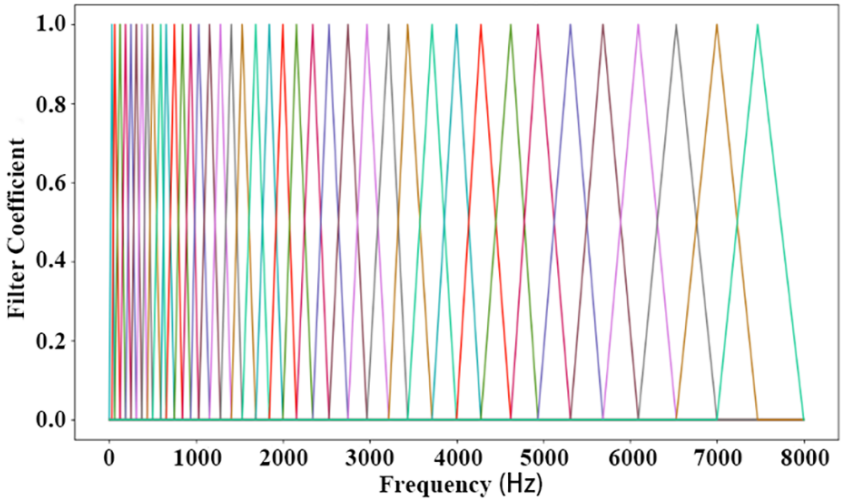


Figure 2. Mel-filter bank.

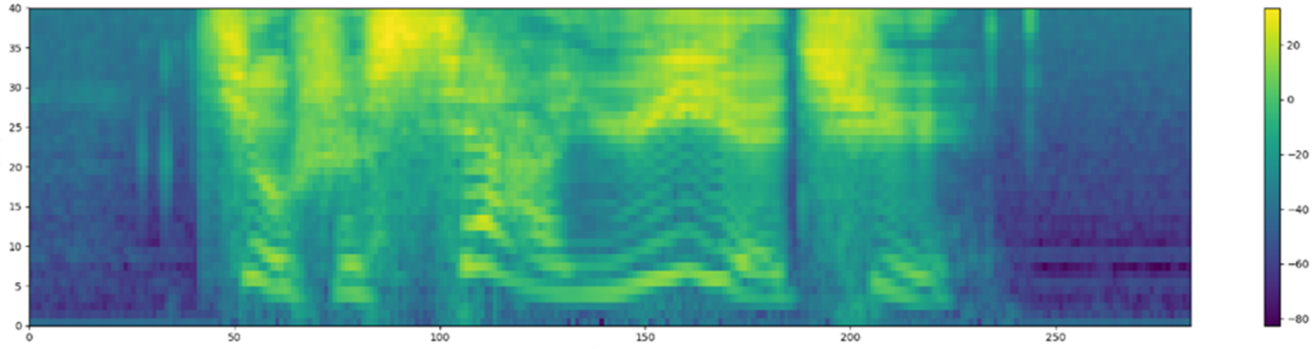


Figure 3. Mel-spectrogram.

It is generally believed that the more primitive the information, the more complete the information, and direct conversion of data generally leads to information loss. In theory, if the input is waveform data, there is a network that can automatically achieve FFT calculation, and it can also achieve any form of feature extraction. However, based on experience, manually defined transformations, such as spectrum or MFCC, are considered to perform well in speech emotion recognition tasks, so it is unnecessary to try another neural network to discover a better feature, which requires additional computational costs. The characteristic of Mel-spectrogram is that it contains both time-domain and frequency-domain information, and traditional transformations, such as FFT and Mel-filter, are also included. Therefore, Mel-spectrogram contains a large amount of original information, and some necessary transformations reduce the computational burden of the model. In addition, Mel-spectrogram can directly link convolutional layers, which makes it very suitable as an input for DRSN. Therefore, Mel-spectrogram is selected as input feature in this research.

3.2. Deep Residual Shrinkage Network

The notion behind the incorporation of the residual shrinkage module into the deep residual network has led to the development of the deep residual shrinkage network, as exemplified in Figure 4. The deep residual shrinkage network (DRSN) is a sophisticated deep neural network that employs attention mechanism and soft threshold function to remove noise in data. The self-attention mechanism utilized by the DRSN methodically selects relevant features while eliminating ineffectual features and noise, thereby elevating the deep neural network's capacity to extract valuable features from noisy data.

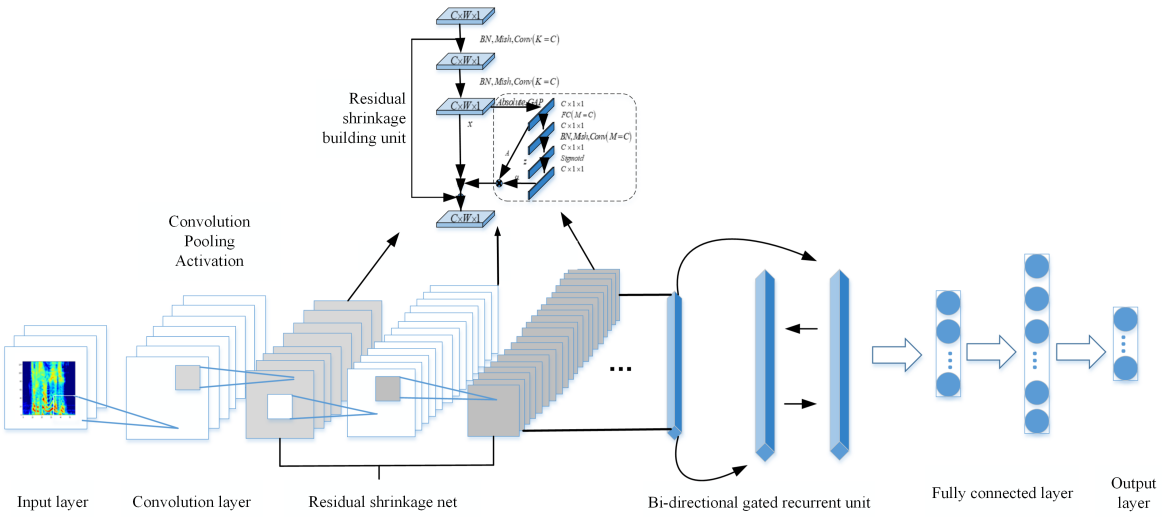


Figure 4. Structure of deep residual shrinkage network.

Convolution layer is an important component of the model, which can extract local features and reduce the number of weights through local connections and weight sharing of convolution kernels. The size of convolution kernels is an important parameter in building a model. Common convolution kernel sizes include 3×3, 5×5, and 7×7. A large convolution kernel size can increase the receptive field of the model, but the convolution calculation will also increase. If the size of the convolution kernel is small, the convolution calculation can be reduced, but then the receptive field becomes smaller. In the research of VGG-19, it is analyzed that stacking two layers of 3×3 convolution layer can be equivalent to 5×5 receptive field, and stacking three layers of 3×3 convolution layer can be equivalent to 7×7 receptive field [30]. When the receptive field is the same, multiple 3×3 convolution layers use multiple non-linear activation function, which will increase the non-linear expression ability, thus providing more complex pattern learning. In addition, using 3×3 convolution kernels can reduce parameters. Assuming that there are currently three layers of 3×3 convolution kernels stacked in a convolution layer with C input and output channels, its total number of parameters is $3 \times 3 \times 3 \times C \times C = 27C^2$. For one convolution layer with the same size of receptive field, the convolution kernel is 7×7. If the number of input and output channels are both C, the total number of its parameters is $7 \times 7 \times C \times C = 49C^2$. Overall, the stacking of 3×3 convolution kernel not only increases the depth of the network but also reduces the number of parameters, so the stacking of layers with 3×3 convolution kernel is selected for the network in this paper.

3.2.1. Residual Shrinkage Building Unit

Residual Shrinkage Building Unit (RSBU) is the basic unit of DRSN. The structure of residual shrinkage module is shown in Figure 5.

As shown in Figure 5, the DRSN is composed of two-batch normalization (BN), two activation functions (Mish), two convolution layers (Conv), one identity mapping, and one soft threshold learning subnetwork. There is a sub-network in each part and its function is to learn a set of thresholds independently. Its learning process is as follows:

(1) Take absolute values for all input features;
(2) An feature A is obtained after global average pooling (GAP);
(3) Input A into a fully connected network. The sigmoid function is on the last layer, and the output is a coefficient α in the range of 0 to 1;
(4) αA is the threshold to be learned.

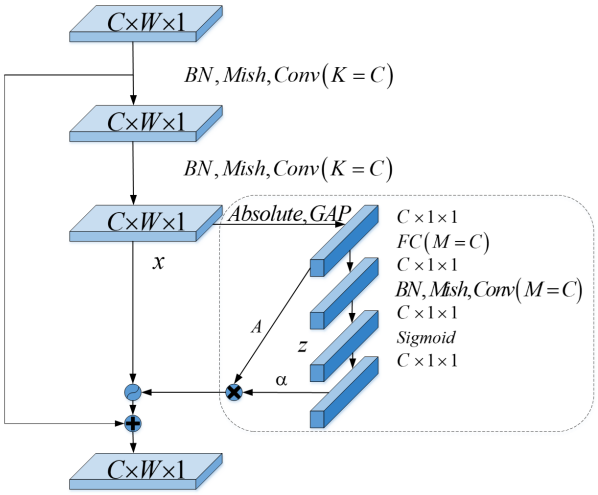


Figure 5. Residual shrinkage building unit.

In this way, the threshold is guaranteed to be positive and not too large. Different feature maps learn different thresholds, so the above subnetworks can be used as an

attention mechanism, the soft threshold converts the observed invalid features to zero; important features noted are preserved.

3.2.2. Soft Thresholding

In this paper, the soft threshold function is used to remove the features close to 0 and retain the positive and negative features. The soft threshold function is shown in Equation (4). In the equation, x is the input feature, y is the output feature, and thr is the threshold.

$$y = \begin{cases} x - thr & x > thr \\ 0 & -thr \leq x \leq thr \\ x + thr & x < -thr \end{cases} \tag{4}$$

Equation (5) is derived from Equation (4), and the soft threshold function becomes 1 or 0 after derivation.

$$\frac{\partial y}{\partial x} = \begin{cases} 1 & x > thr \\ 0 & -thr \leq x \leq thr \\ 1 & x < -thr \end{cases} \tag{5}$$

The process of soft threshold can also be converted into Equation (6), in which, x is the input data of the sub-network, thr is the threshold after network learning, and $sign(x)$ is the symbolic function.

$$soft = (x, a) = sign(x) * \max\{|x| - thr, 0\} \tag{6}$$

The key of soft thresholding is to obtain the range of threshold independently. The deep neural network has good performance in self-learning, so the combination of soft thresholding and deep neural network can well distinguish effective features from irrelevant features.

3.2.3. Self-Attention Mechanism

The attention mechanism principle involves facilitating neural network models to independently learn input features and assign weights to them. This approach allows the model to allocate ample computing resources to acquire crucial features, leading to improved performance.

The attention mechanism involves a process of key-value mapping through the query operation of vector matrix. This entails computing the similarity between dimension vectors and deriving the corresponding weight values, which are then normalized via the SoftMax function. Subsequently, the weight value is multiplied to the vector of each matrix dimension and their sum is obtained to formulate the final attention matrix. When $K = V = Q$, the process is referred to as self-attention and can be represented by Equation (7).

$$Attention(Q, K, V) = soft\max\left\{\frac{QK^T}{\sqrt{d_k}}\right\} \tag{7}$$

In which, $Q = (q_1, q_2, ..., q_l) \in R^{n \times d}$, $K = (k_1, k_2, ..., k_l) \in R^{n \times d}$, $V = (v_1, v_2, ..., v_n) \in R^{n \times d}$, d is the dimension of a single vector, n is the number of input vectors, which are obtained by linear transformation of input matrix X. K^T is the transposition of K, and d_k is a matrix with dimension k for adjusting inner product. The input sequence linear mapping process formula is as follows:

$$\begin{cases} Q = W_q X \\ K = W_k X \\ V = W_v X \end{cases} \tag{8}$$

In which, w_q, w_k, and w_v are three linear mapping parameter matrices, which are self learned in model training process.

3.3. Bi-Directional Gated Recurrent Unit

The gated recurrent unit (GRU) is a type of lightweight cyclic neural network that differs from other neural networks due to its internal "gate" structure. This unique structure enables the network to determine which data are relevant and which data can be discarded based on their relationship, facilitating effective data transmission within the network and effectively controlling internal information. As a result, the GRU partially addresses the issue of long-term dependence in neural networks. The internal structure of the GRU consists of three main components, the reset gate, update gate, and hidden state, which work together to extract temporal information and obtain long-term dependencies. The individual functions of these gates are described in further detail below, while the internal structure of the GRU is displayed in Figure 6.

(1) Reset gate and update gate

The input for the reset and update gates is obtained through the fully connected layer that operates on the current input x_t and the hidden state of the previous time step h_{t-1}. An activation function is applied to transform the input of the reset gate into a value between 0 and 1. This value is then used to determine the relevancy of the input. Specifically, an input with a value of 0 will be discarded whereas an input with a value of 1 will be retained. The product of the reset output and the previous hidden state is calculated element-wise. The update gate regulates the extent to which the previous state information is incorporated into the current state. Both the reset and update gates influence the candidate hidden state of the current neuron simultaneously. The reset gate can be computed as per Equation (9).

$$r_t = sigmoid\left(\left[w_{hr}|w_{xr}\right] \otimes \left[h_{t-1}|x_t\right]^T\right) \tag{9}$$

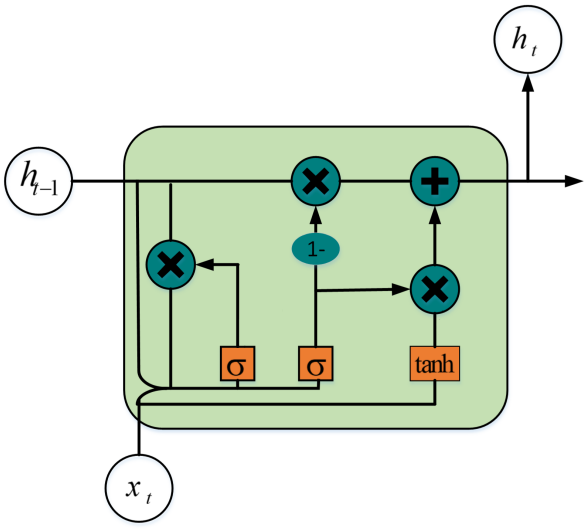


Figure 6. Structure diagram of gated recurrent unit.

In which $\otimes$ is cross product of matrix, t is the current time point, $t-1$ is the last time point, x_t is the input sequence of current time point, h_{t-1} is the hidden state of last time point, and w_{hr} and w_{xr} are the hidden state of last time and weight matrix of input sequence at current time point. According to the current information selection of the reset gate, the candidate hidden state is calculated by Equation (10).

$$\tilde{h_t} = \tanh\left(w_{\tilde{h}} \otimes \left[h_{t-1} \circ r_t\right) | x_t\right]\Big) \tag{10}$$

In which ∘ is the Hadamard operator, h_t is the hidden state and w_h is the weight matrix of candidate hidden state. The output of update gate is calculated by Equation (11), where the data should be updated.

$$z_t = sigmoid\left([w_{hz}|w_{xz}] \otimes [h_{t-1}|x_t]^T\right) \tag{11}$$

In which z_t is the output of update gate at current time. According to update gate, the retained information of hidden state at the last time point is calculated by Equation (12).

$$h_{t-1}^* = (1 - z_t) \circ h_{t-1} \tag{12}$$

(2) Hidden status

The hidden status is the output of the GRU through the candidate hidden status and the status information of the last moment reserved by the update gate, which is calculated by Equation (13).

$$h_t = \left(z_t \circ h_t'\right) \oplus h_{t-1}^* \tag{13}$$

In which, $\oplus$ is matrix addition operator.

(3) Bi-directional gated recurrent unit

The structure of BiGRU is shown in Figure 7.

For GRU, the information obtained at present is only affected by the information at the last moment. Generally, the status of hidden state transmission in GRU is from front to back. However, for the extraction of speech affective feature, we need to pay attention not only to the information at the current moment and the information at the previous moment, but also to the time state from the back to the front. It can be seen from Figure 7 that the hidden state of the current time step in the BiGRU network is affected by the current time step input sequence, the forward hidden state, and the backward hidden state. x_t is the input of BiGRU, $x_t \in R^{n \times d}$ (n is the number of small batch of input samples, d is the number of input moments). It is assumed that the forward hidden state is $\overrightarrow{h_t} \in R^{n \times h}$ and the backward hidden state is $\overleftarrow{h_t} \in R^{n \times h}$ which are calculated by Equations (14) and (15).

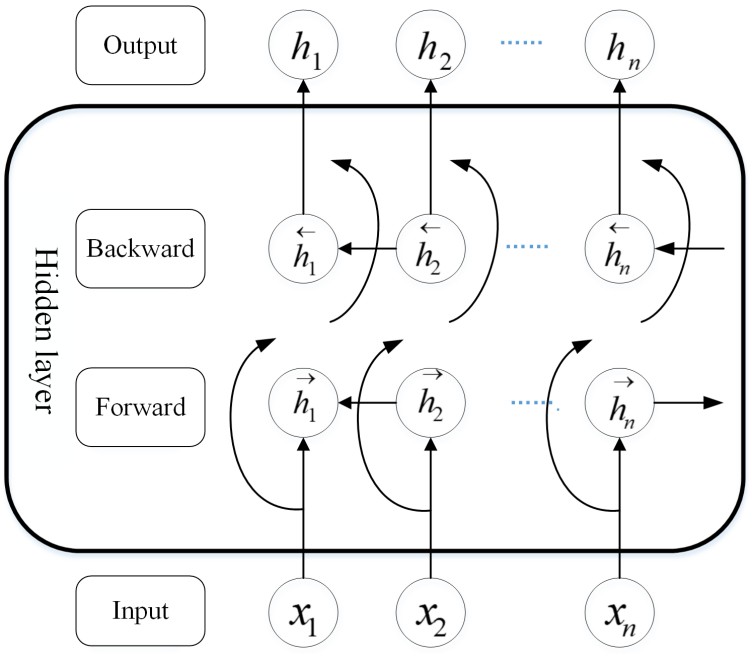


Figure 7. Structure of bidirectional gated recurrent unit.

$$\overrightarrow{h_t} = sigmoid\left(x_t w_{xh}^{(f)} + h_{t-1}\overrightarrow{w_{hh}^{(f)}} + b_h^{(f)}\right) \tag{14}$$

$$\overleftarrow{h_t} = sigmoid\left(x_t w_{xh}^{(b)} + \overleftarrow{h_{t-1}} w_{hh}^{(b)} + b_h^{(b)}\right) \tag{15}$$

In which, $w_{xh}^{(f)} \in R^{d\times h}$, $w_{hh}^{(f)} \in R^{h\times h}$, $w_{xh}^{(d)} \in R^{d\times h}$, $w_{hh}^{(b)} \in R^{h\times h}$ are weight matrix. Then the calculated $\overrightarrow{h}$ and $\overleftarrow{h}$ are connected to obtain the hidden state $h_t \in R^{n\times 2h}$ at current time, and the output layer $o^t \in R^{n\times q}$ is calculated by Equation (16).

$$o_t = h_t w_{hq} + b_q \tag{16}$$

4. Results

This paper used the DRSN to classify the emotion of the datasets, and use Tensorflow 2.2.0 deep learning framework and Python 3.8.0 to build the model. The hardware for the experiment are Intel Core i5-12750H CPU, RTX 3060 GPU, 32GB memory, and the parameter settings are shown in Table 2 below.

Table 2. Model training parameters.

Parameter	Value
Learning rate	10^{-4}
Epochs	160
Batch size	64
Optimization method	RMS Prop

Speech data were converted into Mel-spectrogram features as the input parameters of the network model, and DRN-BiGRU and DSRN-BiGRU were tested on CASIA and IEMOCAP speech emotion datasets, respectively. The performance of depth residual shrinkage module on the datasets are analyzed.

4.1. Datasets Introduction

The CASIA Chinese emotional corpus was curated by the esteemed Institute of Automation at the Chinese Academy of Sciences. The repository comprises of recordings by four proficient speakers and encompasses six distinct emotional states, namely anger, happiness, sadness, fear, surprise, and neutrality, totalling an impressive 9600 pronunciations. A noteworthy aspect of the corpus is that, while it contains 300 identical texts, the different emotional renderings of each exhibit the diverse acoustic and rhythmic performances through precise comparative analysis. In addition, the corpus features 100 distinct texts, which the literal meaning suggests as having inherent emotional leaning, facilitating more consistent and accurate emotion depiction. For our experiment, we handpicked the corpus's identical text for use as the experimental data, where the first 200 readings were assigned to the training set and the residual 100 allocated for testing. Furthermore, the experiment conducted partitioning of the dataset [31].

The Interactive Emotional Binary Motion Capture Database (IEMOCAP) was collected by the Speech Analysis and Interpretation Laboratory (SAIL) of the University of Southern California (USC). This database is an invaluable resource for the research and modeling of multimodal and expressive human communication, as it contains over 12 h of data recorded from 10 actors engaged in binary conversation scenes, with detailed motion capture information that identifies their facial expressions and hand movements. Additionally, the database includes interactive settings that stimulated specific emotions (such as happiness, anger, sadness, and depression) in emotionally scripted and naturally occurring conversation scenarios [32].

The MELD dataset, derived from the Emotion Lines dataset, features over 1400 conversations and 13,700 sentences drawn from the classic TV series Friends. In addition to the pure text dialogue present in the original dataset, MELD expands it by incorporating multimodal data, such as video and audio, including an average of 9.5 sentences per conversation and an average of 3.6 s per sentence. Each sentence is assigned one of seven emotional labels, including anger, disgust, sadness, happiness, neutrality, surprise, and fear, with corresponding emotional labels that are categorized as either positive, negative, or neutral [33].

The model proposed in this paper is trained by datasets, so the quality of data is important for the performance of the model. Three datasets are compared in Table 3 and there are enough data for the three datasets. Only CASIA dataset has a small number of speakers. In order to verify the performance of the model, all the three datasets will be used for experiments.

Table 3. Datasets ingrotuction.

Name	Speakers	Language	Data Volume	Emotions
CASIA	4 proficient speakers	Chinese	9600 pronunciations	anger, happy, sad, fear, surprise, neutral
IEMOCAP	10 actors	English	12 h	happy, angry, neutral, sad
MELD	many actors	English	13,700 sentences	happy, angry, sad, fear, surprise, neutral

4.2. Activation Funtion

The common activation function in deep learning are ReLU and Mish. This paper selects Mish as the activation function of the model which is shown in Equation (17). The positive value of Mish has no upper limit, which solves the problem of saturation caused by the upper limit. Compared with ReLU, whose negative values are all set to 0, Mish's slightly negative value theory can make gradient flow better. The smooth nature of Mish's can make the information in the input neural network more in-depth, thus improving the accuracy and generalization.

$$Mish(x) = x \times \tanh(\ln(1 + e^x)) \tag{17}$$

The activation function curve is shown in Figure 8. When the input data are negative, some small negative inputs can be retained as negative outputs to improve the interpretability and gradient flow of the network. When the input data are positive, linear operations are performed on the output and input to improve the convergence rate of the model.

The result of model with different activation is shown in Table 4. From the result, it can be seen that Mish performs better than ReLU on all three datasets. So the model proposed in this paper will select Mish as activation function.

Table 4. Comparison of actication functions.

Activation Function	CASIA	IEMOCAP	MELD
ReLU	84.62%	85.79%	69.6%
MiSH	86.03%	86.07%	70.57%

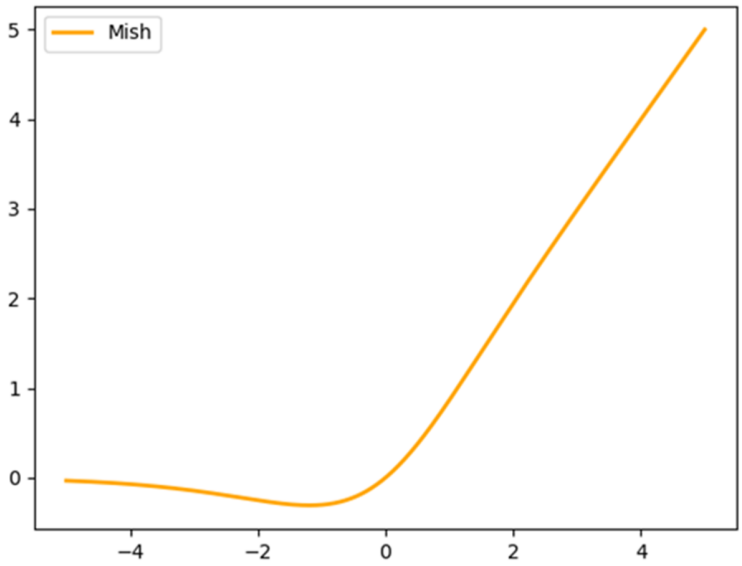


Figure 8. Mish function curve.

4.3. Optimization of RSBU Quantity

This paper outlines the composition of the DRSN-BiGRU model, incorporating multiple depth residual shrinkage building units and bidirectional gated recurrent units. The number of residual shrinkage building units employed correlates positively with the amount of speech emotional feature information extracted. However, an excess of residual shrinkage units can impede the network convergence speed, and only selective units can extract pivotal features and eliminate redundant ones. Thus, an experiment was conducted to determine the optimal number of residual shrinkage building units by comparing the model structures of varied units. The accuracy of the resulting network on CASIA and IEMOCAP datasets is illustrated in Figure 9.

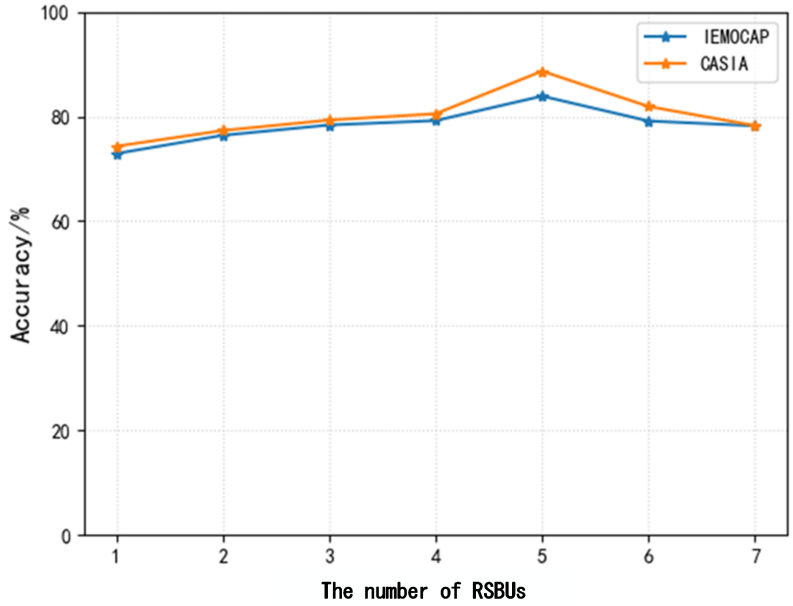


Figure 9. Influence of RSBU quantity.

Upon conducting an analysis, it was discovered that the utilization of a single residual shrinkage unit resulted in accuracy rates of 74.30% and 72.90% for CASIA and IEMOCAP, respectively. The addition of one more unit yielded improved accuracy rates of 79.31% and 78.37% for the respective datasets. Increased usage up to five residual shrinkage units provided the network with peak accuracy rates of 88.45% and 83.26%. However, adding further residual shrinkage building units cannot yield significant improvements and, in some cases, even resulted in decreased accuracy. This suggests that the initial residual shrinkage units are effective in extracting feature information, and subsequent additions only serve to complicate the network structure and disrupt the model. Based on the comprehensive analysis, it was determined that utilizing five residual shrinkage units results in the best performance for speech emotion recognition, and, thus, the DRSN-BiGRU in this paper employs the use of five residual shrinkage units.

4.4. Performance of DBSN-BiGRU

The DRSN-BiGRU model underwent training on the IEMOCAP dataset for 160 epochs. The accuracy curve depicting the learning process is displayed in Figure 10. It is evident from the learning curve that the performance of the model exhibited rapid enhancement during the initial 40 epochs of training. Subsequently, the performance level gradually stabilized and eventually reached a state where it no longer fluctuated significantly, manifesting a highly effective learning outcome. During the training process, the training curve and testing curve are almost synchronous, and there is no over fitting training.

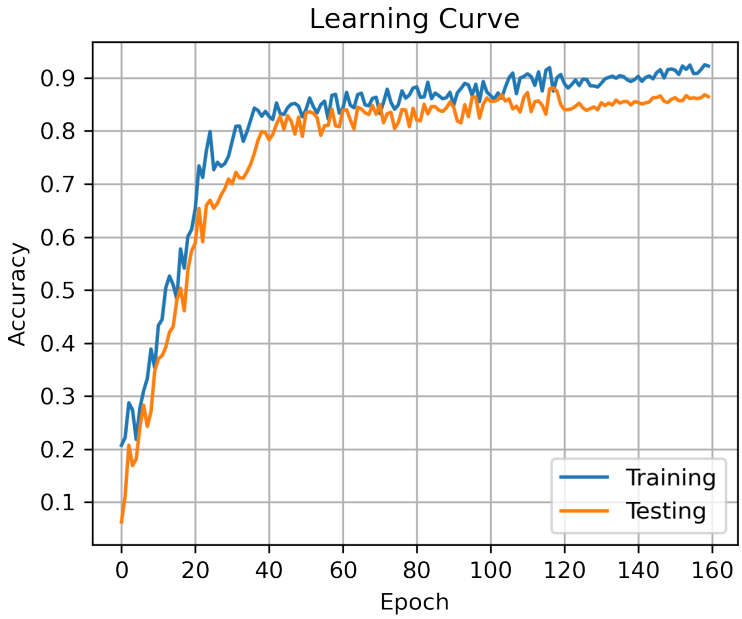


Figure 10. Learning curve of the models on IEMOCAP dataset.

The confusion matrix of testing result on IEMOCAP is shown in Figure 11. The highest accuracy is 90.2% for happy emotion and the lowest accuracy is 83.52% for angry emotion. The mean accuracy on IEMOCAP is 86.07%.

The DRSN-BiGRU model also underwent training on the CASIA dataset for 160 epochs. The accuracy curve depicting the learning process is displayed in Figure 12. It is evident from the learning curve that the performance of the model exhibited rapid enhancement during the initial 30 epochs of training which is similar to IEMOCAP dataset. Subsequently, the performance level gradually stabilized and eventually reached a state where it no longer fluctuated significantly, manifesting a highly effective learning outcome. During the training process, the training curve and testing curve are almost synchronous, and there is no over fitting training.

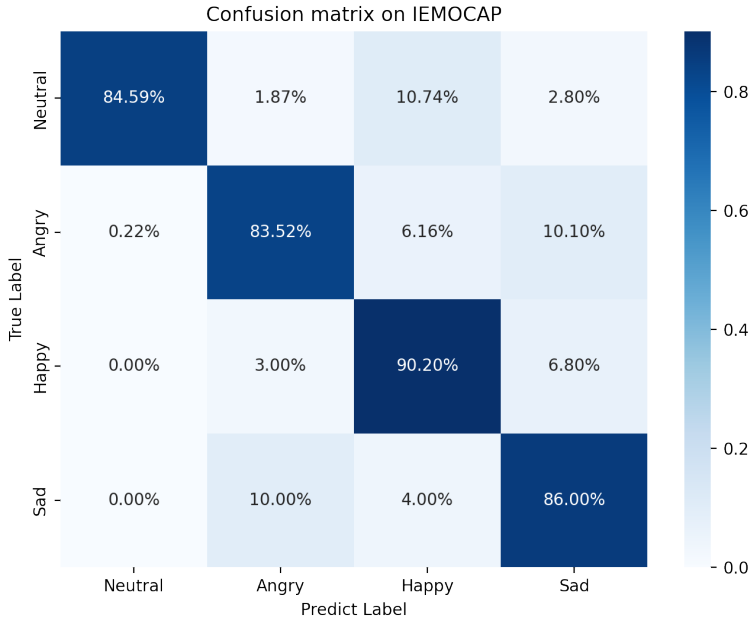


Figure 11. Confusion matrix of DRSN-BiGRU on IEMOCAP.

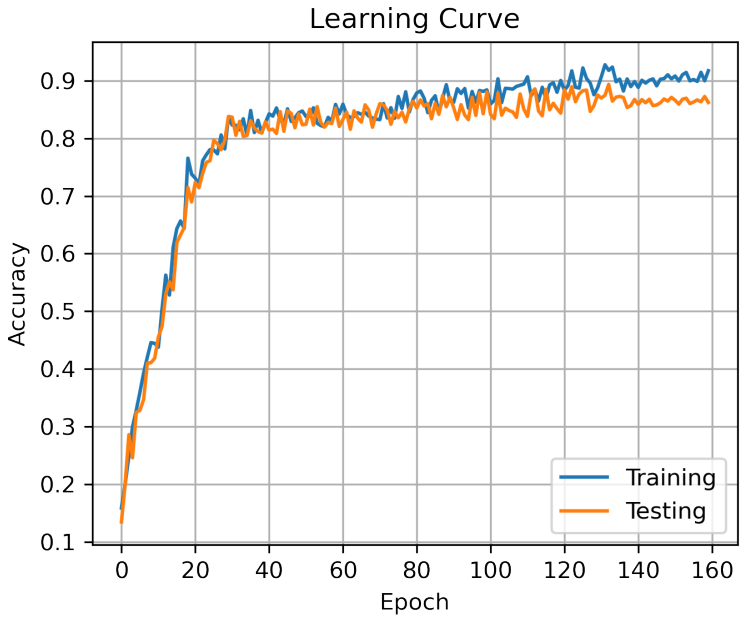


Figure 12. Learning curve of the models on CASIA dataset.

The confusion matrix of testing result on CASIA dataset is shown in Figure 13. The highest accuracy is 92.88% for happy emotion and the lowest accuracy is 78.65% for fear emotion. The mean accuracy on IEMOCAP is 86.03%.

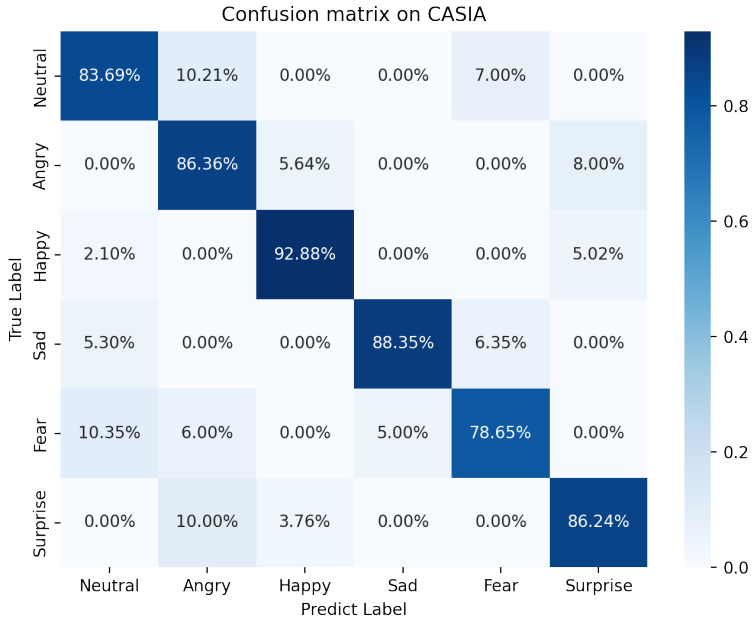


Figure 13. Confusion matrix of DRSN-BiGRU on CASIA dataset.

The same experiment are also carried out on MELD dataset and the accuracy curve depicting the learning process is displayed in Figure 14. It is evident from the learning curve that 160 epochs are also enough for MELD dataset and there is no over fit training.

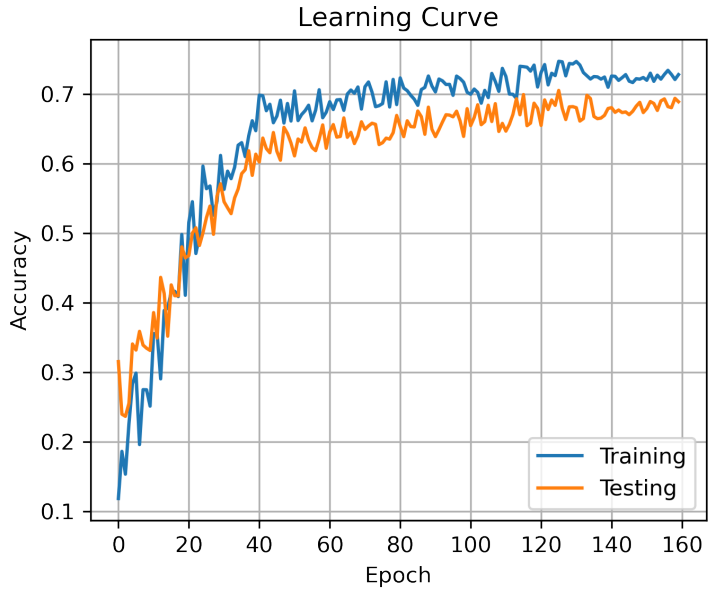


Figure 14. Learning curve of the models on MELD dataset.

In natural situations, human emotions are more multifaceted, influenced by environmental stimuli, rapid emotional shifts, mixed emotions, and subjective hidden emotions. Thus, recognizing emotions in real-life situations is more challenging. MELD dataset extracted from the Friends television series closely resembles natural speech patterns. The confusion matrix of the DRSN-BiGRU network model tested on the MELD dataset is illustrated in Figure 15.

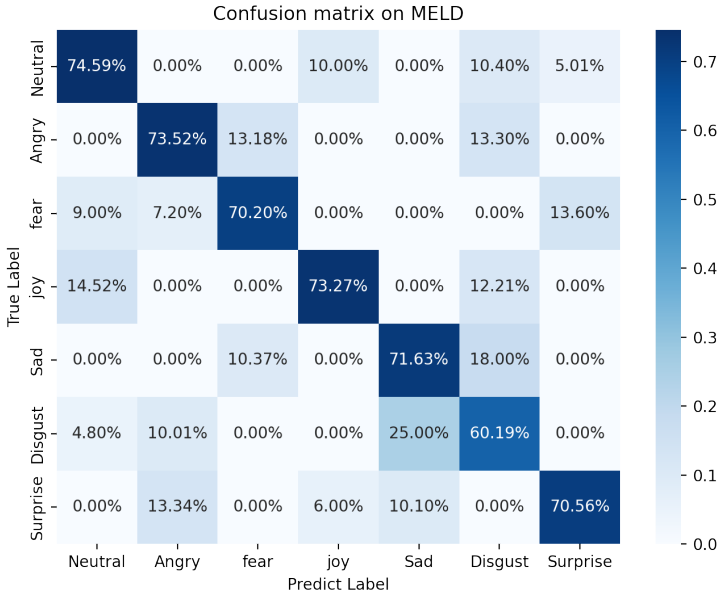


Figure 15. Confusion matrix of DRSN-BiGRU on MELD.

Upon analyzing the confusion matrix, it is evident that DRSN-BiGRU demonstrates a recognition accuracy of 70.57% on the MELD dataset. The model's accuracy is comparatively lower than CASIA and IEMOCAP due to the complexity of speech emotion recognition in natural situations, especially in the case of a seven-classification problem as in the MELD dataset. The DRSN-BiGRU performs exceptionally well in neutral, angry, and happy emotions, with a recognition accuracy of 74.59%, 73.52%, and 73.27%, respectively. Nevertheless, the model exhibits poor recognition on "disgust" emotion, with a recognition accuracy rate of only 60.19%. During the classification process of "disgust" emotion, the model has a 25% probability of categorizing it as "sad", a 10.01% probability of categorizing it as "angry", and a 4.8% probability of categorizing it as "neutral".

To further evaluate DRSN-BiGRU's efficacy in natural situations, the model was compared to DCNN-LSTM, CNN-BiGRU, and DRN-BiGRU, and their results are presented in next section.

5. Discussion

5.1. Results Discussion

From the comfusion matrix on IEMOCAP (Figure 10), it can be seen that anger and sadness are the most obvious confusing emotions. Because anger and sadness are both negative emotions, there is a portion of anger speech that does not erupt and also shows a relatively low voice, which is similar to the characteristics of sadness, leading to confusion between the two. Sometimes the speech of anger status depends on the speaker's personality. Some introverted people tend to be relatively introverted and less likely to burst out when angry, while some extroverted people tend to express aggressive behavior when angry, such as faster speaking speed, louder volume, and higher pitch. The differences in personality create differences in the expression of angry speech, which poses significant challenges in emotion recognition.

Compared to the IEMOCAP dataset, the CASIA dataset has increased the number of emotional labels, but the recognition accuracy has not decreased. This is because there are only four speakers in the CASIA dataset and the speech style is more unified, which, to some extent, compensates for the defect of more emotional labels. Another pair of easily confused emotions is neutrality and fear, which is manifested in the experimental results as 7% of neutrality being recognized as fear, and 10% of fear being recognized as neutrality.

From a psychological perspective, people sometimes conceal their inner emotions in a state of fear, and their speech is plain and neutral, so the two are easily confused. From the two datasets of CASIA and IEMOCAP, it can be found that the highest recognition accuracy is happy emotion. Happiness is the only positive emotion in the emotional labels, so the graph features are significantly different from others. In traditional emotion recognition methods based on feature extraction, such as energy, pitch, MFCC, formant, etc., it is easy to confuse happiness and anger. From the current research results, it can be seen that there is a significant difference between spectrogram features and traditional features.

In addition, it can be seen from the confusion matrix of MELD (Figure 15) that sadness and disgust are easy to be confused. From the perspective of language, these two emotions belong to negative emotions, and the vocal performance is relatively light, slightly slow, and more low-frequency voice, which makes the two emotions closer in Mel-spectrogram and leads to more confusion in recognition.

5.2. Result Comparison and Discussion

In order to verify the performance of DRSN-BiGRU algorithm, this paper compares it with CNN-BiLSTM, DCNN+LSTM, and DRN-BIGRU's accuracy of emotion recognition in IEMOCAP and CASIA, respectively. The accuracy of these algorithms for different emotions in CASIA dataset is shown in Table 5.

Table 5. Comparison result on CASIA.

Model	Surprise	Fear	Sad	Happy	Angry	Neutral
DCNN-LSTM	78.46	82.21	84.63	80.53	75.44	75.77
CNN-BiLSTM	70.46	73.21	74.31	70.46	65.13	65.32
DRN-BiGRU	81.38	80.46	86.68	82.46	71.45	80.94
DRSN-BiGRU	86.24	78.65	88.35	92.88	86.36	83.69

Table 5 highlights the significant improvement in accuracy achieved by the DRSN-BiGRU algorithm in recognizing emotions, such as "happiness", "anger", "surprise", and "neutrality", when compared with the other three algorithms. Additionally, DBSN-BiGRU shows a slight improvement in the recognition accuracy of "sad" emotions when compared with DRN-BiGRU. DRSB-BiGRU performs slightly inferior in recognizing "fear", but exhibits a significant improvement over CNN-BILSTM, and is outperformed by DCCN+LSTM and DRN-BIGRU. Overall, the results demonstrate that DRSN-BiGRU model outperforms the other three models. To further validate the performance of DRSN-BiGRU, Table 6 presents a comparison of results obtained from IEMOCAP.

Table 6. Comparison result on IEMOCAP.

Model	Sad	Happy	Angry	Neutral
DCNN-LSTM	77.46	81.21	81.63	78.53
CNN-BiLSTM	74.31	70.54	65.13	65.32
DRN-BiGRU	80.96	79.46	85.68	81.91
DRSN-BiGRU	86.00	90.20	83.52	84.59

Table 6 illustrates that the IEMOCAP emotional dataset consists of four categories. In contrast, DRSN-BiGRU demonstrates exceptional performance for the emotions of "happy", "sad", and "neutral" significantly enhancing recognition accuracy when compared to the remaining three algorithms. Moreover, DRSN-BiGRU's accuracy for recognizing "anger" surpasses that of CNN and DCNN, albeit slightly lower than that of DRN-BiGRU. This affirms that DRSN-BiGRU remains the top-performing algorithm among the four.

In order to further verify the superiority of DRSN-BiGRU in natural situations, the algorithm was also compared with DCNN-LSTM, CNN-BiGRU, and DRN-BiGRU in MELD dataset, and the results are shown in Table 7.

Table 7. Comparison result on MELD.

Model	Neutral	Angry	Fear	Joy	Sadness	Disgust	Surprise
DCNN-LSTM	68.32	52.13	54.21	50.32	55.43	53.31	56.65
CNN-BiLSTM	71.65	47.89	45.68	61.86	55.59	48.87	59.54
DRN-BIGRU	70.23	58.56	62.78	64.45	59.35	52.31	60.34
DRSN-BIGRU	74.59	73.52	70.20	73.27	71.63	60.19	70.56

The analysis of Table 7 reveals a notable decline in the performance of the four models on MELD, as opposed to their performance on CASIA and IEMOCAP. This outcome underscores the influence of naturalistic emotional complexity on the efficacy of these models. It is worth noting that DNSN-BiGRU yields the highest recognition accuracy for the seven emotions in question, establishing it as the optimal model for MELD.

In conclusion, the addition of residual shrinkage unit has led to significant improvements in speech emotion recognition accuracy for DRSN-BiGRU compared to CNN-BiGRU, DCNN, and DRN-BiGRU models. This is attributed to the residual network's capability of transmitting shallow information to the deeper network and the automatic removal of redundant features by the shrinkage unit. These experiments confirm the effectiveness of this method in speech emotion recognition.

5.3. Complexity Analysis and Discussion

The time complexity of a Deep Residual Shrinkage Network (DRSN) includes both forward and backward propagation. During forward propagation, the computing time complexity of DRSN is similar to that of a traditional fully connected neural network, $O(n^2)$, where n is the number of input samples. This is because, in each convolution layer, convolution and pooling operations need to be performed for each input sample, and the results are then passed to the next layer. During backward propagation, the time complexity of DRSN depends on the implementation of the back propagation algorithm. If the standard back propagation algorithm is used, the time complexity is $O(nm)$, where n is the number of samples and m is the number of parameters in the network. This is because for each parameter, a gradient calculation needs to be performed, and the time complexity of computing a gradient for each sample is $O(n)$.

The time complexity of this model is mainly reflected in the training process, so we discuss the time complexity of the training process. Due to the shared weights in the convolution layer, the size of the convolution kernel directly determines the training time complexity of the model. In this paper, a 3×3 convolution kernel is used, so its time complexity will be much smaller than the model uses 5×5 or 7×7 convolution kernel. The model structure also adopts a general DRSN structure, and there is no significant increase in network depth or the number of residual shrinkage units, so there is no significant improvement in time complexity. In addition, this paper adopts BiGRU, where the forgetting gate has forgetting characteristics, which can effectively avoid the gradient explosion problem happened during the training process. Overall, the time complexity of the entire model is controllable. When deploying the model, it is necessary to consider factors, such as the real-time requirements of the application scenario and the amount of data, and provide necessary computational support according to actual needs.

5.4. Real-Life Applications Discussion

Speech emotion recognition technology can enhance the user experience in human–computer interaction by providing more natural and personalized communication with computers, and the method proposed in this paper have reached a good performance. However, the real-life application of this technology still faces many challenges. First, model training depends on a big dataset. So far the training datasets are biased and lack representation of minority groups, which can lead to algorithms with poor performance in real-life application. Developing methods to increase dataset diversity and overcome this issue is necessary. Secondly, speech emotion recognition performance is influenced by language background, culture, and accent. Future research should investigate methods for overcoming these linguistic challenges. Thirdly, speech signals are often interfered by environmental noise, which can affect the accuracy of speech emotion recognition algorithms. Developing methods for enhancing speech signals in the presence of noise is crucial. Finally, computational complexity also needs to be considered. Our current experimental research does not require real-time calculation results. In human–computer interaction applications, real-time performance greatly enhances user experience rather than emotional understanding. So in real-life applications, it is necessary to research on deploying the model and the computational power requirements.

6. Conclusions

The utilization of speech emotion recognition technology plays a crucial role in the field of human–computer interaction. It offers a significant enhancement in user experience during human–computer interaction processes and serves a vital function in product recommendation, public opinion monitoring, human–computer dialogue, and other related areas. This paper focuses on exploring speech emotion recognition technology. Firstly, it suggests the use of Mel-spectrogram of speech as a feature, and elaborates on the extraction process of Mel-spectrogram meticulously. Then, a deep residual shrinkage network (DRSN) is proposed for speech emotion recognition. This network incorporates residual shrinkage building units, attention mechanisms, and bi-directional gated recurrent unit on top of the deep residual network (DRN), effectively allowing the network to disregard signal noise and enhance its ability to extract relevant features from noisy data. Finally, the paper performed network optimization and comparative verification on open emotional speech datasets, such as CASIA, IEMOCAP, and MELD. Experimental results indicate that the network performs best with five residual shrinkage units. Deploying the DRSN-BiGRU model resulted in an accuracy of 86.03% in CASIA, 86.07% in IEMOCAP, and 70.57% in MELD datasets. The model's performance in MELD is relatively poorer than the other datasets due to variations in human emotion's voice expressions in natural situations. Lastly, the paper compared DBSN-BiGRU to other state-of-the-art models, such as DCNN+LSTM, CNN+BiLSTM, and DRN-BiGRU on three datasets, and our proposed method proved to be superior.

Author Contributions: Methodology, T.H.; writing—original draft preparation, Z.Z.; conceptualization, C.D.; data curation, M.R.; supervision, X.J.; software, Q.Z.; project administration, T.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Jinhua Science and Technology Bureau, grant number 2022-1-046 and Jinhua Advanced Research Institute, grant number G202209 and G202207.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: http://www.chineseldc.org/resource_info.php?rid=76 and https://sail.usc.edu/iemocap/iemocap_release.htm.

Acknowledgments: This research was supported by Jinhua Science and Technology Bureau and Jinhua Advanced Research Institute.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wani, T.M.; Gunawan, T.S.; Qadri SA, A.; Kartiwi, M.; Ambikairajah, E. A comprehensive review of speech emotion recognition systems. *IEEE Access* **2021**, *9*, 47795–47814. [CrossRef]
2. Akçay, M.B.; Oğuz, K. Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers. *Speech Commun.* **2020**, *116*, 56–76. [CrossRef]
3. Zvarevashe, K.; Olugbara, O. Ensemble learning of hybrid acoustic features for speech emotion recognition. *Inf. Process. Manag.* **2020**, *13*, 70. [CrossRef]
4. Zhao, Z.; Bao, Z.; Zhao, Y.; Zhang, Z.; Cummins, N.; Ren, Z.; Schuller, B. Exploring deep spectrum representations via attention-based recurrent and convolutional neural networks for speech emotion recognition. *IEEE Access* **2019**, *7*, 97515–97525. [CrossRef]
5. Bhavan, A.; Chauhan, P.; Shah, R.R. Bagged support vector machines for emotion recognition from speech. *Knowl.-Based Syst.* **2019**, *184*, 104886. [CrossRef]
6. Fahad, M.S.; Deepak, A.; Pradhan, G.; Yadav, J. DNN-HMM-Based Speaker-Adaptive Emotion Recognition Using MFCC and Epoch-Based Features. *Circuits Syst Signal Process.* **2021**, *40*, 466–489. [CrossRef]
7. Shahin, I.; Nassif, A.B.; Hamsa, S. Emotion recognition using hybrid Gaussian mixture model and deep neural network. *IEEE Access* **2019**, *58*, 26777–26787. [CrossRef]
8. Liu, Z.T.; Wu, M.; Cao, W.H.; Mao, J.W.; Xu, J.P.; Tan, G.Z. Speech emotion recognition based on feature selection and extreme learning machine decision tree. *Neurocomputing* **2018**, *273*, 271–280. [CrossRef]
9. Ke, X.; Zhu, Y.; Wen, L.; Zhang, W. Speech emotion recognition based on SVM and ANN. *Int. J. Mach. Learn. Comput.* **2018**, *8*, 198–202. [CrossRef]
10. Daneshfar, F.; Kabudian, S.J.; Neekabadi, A. Speech emotion recognition using hybrid spectral-prosodic features of speech signal/glottal waveform, metaheuristic-based dimensionality reduction, and Gaussian elliptical basis function network classifier. *Appl. Acoust.* **2020**, *166*, 107360. [CrossRef]
11. Alex, S.B.; Mary, L.; Babu, B.P. Attention and feature selection for automatic speech emotion recognition using utterance and syllable-level prosodic features. *Circuits Syst. Signal Process.* **2020**, *39*, 5681–5709. [CrossRef]
12. Patnaik, S. Speech emotion recognition by using complex MFCC and deep sequential model. *Multimed. Tools Appl.* **2023**, *82*, 11897–11922. [CrossRef]
13. Bhangale, K.; Kothandaraman, M. Speech Emotion Recognition Based on Multiple Acoustic Features and Deep Convolutional Neural Network. *Electronics* **2023**, *12*, 839. [CrossRef]
14. Patil, S.; Kharate, G.K. PCA-Based Random Forest Classifier for Speech Emotion Recognition Using FFTF Features, Jitter, and Shimmer. *Proc. ICEEE* **2022**, *2*, 194–205.
15. Gumelar, A.B.; Yuniarno, E.M.; Adi, D.P.; Setiawan, R.; Sugiarto, I.; Purnomo, M.H. Transformer-CNN Automatic Hyperparameter Tuning for Speech Emotion Recognition. In Proceedings of the 2022 IEEE International Conference on Imaging Systems and Techniques, Kaohsiung Taiwan, China, 21 June 2022.
16. Kaya, H.; Fedotov, D.; Yesilkanat, A.; Verkholyak, O.; Zhang, Y.; Karpov, A. LSTM Based Cross-corpus and Cross-task Acoustic Emotion Recognition. In Proceedings of the Interspeech 2018, Hyderabad, India, 2–6 September 2018.
17. Zhao, J.; Mao, X.; Chen, L. Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomed. Signal Process. Control.* **2019**, *47*, 312–323.
18. Zhang, S.; Zhang, S.; Huang, T.; Gao, W. Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching. *IEEE Trans. Multimed.* **2018**, *20*, 1576–1590. [CrossRef]
19. Sun, L.; Zou, B.; Fu, S.; Chen, J.; Wang, F. Speech emotion recognition based on DNN-decision tree SVM model. *Speech Commun.* **2019**, *115*, 29–37. [CrossRef]
20. Huang, J.; Tao, J.; Liu, B.; Lian, Z. Learning Utterance-Level Representations with Label Smoothing for Speech Emotion Recognition. In Proceedings of the INTERSPEECH, Shanghai, China, 25–29 October 2020.
21. Atmaja, B.T.; Akagi, M. Evaluation of error-and correlation-based loss functions for multitask learning dimensional speech emotion recognition. *J. Physics Conf. Ser. IOP Publ.* **2021**, *1896*, 012004. [CrossRef]
22. Cai, X.; Yuan, J.; Zheng, R.; Huang, L.; Church, K. Speech Emotion Recognition with Multi-Task Learning. In Proceeding of the Interspeech, Brno, Czechia, 30 August–3 September 2021.
23. Yeh, S.L.; Lin, Y.S.; Lee, C.C. Speech Representation Learning for Emotion Recognition Using End-to-End ASR with Factorized Adaptation. In Proceedings of the Interspeech, Shanghai, China, 25–29 October 2020.
24. Bakhshi, A.; Wong, A.S.W.; Chalup, S. End-to-end speech emotion recognition based on time and frequency information using deep neural networks. In Proceedings of the ECAI 2020, Santiago de Compostela, Spain, 29 August–8 September 2020; IOS Press: Amsterdam, The Netherlands, 2020; pp. 969–975.
25. Sun, T.W. End-to-end speech emotion recognition with gender information. *IEEE Access* **2020**, *8*, 152423–152438. [CrossRef]

26. Sajjad, M.; Kwon, S. Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM. *IEEE Access* **2020**, *8*, 79861–79875.
27. Issa, D.; Demirci, M.F.; Yazici, A. Speech emotion recognition with deep convolutional neural networks. *Biomed. Signal Process. Control.* **2020**, *59*, 101894. [CrossRef]
28. Wang, Y.; Shen, G.; Xu, Y.; Li, J.; Zhao, Z. Learning Mutual Correlation in Multimodal Transformer for Speech Emotion Recognition. In Proceedings of the Interspeech, Brno, Czechia, 30 August–3 September 2021.
29. Zou, H.; Si, Y.; Chen, C.; Rajan, D.; Chng, E.S. Speech emotion recognition with co-attention based multi-level acoustic information. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022.
30. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
31. Li, Y.; Tao, J.; Chao, L.; Bao, W.; Liu, Y. CHEAVD: A Chinese natural emotional audio–visual database. *J. Ambient. Intell. Humaniz. Comput.* **2017**, *8*, 913–924. [CrossRef]
32. Yu, Y.; Kim, Y.J. Attention-LSTM-attention model for speech emotion recognition and analysis of IEMOCAP database. *Electronics* **2020**, *9*, 713. [CrossRef]
33. Poria, S.; Hazarika, D.; Majumder, N.; Naik, G.; Cambria, E.; Mihalcea, R. Meld: A multimodal multi-party dataset for emotion recognition in conversations. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Anomalous Behavior Detection with Spatiotemporal Interaction and Autoencoder Enhancement

Bohao Li [1,2,†], Kai Xie [1,2,3,*,†], Xuepeng Zeng [1,2], Mingxuan Cao [1,2], Chang Wen [3,4], Jianbiao He [5] and Wei Zhang [5]

[1] School of Electronic Information, Yangtze University, Jingzhou 434023, China; dqfh.st@yangtzeu.edu.cn (B.L.); xuepeng_zeng.st@yangtzeu.edu.cn (X.Z.); m-xcao.st@yangtzeu.edu.cn (M.C.)
[2] National Electrical and Electronic Experimental Teaching Demonstration Center, Yangtze University, Jingzhou 434023, China
[3] Western Research Institute, Yangtze University, Karamay 834000, China; wenchang@yangtzeu.edu.cn
[4] School of Computer Science, Yangtze University, Jingzhou 434023, China
[5] College of Information Science and Engineering, Central South University, Changsha 410083, China; jbhe@mail.csu.edu.cn (J.H.); csuzwzbn@csu.edu.cn (W.Z.)
* Correspondence: xiekai@yangtzeu.edu.cn; Tel.: +86-136-9731-5482
† These authors contributed equally to this work.

Abstract: To reduce the cargo loss rate caused by abnormal consumption behavior in smart retail cabinets, two problems need to be solved. The first is that the diversity of consumers leads to a diversity of actions contained in the same behavior, which makes the accuracy of consumer behavior identification low. Second, the difference between normal interaction behavior and abnormal interaction behavior is small, and anomalous features are difficult to define. Therefore, we propose an anomalous behavior detection algorithm with human–object interaction graph convolution and confidence-guided difference enhancement. Aiming to solve the problem of low accuracy of consumer behavior recognition, including interactive behavior, the human–object interaction graph convolutional network is used to recognize action and extract video frames of abnormal human behavior. To define anomalies, we detect anomalies by delineating anomalous areas of the anomaly video frames. We use a confidence-guided anomaly enhancement module to perform confidence detection on the encoder-extracted coded features using a confidence full connection layer. The experimental results showed that the action recognition algorithm had good generalization ability and accuracy, and the screened video frames have obvious destruction characteristics, and the area under the receiver operating characteristic (AUROC) curve reached 82.8% in the detection of abnormal areas. Our research provides a new solution for the detection of abnormal behavior that destroys commodity packaging, which has considerable application value.

Keywords: intelligent retail; anomaly detection; graph convolutional networks; action recognition; semantic segmentation

1. Introduction

The integration of deep learning into traditional vending machines has led to the emergence of smart retail containers that provide a better consumer experience through open-door shopping and automated checkout processes. This innovation, however, has also given rise to abnormal consumption patterns. For enterprises, relying on visual recognition technology to achieve large-scale operation of smart retail cabinets is currently the most profitable choice. Consumption behavior recognition under smart retail is mainly achieved through commodity identification. The first step is to identify the customer's purchase behavior for the movement trajectory of the product [1,2], and the second is to use the state and quantity of the product before and after the consumption behavior to identify the purchase behavior, for example, through the use of RFID [3] and image recognition [4,5] technology. Visual recognition schemes based on product identification

can satisfy the requirement of identifying normal consumption action, but are far from meeting the challenge of abnormal consumption action recognition, including destructive action. Detecting these anomalous or abnormal consumption patterns has considerable economic value for enterprises. However, few algorithms for detecting abnormal action in intelligent retail design are available.

Anomaly detection refers to detecting instances of data that are significantly different from most ordinary instances. Anomaly detection is emphasized in areas such as data mining, computer vision, and deep learning. In recent years, the widespread adoption of deep learning has led to a series of deep anomaly detection methods, which have shown high practical performance in practical applications such as autonomous driving [6–9] and pathological detection [10,11]. According to the classification of supervision methods, anomaly detection methods based on deep learning can be divided into unsupervised, semi-supervised and weakly supervised methods. Among them, semi-supervised and weakly supervised methods are more mature. According to the anomaly learning method, the anomaly detection algorithm involves two learning methods: general normality feature learning and anomaly measure-related feature learning. General normality feature learning methods typically involve predictability modeling using autoencoders or generative adversarial networks (GANs) [12–15]. This provides guidance for anomalous action detection through anomalous features. The general normality feature learning method shows excellent performance in detecting anomalous instances; however, there is another side to this approach. Because the resulting network can easily be misled into generating instance data from multiple instances, this can negatively affect anomalous determination results.

An autoencoder network is a mainstream algorithm for video-based anomaly detection. Deng H. et al. [16] proposed a new T-S model composed of a teacher encoder and a student decoder, and introduced the "reverse distillation" paradigm because of the good performance of knowledge extraction within the unsupervised anomaly detection problem. Piergiovanni A. J. et al. [17] proposed a universal visual backbone which easily adapts a ViT architecture to videos. Due to the fact that smart retail cabinets often capture videos with only one customer, it is better to use action recognition algorithms instead of autoencoder networks to detect abnormal actions. In consumer–action recognition problems, human actions and human–object interactions have an impact on the final consumer action judgment results; therefore, the human–object interaction (HOI) algorithm [18–21] and video data-based action recognition algorithm can be used for consumer–action recognition. Most existing HOI detection approaches are instance-centric, and cannot capture complex HOI behavior with only appearance features; therefore, Wang T. et al. [22] proposed a novel and fully convolutional version of HOI, and a method wherein HOI detection is posed as the key to solving detection and grouping problems. However, most current action recognition algorithms focus more on human action, which leads to a lack of interactive action information. Meanwhile, the human–object interaction recognition algorithm that can use the interactive action information cannot process the video data, that is, the time information of the action cannot be used. Secondly, abnormal actions that include destroying commodities are the main reason for the commodity loss rate of smart retail cabinets. This is because this type of behavior involves the customer taking the product and damaging it, then putting the product back in its original place. Therefore, this kind of behavior can easily be identified as the customer not purchasing the commodity. This suggests that the boundary between abnormal and normal interaction behavior is difficult to define.

In this paper, we detect anomalous interactions in both commodity and customer behavior, i.e., vandalism that includes tearing up the packaging of the commodities/unscrewing the cap. Graph convolution with skeletal nodes can filter out a large amount of background redundant information. Inspired by [23], which defines graph nodes as the suggested regional features of objects through different frames, we propose a human–object interaction graph convolution network (HOI-GCN) to connect commodity nodes and skeleton nodes for interaction feature extraction. Zhian Liu et al. [24] proposed a novel hybrid framework with a combination of flow reconstruction and flow-guided frame

prediction for video anomaly detection (HF2-VAD). The trained autoencoder network can reconstruct the normal behavior with a lower reconstruction error, and the reconstruction error for anomalous features is much larger. Reconstruction not only detects anomalous data, but also discovers the area wherein anomalies are located. Considering that the abnormal characteristics of the commodity are the key to distinguishing normal interaction behavior from abnormal interaction behavior, we use a confidence-guided difference enhancement module in the autoencoder networks to locate the damage area.

In conclusion, our contributions can be summarized as follows:

(1) To recognize the abnormal action and obtain the corresponding video frames, we propose an action recognition strategy for interactive action. We construct a human–object interaction graph convolution network to extract the features of abnormal action, and to recognize actions using interaction and temporal characteristics.
(2) To distinguish between abnormal interaction action and normal interaction action according to abnormal characteristics, we propose a confidence-guided anomaly enhancement module. We reconstruct less confident coded features using ordinary features with minimal similarity. At the same time, a perceptual loss function and a three-channel cosine similarity loss function are introduced to calculate the anomaly score of the image, and the division of the anomaly region is obtained.

2. Related Works

2.1. Anomaly Detection

Anomaly detection can be divided into GAN-based, CNN-based, and auto-encode-based detection. Morais R. et al. [25] proposed a method to model the normal patterns of human motion in surveillance videos, and used dynamic skeleton features for anomaly detection. Neelu Madan et al. [26] proposed a self-supervised block that can be applied in various advanced image and video anomaly detection algorithm frameworks. Feng et al. [27] proposed a multi-instance framework (MIST) and applied it to weakly supervised video anomaly detection (WS-VAD) to optimize a multiple-instance pseudo-label generator and self-guided attention-boosted feature encoder in MIST, using a self-training scheme. In the field of unsupervised learning, the earliest study using GAN-based predictability modeling was AnoGAN [28], proposed by Thomas Schlegl, which involved unsupervised learning of normal anatomical changes in the latent space of a normal sample through deep convolution GAN (DCGAN). The authors also designed two loss functions in the inference stage to calculate the difference between the generated and original graphs. The anomalous area was determined based on the weighted value and set threshold. Xia et al. [29] proposed two detection methods for fault detection and anomaly detection in semantic segmentation. They designed a network framework using the semantic segmentation module M combined with GAN networks to achieve the detection of anomalous items. In industrial anomaly detection, self-coding networks are the mainstream network. Memory modules and knowledge distillation models [30,31] are also widely used in self-coding networks for anomaly detection. Bae J. et al. [32] considered that the location and neighborhood information are ignored during the modeling of the current autoencoder network, which affects the distribution of normal features. They proposed a new modeling method, in which the conditional probability of a given neighborhood feature estimates the normal distribution and uses a multilayer perceptron network for modeling. To improve the speed of industrial anomaly detection, Kim D. et al. [33] proposed a method of fast adaptive patch memory (FAPM), eliminating unnecessary calculations, with good accuracy and speed.

2.2. Action Recognition

The basis of the solution can be divided into two parts: the first is relying on the extracted human body skeleton information for pose estimation, and the second is directly extracting the spatiotemporal features of the image for classification. Si Jie et al. [34] proposed a spatiotemporal graph convolution network (ST-GCN) based on the human skeleton, which represents the human skeleton node as a graph convolution node by

modeling the dynamic bone, and uses a spatiotemporal convolution graph to express the multi-layer skeleton joint sequence. Furthermore, Li proposed an action–structure diagram convolutional network (AS-GCN), combining action links and structural links [35] with an ability to learn the characteristics of spatiotemporal actions simultaneously; they proposed that the recognition head and the future pose prediction head should be added together, and that the action model should be established through self-supervision. Zhang X. [36] used asymmetric correlation metrics and a more advanced context-aware graph convolution network (CA-GCN) to compute contextual information, further expanding the flexibility of the algorithm. Cheng et al. [37] proposed a new shift GCN (shift-GCN) for modeling a graph convolutional neural network with the human skeleton as a space-time map, and effectively reduced the computational complexity.

In the image-based spatiotemporal algorithm, the dual-stream expansion 3D convolutional network (I3D) proposed by Carreira et al. [38] achieves the learning of spatiotemporal features through expanding the convolutional layer and pooling the kernels from 2D to 3D. The dual-stream architecture composed of optical flow and RGB also enhances the network performance. Liu T. [39] proposed STILT, a dual-stream network for spatiotemporal interactive learning. STILT abandons the practice of separating spatial information and temporal information capture in the early dual-stream network, and pays greater attention to the strong complementarity and correlation of the spatial and temporal information in a video. It was able to effectively improve the recognition accuracy by establishing the interconnection of time flow and spatial stream. Ji Lin et al. [40] proposed a computationally efficient temporal shift module (TSM) combined with 2D CNN backbone networks, which makes it possible to achieve low-latency video recognition with edge devices. Limin Wang et al. [41] used an RGB difference and curved optical flow field as the network input and proposed a time period network (TSN) to simulate the long-term structure of an entire video for long-distance time series modeling as a new framework for action recognition.

3. Methods

Figures 1 and 2 shows the flowchart of the algorithm, the action recognition based on human–object interaction graph convolution, and commodity damage feature detection under an autoencoder network based on a confidence-guided anomaly enhancement module. (1) The human–object interaction convolutional network is used to extract human action and interaction features. The position encoding and visual features are fused and fed into the interactive graph convolution and temporal interaction modules. (2) The corresponding encoder and decoder are obtained through network training. Confidence-guided difference enhancement is used to reconstruct latent features. The image generated by the encoder is compared with the original image at the pixel level and using depth semantics, and the damage calibration is obtained.

3.1. Action Recognition

3.1.1. Human–Object Interaction Feature Extraction

By observing videos of destructive action, we find that in the fragments in which the destructive behavior occurs, the hands are more related to the commodities in terms of movement and feature, especially during actions with obvious damage characteristics. Therefore, we encode the motion relationship between different frame objects. We define individual commodities and hands as our detection objects. Among these, the features of each object contain the center coordinate of the object along with its height, width and label as a vector, and we forward these to a multi-layer perception (MLP) network, yielding a d-dimensional feature and specifying the characteristics. Among these, the features of each object contain their own 2D coordinates and label information, taking the commodities as the key point.

Using a backbone CNN (e.g., ResNet-152 [42]), we can extract the image feature F of each object. The observations in [21] show that position embedding performs better

concerning actions that directly describe the movements of objects, such as "put something" and "take something".

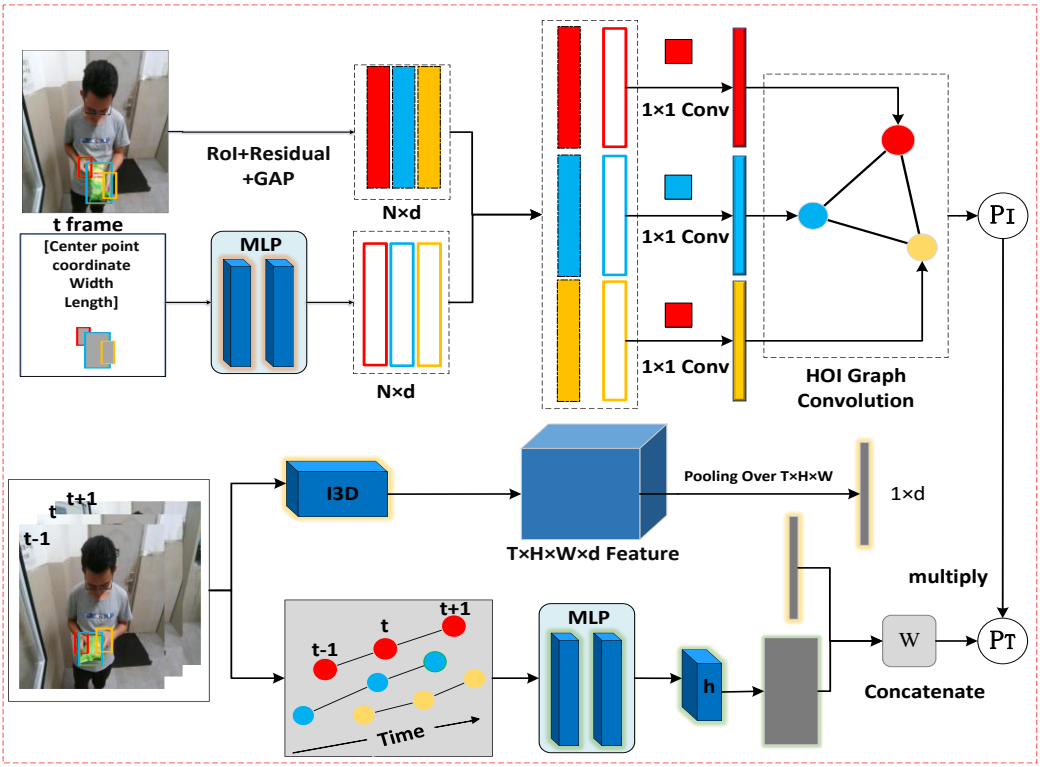


Figure 1. Overview of algorithm flow of the proposed architecture for action recognition under human object interaction graph convolution. For t-frame, we extract the features of each object using ROI + Residual + GAP, and encode the position information using two MLP layers. We also use 1×1 convolution to fuse the above two characteristics of each object. The fusion features are connected to form the characteristics of object nodes in the graph. Through the HOI graph convolution composed of the adjacency matrix G^I, the node matrix X, and the weight matrix W, we can obtain the action class probabilities P_I corresponding to the video frame t. For the temporal interaction module, we input the trajectory of the video sequence with the t frame as the center frame into the MPL network to generate the tracklets, and use the function h to combine and aggregate the information of the tracklets. The video features extracted using I3D are stitched with the motion feature matrix to classify the motion features. The video features extracted using I3D are stitched with the motion feature matrix, and sent into the matrix W to obtain the temporal interaction probabilities P_T. The final prediction is generated using P_I and P_T.

However, this method performs worse concerning actions that are associated more with changes in the intrinsic properties of an object, such as "tearing". For the t-frame video frame, the feature of object i is defined as f_i^t, which is calculated as follows:

$$f_i^t = GAP(Res_h(RoI(F))) \tag{1}$$

3.1.2. Human–Object Interaction Graph Convolutional Network

We first define the node characteristics in the graph convolution. After obtaining the position-encoded features of a frame of the region of interest, we connect the position features of the node and the image features. The fusion of the two features is achieved with a 1×1 convolution kernel. Through this step, we will obtain the position of the object and the fusion characteristics x_i^t of the image feature. The fusion features can be concatenated to form a feature of the object nodes in a graph, that is, the node matrix X.

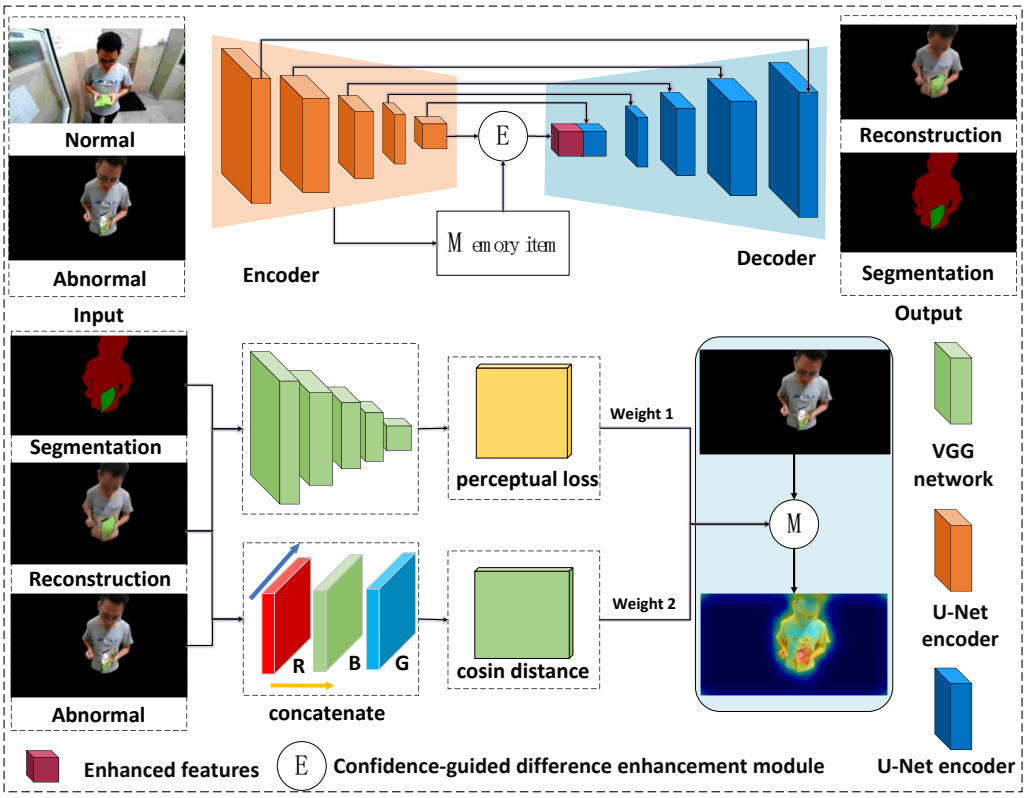


Figure 2. Overview of algorithm flow of the proposed architecture for confidence-guided anomaly detection. We use the U-net network as the autoencoder. The autoencoder is divided into two parts; one is the semantic segmentation network, which outputs the semantic segmentation map, and the other is the autoencoder with a confidence-guided anomalous enhancement module, which outputs the reconstruction image. We use two functions for the original map, semantic segmentation map and reconstruction map. By calculating the cosine distance between the reconstructed map and the original map and the perceived loss between the segmentation map and the reconstruction map, we obtain two anomaly maps. The final anomaly map is obtained using weighted fusion.

When creating an interactive adjacency matrix for different objects in video frame t, a popular method is to use the position information of the objects to define the continuity between nodes. We define the coordinates of object i node as U_i. The relative coordinates are fed into the MLP layer, which is put into the corresponding point of the critical matrix as a proposal probability of size 1 between neighboring nodes. The adjacency matrix G of size N × N is obtained:

$$G_{i,j}^{I} = MLP(U_{i^t} - U_{j^t}) \tag{2}$$

We used graph convolutional networks (GCNs) [43] to generate human–object interaction graphs. Graph convolution is different from standard convolution in that performing graph convolution is equivalent to performing message passing within the graph nodes. The output of the GCN is the updated feature of each object node; all the features of all object modes can be aggregated together for video classification. Formally, we can express the convolution of a layer of graphs as

$$P = GXW \tag{3}$$

where G represents an adjacency graph with N × N dimensions, X is the input feature of the object nodes in a graph with N × d dimensions, W is the weight matrix of d × d dimension layers, and the output of a graph convolutional layer Z still has N × d dimension. In our network, the W is the weight matrix of d × 1 dimension. It is used as the final classifier.

Additionally, the output is the action class probabilities. We defined output P of the layer Grap^{I} as

$$P_I = \sigma(G^I X W) \tag{4}$$

Finally, the output results after the graph convolution calculation were forwarded to the sigmoid function σ, which calculates the action class probabilities P_I of size N.

3.1.3. Temporal Interaction Module

Trajectory characteristics in motion are critical to understanding video. When describing the motion, we use the temporal interaction module proposed in [21]. It computes the feature of the tracklet as $g(x_i^1, \ldots, x_i^T)$, and uses the non-local block [44] as the function h to combine and aggregate the information of the tracklets. It also uses the architecture of the 3D convolutional neural network proposed in [23] to perform 3D convolution operations and average pooling on the video sequence; subsequently, the feature P_T of N dimension will be obtained. We stitch together the 3D features P_D and tracklet features as follows:

$$P_T = \sigma(W_p^T [P_D, h(g(x_i^1, \ldots, x_i^T)_{i=1}^N)]) \tag{5}$$

where [] represents the concatenation, and W_p^T is used as a final classifier with cross-entropy loss. Finally, we combine the action class probabilities P_I and temporal interaction probabilities P_T by multiplying the probabilities, a method similar to previous work [45,46]. P_F represents the final prediction vector of size N.

$$P_F = P_T \times P_I \tag{6}$$

3.2. Confidence-Guided Anomaly Detection

In the process of identifying anomaly action, the appearance of commodity packaging damage characteristics is the most reliable judgment for identifying the occurrence of sabotage; however, detecting and defining the damaged parts remains difficult, because the damage feature does not appear in normal consumption activities. The detection of damage features in the image is equivalent to the detection of out-of-detection (OOD) objects. In classification, OOD objects will be classified as any possible objects in the distribution. Based on action recognition, we performed semantic segmentation of commodity areas, and to simplify the complexity of the calculation, we classified the classification results of the segmentation model into $L = n + 2$ types, that is, n commodities, customers, and backgrounds. Specifically, x is a video frame of size $w \times h$, and $L = \{1, 2, \cdots, L\}$ is a set of integers representing semantic labels. A pixel-level semantic label is obtained by sending image x to segmentation model M, where we use U-Net [47] for our segmentation model. At the same time, another type of U-Net [48] was used as autoencoder. It removed the last batch normalization and ReLU layers in the encoder instead of an L2 normalization layer, causing the feature to have a common scale.

After the segmentation was complete, we masked the background of the image before and after synthesis according to the segmented semantic label map, and processed the pixels in the background as black in order to reduce the interference generated by the background pixels during the comparison process, and to reduce the amount of computation required for training and testing.

3.2.1. Confidence-Guided Difference Enhancement Module

We use the encoder to extract features from the input video frame t to then generate the query map q_t. The size of it is $H \times W \times C$, where $H, W,$ and C are height, width, and the number of channels. $q_t^k \in \mathbb{R}^C (k = 1, \cdots K)$ denotes individual queries of size $1 \times 1 \times C$ in the query map, where $K = H \times W$. We also use the memory items, which contain M items recording various normal data. The normal data were generated by the encoder of the normal frame containing the people and commodities. The normal frame of the customer was captured at the beginning of the video. We denote the item in the memory

using $v_m \in \mathbb{R}^C (m = 1, \cdots M)$. We calculate the cosine similarity between each query q_t^k and all items v_m in the memory to obtain a two-dimensional correlation graph of size M × K. By performing a global softmax function, the probability w_t^k of q_t^k matching the v_m is obtained. Figure 3 shows the flowchart of the confidence-guided difference enhancement module.

$$w_t^{k,m} = softmax\left[\frac{exp((v_m)^T q_t^k)}{\sum_{m'=1}^{M} exp\left((v_{m'})^T q_t^k\right)}\right] \tag{7}$$

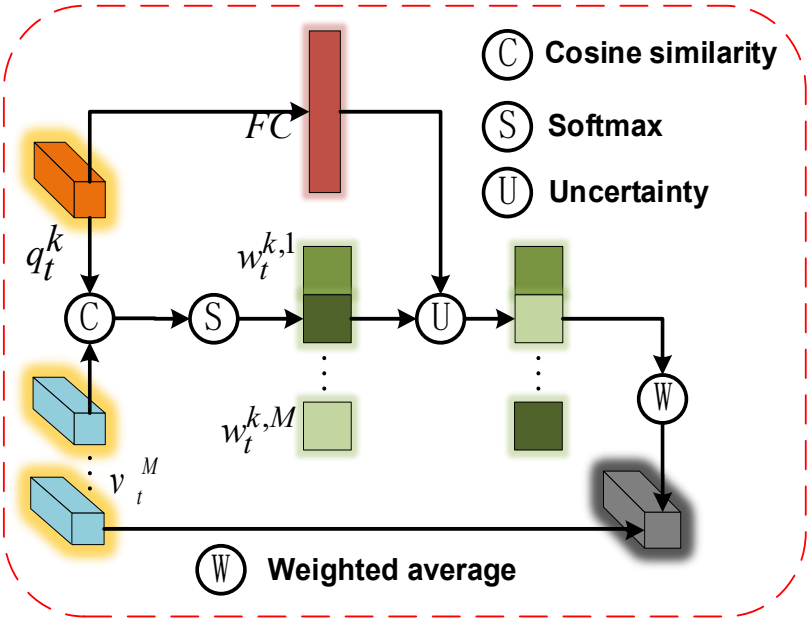


Figure 3. Illustration of a confidence-guided difference enhancement module. To read items in the normal memory, we compute matching probabilities $w_t^{k,m}$ in between the query q_t^k and items v_m, and applied a weighted average of the items with the possibilities to obtain the feature. With the confidence of the query q_t^k, a fully connected layer is used to change the weight of items. Finally, we multiply items by their weight, and add the items together to obtain the reconstructed query.

We add a fully connected layer to q_t^k as a confidence branch of the network, measuring the confidence level of the network evaluation. When the network confidence level is low, we reverse the original weight $w_t^{k,m}$ in succession to obtain features that are more different from the original image. By guiding the decoder to direct the anomalous features in the direction in which they are least similar, the difference can be increased. The transformed feature map and query features are assembled into the decoder. The formula is as follows.

$$\hat{v}_t^k = \begin{cases} \sum_{m'=1}^{M} w_t^{k,m'} v_{m'} & if\ FC(q_t^k) < \delta \\ \sum_{m'=1}^{M} \left(1 - w_t^{k,m'}\right) v_{m'} & if\ FC(q_t^k) \geq \delta \end{cases} \tag{8}$$

To train a fully connected layer of confidence, we only feed normal items into the fully connected layer. We denote the similarity of normal feature q_t^k to memory item v_m using w_m^k. We define y as the target probability distribution of the input normal features. The target probability distribution can give the network "hints". The confidence c denotes the probability that a query belongs to the memory items. Through training, normal features attain higher confidence in fully connected networks. The formula is as follows:

$$w_t^{k',m} = (1 - c) \cdot w_t^{k,m} + c \cdot y_t^{k,m} \tag{9}$$

We calculate the task loss using the prediction of negative log likelihood. To prevent the network from always choosing to receive the entire ground truth for minimizing the loss function, we add a binary cross-entropy loss function and balance the two losses using a hyperparameter λ; the loss function is as follows:

$$\mathcal{L}_c = \lambda \left(\sum_{m=1}^{M} y_t^{k,m} \log \frac{1}{w_t^{k',m}} \right) + (1-\lambda) \log \left(\frac{1}{c} \right) \tag{10}$$

3.2.2. Comparison of Shallow Features

In the process of calculating the characteristic value of commodity damage, not all abnormal feature values have an equal impact on the judgment results. If only the single feature value of each pixel is compared in the process of comparing the image, the calculation result will produce a large error, which is not conducive to the overall understanding of the damaged area. This is because the production of abnormal features of commodities and the destruction of commodity packaging have a strong causal relationship. Therefore, we synthesized the changes in hand nodes in the n frames before and after the selected video frame, and performed pixel-level feature vector stitching. We let the horizontal and ordinate changes of the nodes of the two adjacent frames be expressed as $K_{x_m}^p, K_{y_m}^p$, specifying that the left and down movements are negative values, where p represents the hand, and m is the m-frame in n frames.

$$\overset{\wedge}{\theta} = \arctan \left(\sum_{m=2}^{n} K_{x_m}^p, \sum_{m=2}^{n} K_{y_m}^p \right) \tag{11}$$

We stitched the pixels in position according to the size of the frame of the selected video frame w × h, where f_M^i is the feature vector of the RGB channel at the i^{th} pixel position of the last layer output of the segmentation model M. By stitching the feature values in the $\hat{\theta}$ direction, we filled the rest of the video frame $\hat{\theta}$ with black according to w×h, and the feature vector of the stitching $f_{\hat{\theta}_w}^i$, where $\lfloor, \rfloor$ represents the splicing of pixels. When the sum value of the motion coordinates is small, we skipped the area aggregation module and directly compared the pixel-level features of each point.

$$f_{\hat{\theta}_w}^i = \left\lfloor f_{\hat{\theta}_w}^{i_1}, f_{\hat{\theta}_w}^{i_2}, \ldots, f_{\hat{\theta}_w}^{i_{h+w}} \right\rfloor \tag{12}$$

Subsequently, for regions with large outliers, we compared the cosine distance de-fined on the intermediate feature of each element in the region:

$$S_n^{(i)} = F(x,r) = 1 - \left\langle \frac{f_M^i(x)}{\| f_M^i(x) \|_2}, \frac{f_M^i(r)}{\| f_M^i(r) \|_2} \right\rangle \tag{13}$$

where $\langle \cdot, \cdot \rangle$ is the inner product of two vectors.

3.2.3. Comparison of Deep Features and Fusion

Although shallow features can make good use of pixel-level information, the expression of higher features is missing, and the global features of the broken area are not expressed. We expressed the semantic differences between the segmentation map and the composite map by calculating the perceived loss of each pixel. Furthermore, by using the semantic differences between abnormal objects in the composite image, we can compare the deep features of each pixel.

The pixels of the original video frame are defined as $f_M^i(x)$, and the pixels of the composite video frame are defined as $f_M^i(x)$, where $F^{(i)}$ represents the elements of layer i

in the N layer of the VGG network, which are normalized between [0, 1]. The perceived difference expression is as follows:

$$D(x,r) = \sum_{i=1}^{N} \frac{1}{M_i} \parallel F^{(i)}\left(f_M^i(x)\right) - F^{(i)}\left(f_M^i(r)\right) \parallel_1 \quad (14)$$

Subsequently, we synthesized the differences of the deep and shallow layers of the same position, and assigned different weights to the deep features and shallow feature differences of each node, wherein we define the weight parameter as λ_S, λ_D. Finally, depending on the presence or absence of abnormal areas in the commodity, a fine judgment of the damage to the commodity can be made. The final synthetic difference is SD, which can be determined using the following formula:

$$SD = \lambda_S \cdot S_n + \lambda_D \cdot D \quad (15)$$

4. Experimental Results and Analysis

4.1. Experimental Platform and Dataset

The following platforms were used in this experiment: Windows 11; Graphics: NVIDIA GeForce GTX1650; Server: Nine days of ascension, Baidu PaddlePaddle. We used the Pytorch deep learning framework, PaddlePaddle 2.3 networking framework, and Python 3.8 to build the network model.

Because the commodity destruction video dataset is still blank in the public dataset of anomaly detection, the actual situation of the popular public dataset of anomaly detection and the abnormal behavior of consumption are quite different. To ensure the validity and accuracy of the experiment, it was conducted using homemade datasets. For each video, only a single customer consumes, excluding multiple people consuming at the same time. The consumer is in a bright environment. Additionally, all the customer's behavior is in the area that can be monitored by the camera, and there is no malicious occlusion and other problems in the customer's consumption behavior. Since the container door is transparent, the camera is placed above the inside of the container, and the shooting area is directly facing the retail container door. The inside of the smart retail container is equipped with light strips. This ensures that customers are also under a good light source when consuming at night. The video starts recording when the customer opens the door, and stops recording when the customer closes the door.

We divided consumer behavior into four behavioral categories (buying, not buying, drinking water, and tearing up packaging) and included three types of commodities (potato chips, biscuits, and drinks). The commodities used did not include those with transparent outer packaging. We also tested our algorithm on the MVTec AD dataset shown in Figure 4. Figure 5 shows the four behavioral data points, and Table 1 lists the details of our dataset. This is a dataset that mimics industrial production scenarios and is mainly used for unsupervised anomaly detection. The dataset contains five textures and ten objects from different domains.

Table 1. Details of video datasets used in our experiments.

Actions	Train	Resolution	Commodity	Test
Destory	397	1280 × 720	4	44
Purchase	891	1280 × 720	7	98
No purchase	412	1280 × 720	7	45
Drink	372	1280 × 720	3	41

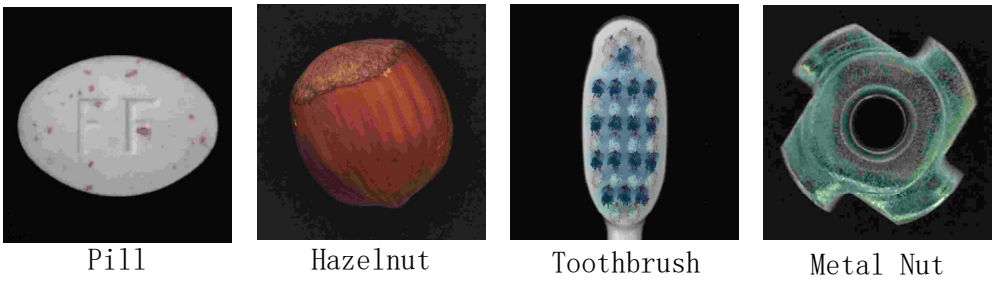


Figure 4. Some presentations from the MVTec AD dataset. It provides pixel-level labeling for anomalous regions, with only normal samples in the training set and normal and defect samples in the test set. Abnormalities include different types of defects such as scratches, dents, contamination, and different structural changes. In our experiments, we mainly use object-type data, which include metal nut, hazelnut, pill, etc.

Figure 5. Some presentations on homemade datasets. Our dataset divides the behavior into four types.

4.2. Comparision with State-of-the-Art Models

4.2.1. Action Recognition

To verify the effectiveness of the human–object interaction graph convolutional network, a comparative experiment was designed to compare two indicators: the area under the curve (AUC), and the receiver operating characteristic curve (ROC). We plotted the ROC curve for each action in the selected algorithm and used the AUC as an indicator to test the generalization ability of the selected model. The parameters in the code were set to 50 epochs with a batch size of 64. Figure 6 presents the ROC curve. Table 2 lists the AUC calculation indicators, the accuracy of the model detection, and the time spent on 57 videos for each network. The video is recorded from a top-down perspective, and this greatly interferes with the performance of the graph convolutional network based on human skeletons. We use the timesformer structure proposed by Gedas Bertasius et al. [49] to understand the video, which is completely based on the self-attention mechanism and does not use the convolutional structure. The timesformer structure not only enhances the spatiotemporal modeling ability of the network, but also reduces the complexity and computational load of the network.

Table 2. Comparison experiment of AUC and Top-1 accuracy of consumer behavior recognition with various algorithms.

Algorithm	Input	AUC	Top-1 Accuracy
ST-GCN	Skeleton	0.76	0.65
ST-GCN(AT)	Skeleton	0.81	0.74
MS-G3D [50]	Skeleton	0.86	0.81
Timesformer	RGB	0.87	0.84
TSN	RGB	0.92	0.85
I3D + STIN + OIE + NL [21]	RGB	0.84	0.82
Ours	RGB	0.90	0.86

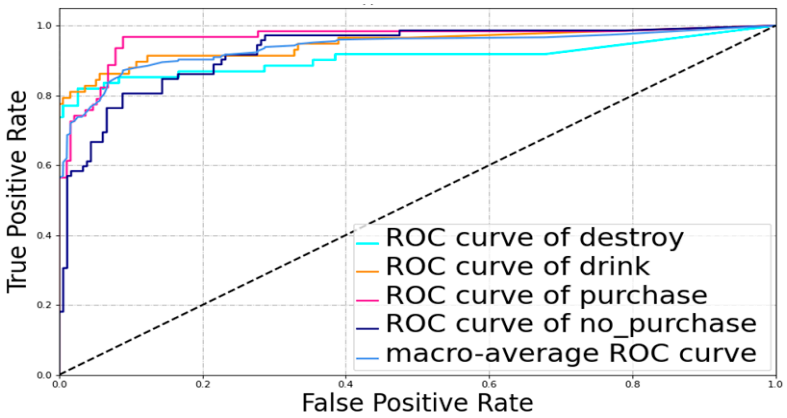


Figure 6. Visual results of ROC curve under our methods, showing ROC curves for the four modes of action recognition, and a macro-average ROC curve showing the generalization ability of the algorithm.

The videos from the visual system are important for identifying results. When the camera's shooting area does not cover the main parts of the human body or occludes too much, the performance of the network based on skeleton nodes will be seriously degraded. The quality of the image taken has a significant impact on the network, wherein the input data is RGB. The more video frames the visual system takes per second of an action, the more data we will obtain. Thus, our vision system for shooting video uses an image resolution of 1280 $\times$ 720 and an fps of 27. The influence of the visual system within the network is minimized.

4.2.2. Anomaly Segmentation Method on MVTec AD

We used AUROC and FPR95 as our evaluation indicators in anomaly segmentation methods. Furthermore, we represent the abnormal as TP at the broken part of the commodity, and as FP if the normal area is marked as abnormal, where i represents the true value, j represents the predicted value, and p_{ij} represents the number of pixels that predict i as j.

In the comparison of public datasets, our algorithm has more descriptions of motion characteristics, which is useless in industrial datasets. Our algorithm tends to increase the error according to the labels. That is, we can decode the abnormal characteristics of the product according to the human body or background characteristics through the confidence-guided difference enhancement module, so as to widen the difference between the abnormal characteristics of the product and the product. This does not work in some industrial data sets. Additionally, some of the perceptual loss functions perform poorly in some feature textures.

Therefore, we only tested some data with two labels. Although our algorithm still has shortcomings compared to the current algorithm, it shows better results in some complex datasets. Table 3 presents the results.

4.2.3. Video Frame Selection

Our action recognition algorithm defines the t-frame as the central frame, and obtains the probability distribution of the destruction. We use the probability of destruction as the selection weight for the t-frame. The classical algorithms for video frame selection include the clustering algorithm, optical flow algorithm, and temporal difference method. Video frame selection aims to obtain the most representative video frames. In the present study, we compared the video frame selection algorithm based on human–object interaction with the above classical algorithm. We used a prediction of correct keyframes (PCK) indicator; PCK is the proportion of correctly estimated video frames within the keyframes. In the algorithm, we set n to 29 and randomly took 140 videos as our detection videos, wherein the total number of video frames and the number of keyframes are average. The total number of frames is 5350.

Table 3. Comparison between industrial anomaly detection methods.

AUROC/Method		MemSeg [51]	FastFlow [52]	Ours
Metal Nut	Pixel	99.3	97.9	96.5
	Image	100	100	98.3
Hazelnut	Pixel	97.8	98.1	98.8
	Image	100	100	99.9
Pill	Pixel	99.5	99.2	97.5
	Image	99.3	99.6	99.8
Toothbrush	Pixel	99.2	97.9	96.8
	Image	99.7	99.3	99.8
Bottle	Pixel	99.1	98.2	96.3
	Image	100	99.6	97.1

Table 4 presents the results. It can be observed from the table that our screening of video frames of destructive action is more effective. The optical flow method performs poorly in experiments; the reason for the poor performance is that the brightness of the opening and closing door changes significantly, due to which the optical flow change when the action occurs is relatively small. Thus, the video frame containing the destruction is not easy to detect. The clustering method has evident differences in the selection effect for different videos, because the optimal threshold selection of different videos is different. After repeating the experiments a few times, we set the threshold to 0.89. Figure 7 presents the experimental results. To detect the broken features, the best keyframe extracted should be the intermediate frame wherein each action occurs. In the experiment, we found that keyframe extraction based on interaction extracts all the keyframes containing related actions and similar actions, which makes the number of keyframe extractions extremely large and scattered, which is not conducive to the detection of broken features. With the addition of motion analysis, the number of keyframes was significantly reduced, and the selected keyframes were concentrated near the intermediate frame in which the action occurs.

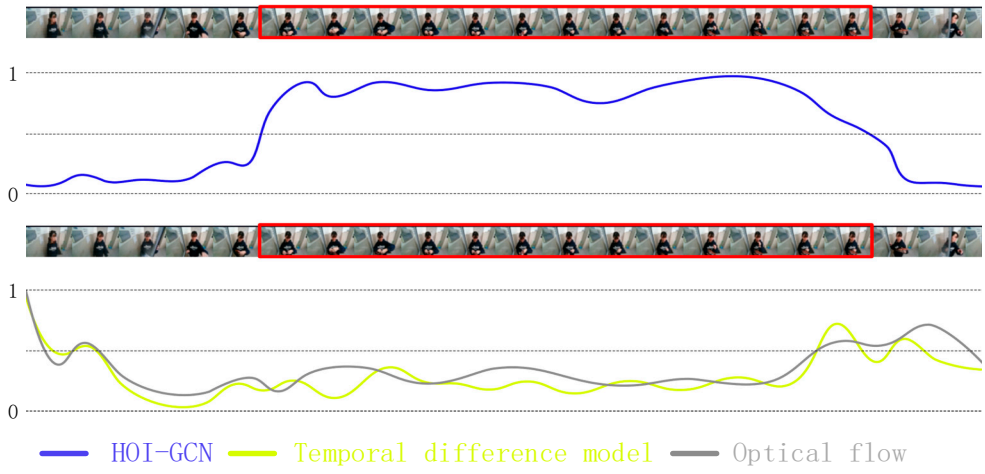


Figure 7. Visual results of video frame selection. The blue curve represents the predicted probability that a video sequence centered on the current frame is predicted to destroy the commodities. The curve below is the score of each video frame as the keyframes. We mark video frames in which anomalous behavior occurs with a red border.

Table 4. Comparison experiment of PCK on various methods.

Method	Key Frames	PCK (%)
Temporal difference method	77	42.1
Optical flow method	297	12.2
Clustering method	336	53.3
Ours	287	82.4

4.3. Ablation Studies

Effectiveness of Confidence-Guided Difference Enhancement Module and Fusion

In the proposed network, one of the core components of the methodology, confidence-guided anomaly enhancement, enhances differences in order to guide the detection of anomalous features. At the same time, the fusion of deep and shallow features provides more comprehensive information for the calculation of anomaly scores. To investigate the effectiveness of confidence-guided anomaly enhancement modules and feature fusions, we study the ablation of six variants: (1) confidence-guided anomalous enhancement modules with shallow features; (2) confidence-guided anomaly enhancement module with deep features; (3) shallow features; (4) deep features; (5) fusion; (6) confidence-guided anomalous enhancement modules with fusion. During the experiment, because the synthesis of the background was unnecessary, we used a U-net segmentation network to separate the consumer's upper body and the commodity, and the remaining space was filled with black.

We compared our algorithm with two methods. Compared with the uncertainty of the last layer of the segmented network, we obtained better results for deep and shallow feature fusion. In the decoding process, when the decoder map of the human body area is compared with the original map, differences can be observed in some nodes, especially after the appearance of the damaged area. The perception difference often expands the size of the abnormal area during the detection process, and thus the accuracy of anomaly detection is reduced. We can converge the detection range of the perception difference by fusing shallow features, and the damaged area can be delineated and distinguished more accurately. Figure 8 displays the experimental figure. Evidently, the result map is more accurate for the boundary definition of some anomalous areas relative to the deep difference, especially for anomaly detection in commodity areas.

In the perceptual difference calculation, we assigned different weights to the perceived loss function of the network in different dimensions of the five layers of the VGG network. We found that although the perceived loss of the fifth layer is theoretically the best, experimentally, the perceived loss of the fourth layer has greater accuracy for the division of the image damage area. Therefore, the order of our assignment of weights from the largest to smallest is as follows: fourth layer, fifth layer, third layer, second layer, and first layer. To verify the effectiveness of the fusion strategy, we detected the deep and shallow features individually and compared their effects after fusion. Table 5 presents the algorithm results for the modules.

Table 5. Ablation studies of the confidence-guided difference enhancement modules and the fusion of multi-layered differences. C stands for confidence-guided difference enhancement modules, S for shallow difference, and F for deep difference.

C	S	F	AUROC
	✓		35.9
		✓	37.2
✓	✓		40.3
✓		✓	42.7
	✓	✓	71.8
✓	✓	✓	82.8

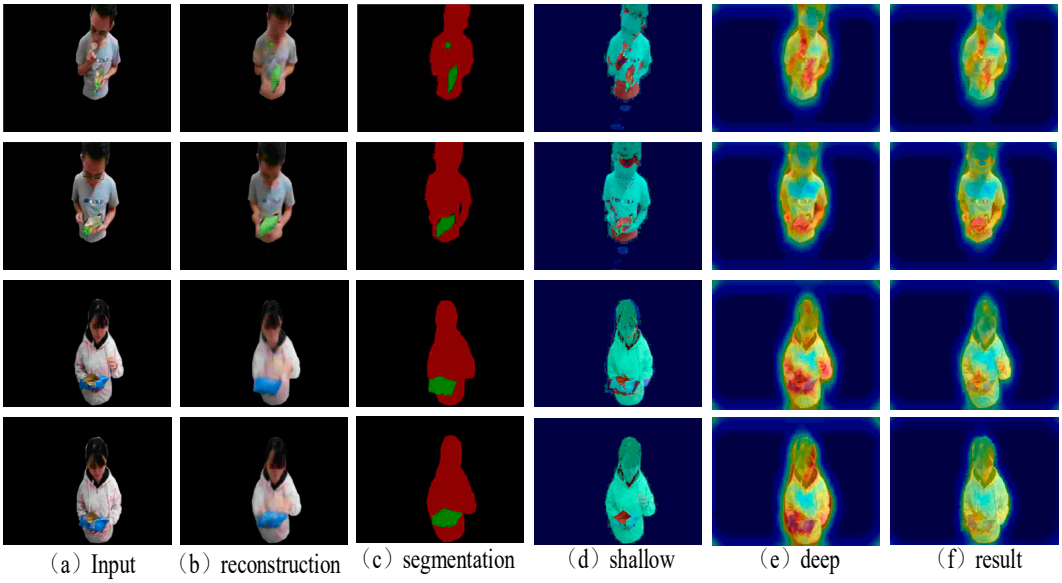


Figure 8. Visual results of ablation study, including the maps processed by the autoencoder network, the anomaly graph computed by the two functions, and the anomaly graph after fusion. After setting the parameters, the color of the abnormal area is red. (**a**) Input images. (**b**) Image after autoencoder. (**c**) The label graph obtained using semantic segmentation. (**d**) The shallow difference calculated using the cosine distance. (**e**) The deep difference calculated using perceptual losses (**f**) The fusion result.

4.4. Analysis of Results

Image regeneration inevitably produces errors, and undamaged commodities may have damaged features due to errors in rebuilding features. We use different fusions λ_S and λ_D in three ratios (0.3, 0.6; 0.2, 0.7; and 0.1, 0.8) because the difference in network synthesis is evident for shallow features, and its difference is small for deep features. Therefore, we randomly assigned the above three proportions to each image. Figure 9 presents the anomaly detection for normal images. The fusion of visible deep and shallow features effectively suppresses the difference generated by synthesis in the commodity area.

The changes in and deformation of objects may generate abnormal features when no destructive action occurs. This has an impact on the abnormal detection. However, as shown in Figure 9, we caused changes in and the deformation of the object through flipping and extruding. The wrong delineation of the zone did not occur. Firstly, these characteristics are present in normal purchases, due to the inevitable changes in and deformation of objects when grasping. As a result, the confidence of the features encoded by such objects is very high throughout the FC layer, that is, the memory module contains similar features. The network reconstructs in the direction most similar to the input feature. Secondly, the detection of anomalies caused by perceptual differences occurs in the features after convolution. The comparison of such deep features is based on whether the features belong to the category of the region in which the segmentation map is located, as has been shown in Figure 8. When the shallow comparison module compares the original image with the restricted image at the pixel level, the detected anomalies only detect the area with a very large difference. For the image reconstructed in the most similar direction, the shallow comparison module cannot play a large role. Therefore, changes in and the deformation of objects will not affect the delineation of abnormal regions.

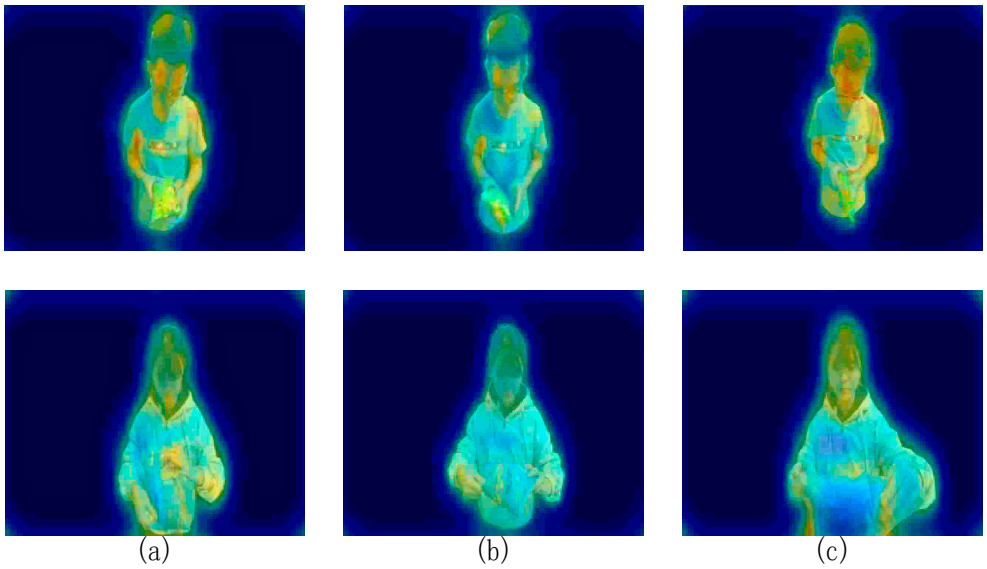


Figure 9. Normal video frame detection output sample. The above image is generated by the three different weights we use: (**a**) $\lambda_S = 0.3$, $\lambda_D = 0.6$. (**b**) $\lambda_S = 0.2$, $\lambda_D = 0.7$. (**c**) $\lambda_S = 0.1$, $\lambda_D = 0.8$.

5. Discussion

In this paper, we propose a method of behavior recognition based on customer behavior and commodity information, which is quite different from methods in previous works focused on SKUs. Because of the small size of the smart retail container, we set the consumption object as a single customer. This makes our algorithm unsuitable for cases of group consumption such as unmanned convenience stores. The abnormal behavior detection algorithm based on damage features consists of two parts. We first identify the actions in the video in order to achieve action recognition in the video. In addition to the usual two actions of either buying or putting the undamaged commodity back, we also defined the behaviors of tearing up packaging to take food and unscrewing the lid of a bottle. If we detected the customer destroying the commodities, we extracted the key frames of this event, detected the damaged part of the commodity, and delineated the abnormal area.

The proposed action recognition strategy combines interaction features and temporal features for consumer behavior recognition, and has excellent detection accuracy and generalization ability. Our action recognition strategy not only considers the temporal characteristics of the action, but also considers the interaction characteristics in a single video frame. The results show that the action recognition module designed using an interactive graph is superior to alternative action recognition algorithms, and the selected video frames show obvious damage characteristics. The ROC curve illustrates the good generalization ability of our algorithm.

The confidence-guided anomaly enhancement module can greatly increase abnormal features and reconstruct normal features well. Compared with other anomaly detection networks based on autoencoder network, our network has added a confidence-guided difference enhancement module and a comparison module. Ablation experiments have proved that the additions of our module are effective. Our strategy of combining a shallow feature comparison network of motion information with the fusion of deep and shallow features can alleviate false alarms caused by the reconstruction process. Based on our results, we can conclude that anomalous features can be precisely divided. Finally, based on the presence or absence of abnormal features of the product, we can accurately identify abnormal behavior.

Author Contributions: Conceptualization, B.L. and K.X.; methodology, B.L.; software, B.L. and X.Z.; validation, B.L., X.Z. and M.C.; formal analysis, C.W.; investigation, J.H.; resources, B.L.; data curation, B.L. and K.X.; writing—original draft preparation, B.L.; writing—review and editing, X.Z.; visualization, K.X.; supervision, K.X.; project administration, W.Z.; funding acquisition, K.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 62272485), the Natural Science Foundation of Xinjiang Uygur Autonomous Region (Grant No. 2020DO1A131), the Teaching Research Fund of Yangtze University (Grant No. JY2020101), and the Undergraduate Training Programs for Innovation and Entrepreneurship of Yangtze University under Grant No. Yz2021040.

Data Availability Statement: Data are unavailable due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, D.H.; Lee, S.; Jeon, J.; Song, B.C. Real-time purchase behavior recognition system based on deep learning-based object detection and tracking for an unmanned product cabinet. *Expert Syst. Appl.* **2020**, *143*, 113063. [CrossRef]
2. Liu, L.; Cui, J.; Huan, Y.; Zou, Z.; Hu, X.; Zheng, L. A Design of Smart Unmanned Vending Machine for New Retail Based on Binocular Camera and Machine Vision. *IEEE Consum. Electron. Mag.* **2021**, *11*, 21–31. [CrossRef]
3. Ramzan, A.; Rehman, S.; Perwaiz, A. RFID technology: Beyond cash-based methods in vending machine. In Proceedings of the 2017 2nd International Conference on Control and Robotics Engineering (ICCRE), Bangkok, Thailand, 1–3 April 2017; IEEE: Piscataway, NJ, USA; pp. 189–193.
4. Zhang, H.; Li, D.; Ji, Y.; Zhou, H.; Wu, W. Deep learning-based beverage recognition for unmanned vending machines: An empirical study. In Proceedings of the 2019 IEEE 17th International Conference on Industrial Informatics (INDIN), Helsinki, Finland, 23–25 July 2019; IEEE: Piscataway, NJ, USA; Volume 1, pp. 1464–1467.
5. Liu, C.; Da, Z.; Liang, Y.; Xue, Y.; Zhao, G.; Qian, X. Product Recognition for Unmanned Vending Machines. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–14. [CrossRef] [PubMed]
6. Lu, Y.; Yu, F.; Reddy, M.K.K.; Wang, Y. Few-shot scene-adaptive anomaly detection. In Proceedings of the Computer Vision–ECCV 2020: 16th Europe-an Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part V 16. Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 125–141.
7. Yao, Y.; Wang, X.; Xu, M.; Pu, Z.; Atkins, E.; Crandall, D. When, where, and what? a new dataset for anomaly detection in driving videos. *arXiv* **2020**, arXiv:2004.03044.
8. Yao, Y.; Xu, M.; Wang, Y.; Crandall, D.J.; Atkins, E.M. Unsupervised traffic accident detection in first-person videos. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 273–280.
9. Kendall, A.; Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? *Adv. Neural-Form. Process. Syst.* **2017**, *30*.
10. Li, Y.; Xu, C.; Han, J.; An, Z.; Wang, D.; Ma, H.; Liu, C. MHAU-Net: Skin Lesion Segmentation Based on Multi-Scale Hybrid Residual Attention Network. *Sensors* **2022**, *22*, 8701. [CrossRef] [PubMed]
11. Shvetsova, N.; Bakker, B.; Fedulova, I.; Schulz, H.; Dylov, D.V. Anomaly detection in medical imaging with deep perceptual autoencoders. *IEEE Access* **2021**, *9*, 118571–118583. [CrossRef]
12. Zenati, H.; Foo, C.S.; Lecouat, B.; Manek, G.; Chandrasekhar, V.R. Efficient gan-based anomaly detection. *arXiv* **2018**, arXiv:1802.06222.
13. Donahue, J.; Krähenbühl, P.; Darrell, T. Adversarial feature learning. *arXiv* **2016**, arXiv:1605.09782.
14. Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–27 October 2017; pp. 2223–2232.
15. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein Generative Adversarial Networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 214–223.
16. Deng, H.; Li, X. Anomaly Detection via Reverse Distillation from One-Class Embedding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 9737–9746.
17. Piergiovanni, A.J.; Kuo, W.; Angelova, A. Rethinking Video ViTs: Sparse Video Tubes for Joint Image and Video Learning. *arXiv* **2022**, arXiv:2212.03229.
18. Liao, Y.; Liu, S.; Wang, F.; Chen, Y.; Qian, C.; Feng, J. PPDM: Parallel Point Detection and Matching for Real-Time Human-Object Interaction Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020.
19. Zhong, X.; Ding, C.; Qu, X.; Tao, D. Polysemy Deciphering Network for Robust Human–Object Interaction Detection. *Int. J. Comput. Vis.* **2021**, *129*, 1910–1929. [CrossRef]

20. Hou, Z.; Yu, B.; Qiao, Y.; Peng, X.; Tao, D. Affordance transfer learning for human-object interaction detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 495–504.
21. Materzynska, J.; Xiao, T.; Herzig, R.; Xu, H.; Wang, X.; Darrell, T. Something-Else: Compositional Action Recognition with Spatial-Temporal Interaction Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020.
22. Wang, T.; Yang, T.; Danelljan, M.; Khan, F.S.; Zhang, X.; Sun, J. Learning human-object interaction detection using interaction points. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 4116–4125.
23. Wang, X.; Gupta, A. Videos as space-time region graphs. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 399–417.
24. Liu, Z.; Nie, Y.; Long, C.; Zhang, Q.; Li, G. A Hybrid Video Anomaly Detection Framework via Memory-Augmented Flow Reconstruction and Flow-Guided Frame Prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021.
25. Morais, R.; Le, V.; Tran, T.; Saha, B.; Mansour, M.; Venkatesh, S. Learning Regularity in Skeleton Trajectories for Anomaly Detection in Videos. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; IEEE: Piscataway, NJ, USA, 2019.
26. Ristea, N.C.; Madan, N.; Ionescu, R.T.; Nasrollahi, K.; Khan, F.S.; Moeslund, T.B.; Shah, M. Self-supervised predictive convolutional attentive block for anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 13576–13586.
27. Feng, J.C.; Hong, F.T.; Zheng, W.S. Mist: Multiple instance self-training framework for video anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14009–14018.
28. Schlegl, T.; Seeböck, P.; Waldstein, S.M.; Schmidt-Erfurth, U.; Langs, G. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *Information Processing in Medical Imaging, Proceedings of the 25th International Conference, IPMI 2017, Boone, NC, USA, 25–30 June 2017*; Springer International Publishing: Cham, Switzerland, 2017; pp. 146–157.
29. Xia, Y.; Zhang, Y.; Liu, F.; Shen, W.; Yuille, A.L. Synthesize then compare: Detecting failures and anomalies for semantic segmentation. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020; pp. 145–161.
30. Salehi, M.; Sadjadi, N.; Baselizadeh, S.; Rohban, M.H.; Rabiee, H.R. Multiresolution knowledge distillation for anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14902–14912.
31. Zhou, H.; Yu, J.; Yang, W. Dual Memory Units with Uncertainty Regulation for Weakly Supervised Video Anomaly Detection. *arXiv* **2023**, arXiv:2302.05160.
32. Bae, J.; Lee, J.H.; Kim, S. Image Anomaly Detection and Localization with Position and Neighborhood Information. *arXiv* **2022**, arXiv:2211.12634.
33. Kim, D.; Park, C.; Cho, S.; Lee, S. Fapm: Fast adaptive patch memory for real-time industrial anomaly detection. In Proceedings of the ICASSP 2023—2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; IEEE: Piscataway, NJ, USA; pp. 1–5.
34. Yan, S.; Xiong, Y.; Lin, D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LO, USA, 2–7 February 2018.
35. Li, M.; Chen, S.; Chen, X.; Zhang, Y.; Wang, Y.; Tian, Q. Actional-Structural Graph Convolutional Networks for Skeleton-based Action Recognition. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; IEEE: Piscataway, NJ, USA, 2019.
36. Zhang, X.; Xu, C.; Tao, D. Context Aware Graph Convolution for Skeleton-Based Action Recognition. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020.
37. Cheng, K.; Zhang, Y.; He, X.; Chen, W.; Cheng, J.; Lu, H. Skeleton-Based Action Recognition with Shift Graph Convolutional Network. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020.
38. Carreira, J.; Zisserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–27 July 2017; IEEE: Piscataway, NJ, USA, 2017.
39. Liu, T.; Ma, Y.; Yang, W.; Ji, W.; Wang, R.; Jiang, P. Spatial-temporal interaction learning based two-stream network for action recognition. *Inf. Sci.* **2022**, *606*, 864–876. [CrossRef]
40. Lin, J.; Gan, C.; Han, S. Tsm: Temporal shift module for efficient video understanding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7083–7093.
41. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. *arXiv* **2016**. [CrossRef]
42. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

43. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
44. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
45. Li, Y.L.; Zhou, S.; Huang, X.; Xu, L.; Ma, Z.; Fang, H.S.; Wang, Y.; Lu, C. Transferable interactiveness knowledge for human-object interaction detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3585–3594.
46. Gkioxari, G.; Girshick, R.; Dollár, P.; He, K. Detecting and recognizing human-object interactions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8359–8367.
47. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015, Proceedings of the 18th International Conference, Munich, Germany, 5–9 October 2015*; Part III 18; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
48. Park, H.; Noh, J.; Ham, B. Learning memory-guided normality for anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 14372–14381.
49. Bertasius, G.; Wang, H.; Torresani, L. Is space-time attention all you need for video understanding? In Proceedings of the 2021 International Conference on Machine Learning, Virtual, 18–24 July 2021; Volume 2, p. 4.
50. Liu, Z.; Zhang, H.; Chen, Z.; Wang, Z.; Ouyang, W. Disentangling and Unifying Graph Convolutions for Skeleton-Based Action Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020.
51. Yang, M.; Wu, P.; Feng, H. MemSeg: A semi-supervised method for image surface defect detection using differences and commonalities. *Eng. Appl. Artif. Intell.* **2023**, *119*, 105835. [CrossRef]
52. Yu, J.; Zheng, Y.; Wang, X.; Li, W.; Wu, Y.; Zhao, R.; Wu, L. FastFlow: Unsupervised Anomaly Detection and Localization via 2D Normalizing Flows. *arXiv* **2021**, arXiv:2111.07677.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

E-ReInForMIF Routing Algorithm Based on Energy Selection and Erasure Code Tolerance Machine

Qiong Wu [1,2], Hai Huang [1], Xinmiao Lu [1,*], Jiaxing Qu [2], Juntao Gu [2] and Cunfang Yang [1]

[1] Harbin University of Science and Technology, Harbin 150080, China
[2] Heilongjiang Network Space Research Center, Harbin 150090, China
* Correspondence: lvxinmiao0611@hrbust.edu.cn; Tel.: +86-187-0462-8841

Abstract: Aiming at the problems of data loss and uneven energy consumption in wireless sensor networks during data transmission, this paper proposes a ReInForM transmission fault-tolerant routing algorithm based on energy selection and erasure code fault-tolerant machines (E-ReInForMIF). The E-ReInForMIF algorithm improves the multi-path routing algorithm by combining an erasure coding fault-tolerant machine and node residual energy sorting selection. First, the erasure coding fault-tolerant machine is used to encode the signal, determine the number of transmission paths through multi-path routing, and then select the specific node of the next hop by sorting the residual energy of the node. The E-ReInForMIF routing algorithm effectively solves the problems of uneven energy consumption and data loss in data transmission, improving network lifespan and transmission reliability. Finally, the signal is decoded. The simulation results show that the E-ReInForMIF routing algorithm is superior to the ReInForM routing algorithm in improving the reliability of data transmission.

Keywords: ReInForM routing protocol; energy selection; erasure correcting coding fault-tolerant machine; node residual energy ranking

1. Introduction

In recent years, with the extensive use of WSN, the requirements for data transmission reliability have been gradually improved, especially, in the fields of national defense and the military [1], environmental monitoring [2], space exploration [3], etc. When transmitting data, the wireless link is unreliable causing severe packet loss and low transmission reliability. In addition, each node consumes different amounts of energy, and cluster head nodes consume higher energy, making them more likely to fail due to insufficient energy, resulting in network paralysis. Therefore, it is important to address the issues of high energy consumption and low reliability in transmission. The reliability of transmission is mainly reflected in packet acceptance rate and packet accuracy rate. At present, aiming at the low reliability of wireless sensor networks, the packet loss rate of inter-node links is reduced by retransmission [4] and the packet acceptance rate and accuracy [5]. The main solutions to the problems of high energy consumption among nodes, uneven energy consumption among nodes [6], and easier failure of cluster head nodes are through energy balance [7].

Xu Jiuqiang et al. introduced the energy balancing method EBETX [8], which uses the ETX value as a routing metric while taking into account the energy consumption of the nodes in the network. This ensures balanced energy consumption among nodes in the network, improving node lifetime, extending network lifetime, and effectively solving the problem of node energy constraints. However, data are easily lost during transmission and the reliability is not high. The scientific literature [9] introduces ReInForM routing into wireless sensor networks. The ReInForM routing algorithm establishes a corresponding path based on the number of observation matrices required for reconstructing the signal.

Experimental results show that this algorithm has high reliability and ensures the effective operation of the network. Wang Dengdi combined the energy balancing method with ReInForM routing and proposed a ReInForM routing algorithm based on energy selection [10]. This algorithm ranks the nodes to be selected in the neighbor node set according to their energy consumption, preferentially selecting nodes that consume less energy and have more residual energy, which effectively solves the energy balancing problem in data transmission. In addition, the adoption of retransmission improves reliability to a certain extent. However, the ReInForMIF algorithm still suffers from data loss and its improved reliability is not high. Adding FEC code to the ReInForM routing algorithm improves the receiving rate of packets but the accuracy of packets is not high. The advantages and disadvantages of the algorithm are summarized in Table 1.

Table 1. Advantages and disadvantages of the method.

Method	Advantage	Disadvantage
Energy balancing method	To balance the energy loss of nodes and extend the network life.	Data are easy to lose; reliability is not high.
ReInForM routing algorithm	High reliability to ensure the effective work of the network.	Uneven energy consumption of each node easily causes network breakdown.
ReInForM routing algorithm based on energy selection	Low energy loss and reliability.	The problem of data loss; the packet acceptance rate is low.
ReInForM Routing Algorithm with FEC Encoding	Improved packet reception rate and reduced energy consumption.	The accuracy of data packets is not high.

To solve the problem with the above algorithm, this paper proposes a new routing algorithm: a ReInForM transmission fault-tolerant routing algorithm based on energy selection. The E-ReInForMIF algorithm evaluates the energy consumption of nodes in the path and adds an erasure coding fault-tolerant mechanism [11]. Before sending the data, the node-processing packet is made into a data slice using the erasure coding fault-tolerant mechanism. At the same time, when the sending node selects the next hop node, it sorts the energy of neighbor nodes and determines the next hop node according to the node's residual energy [12] and prioritizes selecting nodes with less energy consumption and more residual energy. After reaching the node, the data slice is decoded and restored it to the original data packet. During the whole transmission process, a certain number of data packets are allowed to be lost. The next hop is selected by the method of node residual energy ranking, which effectively reduces energy consumption and prolongs the working life of the network. The addition of erasure coding fault-tolerant machines reduces the packet loss rate in data transmission and improves the accuracy of data transmission. The E-ReInForMIF algorithm ensures the accuracy of data transmission, balances the energy of nodes in the network, and improves the reliability of data transmission.

2. E-ReInForMIF Algorithm Principle

Nodes on the ReInForM routing protocol's default data packet transmission path know the error rate e of each path. While the receiving node completes data collection, it analyzes the energy cost required to transmit data, estimates the transmission reliability rate r based on the path energy consumption, and puts r into the data packet being transmitted. If the source node jumps at least h times to reach the sink node, the probability of successful data transmission using a single path is $(1-e)^h$. If $(1-e)^h < r$, the whole transmission does not reach the expected reliability.

In the ReInForM algorithm, the source node makes multiple copies of the data packets that need to be transmitted and transmits these data packets over randomly selected paths. Djukic et al. proposed forward error correction technology [13] (forward error correction, FEC). Based on the load balancing principle, the FEC copies source data packets into multiple copies and creates multiple disjoint paths. The use of FEC technology can reduce the data loss rate and ensure the integrity of the data received by the destination node. S. Kim et al. combined multiple transmission fault-tolerant methods such as retransmission [14] and erasure coding (EC) to improve the reliability of data transmission. However, the overall energy consumption is high, whereas the data accuracy is not high. This paper improves the ReInForM algorithm, quantifies the energy consumption of nodes in the path, and adds an EC fault-tolerance mechanism. The ReInForM transmission fault-tolerant routing algorithm based on energy selection is proposed to ensure accurate transmission of data packets and reduce energy consumption.

The EC fault-tolerant mechanism is adopted at the source node before data transmission. EC can obtain higher data reliability with smaller data redundancy and improve the fault tolerance of data transmission. In the process of data transmission, multi-path transmission [15] is combined with a fault-tolerant mechanism and node energy consumption evaluation mechanism to improve the reliability of data transmission and improve the data transmission rate.

2.1. EC Fault-Tolerant Machine Based on the Quantitative Evaluation Mechanism of Node Energy Consumption

In practical applications, data can be lost during transmission due to various network problems. An EC fault-tolerant mechanism effectively avoids the problem of data packet loss. The specific process is as follows: the data is first encoded during transmission to form multiple data slices, and then these data slices are sent through multiple paths. When the destination node is reached, the data piece is decoded at the destination node, and the data is sorted into source data packets [16]. In the whole mechanism, the accuracy of data transmission is guaranteed within the scope of allowing data packet loss. An EC fault-tolerant machine is shown in Figure 1.

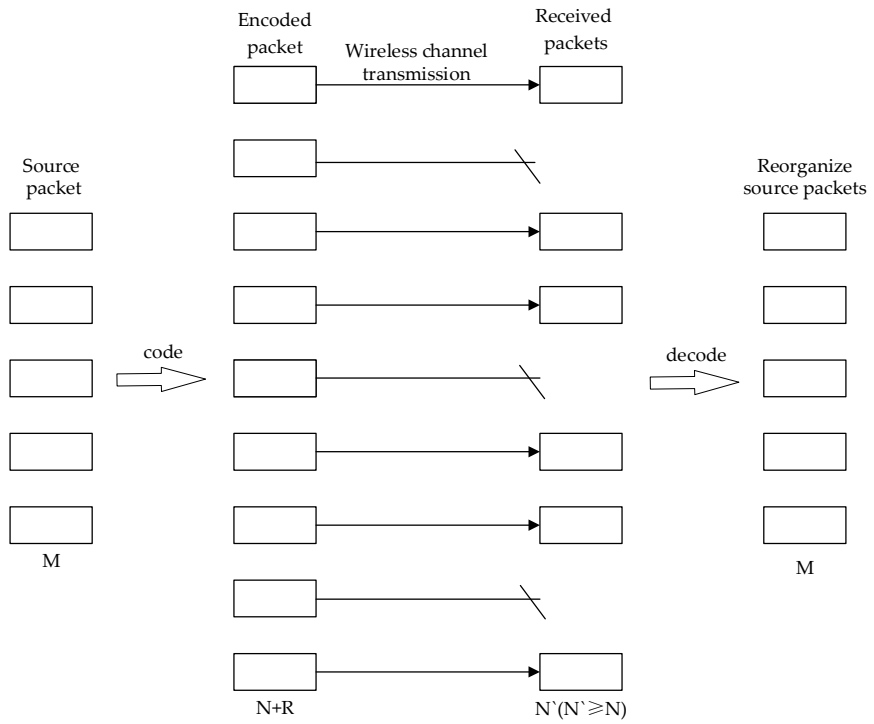


Figure 1. Erasure coding fault-tolerant machine.

In the EC fault-tolerant mechanism, the sending node decomposes the data packet of bM bytes into M data slices of b bytes and encodes them into $N + R$ data slices, where R is the redundancy of data coding. In WSN, the encoded data slice is transmitted through the source node and sent through n paths x_1 to x_n, where the path and encoded data slice need to meet $\sum_{i=1}^{n} x_i = N + R$. The destination node will receive $N'(N \leq N' \leq N + R)$ encoded data slices. In the transmission process, it should be noted that EC allows the loss of data slices not greater than R. When R is greater than the data transmission loss rate, the destination node can use the redundant transmission to recover the source data packets [17]. The destination node needs to receive no less than N encoded data slices to decode and recover M source data. If Z_i random data pieces are received on path x_i, it needs to meet $\sum_{i=1}^{n} Z_i \geq N$.

The EC fault-tolerant machine will consume node energy while ensuring the accuracy of data transmission. Therefore, a quantitative evaluation mechanism for node energy consumption is introduced to balance the energy consumption of the entire network nodes.

By estimating the distance between the surrounding nodes and the sink node, as well as the amount of energy required in the data transmission process and sorting adjacent nodes, the sending node selects the most suitable next-hop node among the neighbor nodes to ensure the reliability of data transmission, balance the network energy consumption, and improve the quality of data transmission [17]. The parameters required for node energy consumption are defined as follows:

(1) $EC(N_j)$ is the energy consumption from node N_j to the sink node. The path from N_j to the sink node is estimated according to the minimum number of jumps.

First, node A sends the data packet to B, and node B continues to forward the data packet to C. At this time, the two transmissions consume the same energy, that is, $e_{AB} \approx e_{BC}$. The energy consumption of nodes is only related to the transmission of data. When transmitting the same data packet, the energy consumption is approximately equal. Secondly, there are many transmission paths for node A to reach the sink node through node B. Therefore, the energy required in the node A to sink node path is equal to the product of e_{AB} and the minimum number of hops h of the node in the transmission data path is $e_{AB} \times h$. At this time, the energy consumption of node A to the sink node can be calculated. The path that requires the least energy consumption is selected for each jump. In the process of data transmission, ensure that each step of transmission consumes the least energy, so that the network can be fully utilized and the service life of the network can also be extended.

Estimating the minimum hop times of a node can improve the sensitivity of the network to path changes, so $EC(N_j) \approx e_{jx} \times h_j$(node x is the next hop neighbor node of node j and h_j is the minimum number of hops from node j to the sink node. Because the energy consumption of the set node is only related to the transmission of data, node x is any neighbor node of node j).

(2) $EM(N_i, N_j)$ is the transmission of energy consumption from the node N_i to the neighbor node N_j.

$$EM(N_i, N_j) = e_{ij}^{\alpha}\left(1 - \frac{\left(R_i^{\beta} - R_{ij}^{\alpha}\right)}{R}\right) = e_{ij}\lambda \tag{1}$$

The energy consumption of data transmission N_i through N_j and the remaining energy of the forwarding node are comprehensively analyzed. Where e_{ij}^{α} is the energy consumption value of the node N_i transmitting data directly to N_j, $e_{ij}^{\alpha} = v_t \times \mu$ (v_t is the number of data packets sent per unit time, and μ is the energy consumption value of forwarding unit data packet). R is the initial energy of the node, R_i^{β} is the residual energy of the node N_i, and λ is the energy consumption impact factor. The greater the λ, the greater the node energy consumption and the smaller the residual energy. α and β are the adjustment coefficients, set to 1.

(3) $C_{N_iN_j}$ is the overall energy consumption of the node N_i sending data to the sink node through N_j.

$$C_{N_iN_j} = EC(N_j) + EM(N_i, N_j) \tag{2}$$

$C_{N_iN_j}$ consists of $EC(N_j)$ and $EM(N_i, N_j)$. Consider the energy consumption of data transmission as a whole and avoid only considering the minimum energy consumption of one hop, resulting in ignoring the overall energy consumption of jumping to the sink node.

The energy consumption of each neighbor node for transmitting data packets is compared with the energy consumption of hops from the source node to the sink node. They are classified according to the comparison between the former and the latter; they are classified into three categories: less 1, the same, and more 1, and the corresponding combination is H^-, H^0, and H^+. Apply $C_{N_iN_j}$ to each neighbor node set H^-, H^0, H^+. Each node in its own set is sorted incrementally according to $C_{N_iN_j}$, as shown in Figure 2.

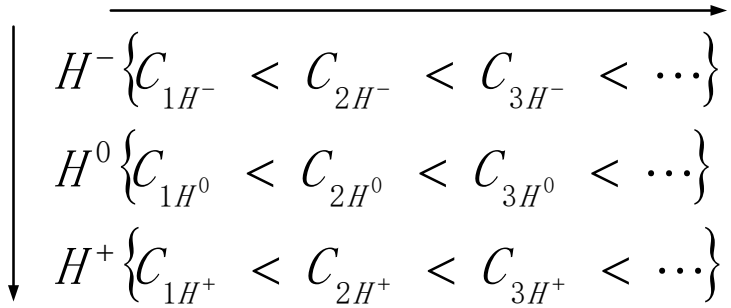


Figure 2. Energy ranking of nodes in each neighbor set.

2.2. Residual Energy of Nodes

Select the next hop node according to the energy order of the node. First, in the H^- node set, select from the smallest to the largest according to the node energy consumption ranking, then select from the H^0 node set, and finally select from the H^+ node set until all nodes on the path have been selected. If there are vacant locations, start with the H^- node set collection and continue the above operations until the nodes on the determined path are fully deployed. The selection of residual energy nodes is shown in Figure 3. The points on the set belong to the set represented by the circle.

After arranging the residual energy of each node set in the path from high to low, and according to the rule of preferentially selecting nodes with high residual energy, we complete the collection of the next hop node and additional nodes. The residual energy sets H^-, H^0, and H^+ of a node comply with this rule. After selecting the node with the highest residual energy, these nodes are placed in the set of next-hop nodes until the number of next-hop nodes is satisfied. If the total number of next-hop node sets for sending node S is 4, there are 3 neighbor nodes and then the number of neighbor nodes for node S is 12 $V1, V2, \ldots, V12$. According to this principle, continue to select two node sets, as well as a set of nodes that are not within the selected range. As shown in Figure 3, $\{V1, V2, V3\}$ is the H^- set, $\{V4, V5, V6\}$ is the H^0 set, $\{V7, V8, V9, V10\}$ is the H^+ set, and $\{V11, V12\}$ is outside the selected range. Compare the residual energy of the set $\{V1, V2, V3\}$; if the node $V1$ has the highest residual energy, consider the node $V1$ as the next hop node. The residual nodes $\{V2, V3\}$ in the set H^- continue to be selected according to this rule. Finally, select $V2$, $V3$, and $V4$ to join the next hop node set. If the data does not reach the sink node and the node cannot find the next-hop node of the next-hop node, the path is rebuilt at the next-hop node, and the residual energy of the nodes is ordered. The residual energy of the node is sorted. Repeat the above operation until the data is transmitted to the sink node.

In this process, the node selects the node set with the highest residual energy during each hop, ensuring the user's expected reliable value for data transmission. Therefore, while improving the reliability of data transmission, the improved algorithm balances the use of energy and the residual amount in the network, prolonging the usage time of the network.

The traditional ReInForM routing algorithm does not analyze and compare the residual energy of the next hop node, resulting in poor overall transmission reliability and unbalanced network energy consumption. The improved algorithm not only improves the data coding and ensures the integrity of data transmission but also analyzes and compares the energy consumption of nodes in the path and the next hop, reducing the number of failed nodes and extending the life cycle of the network.

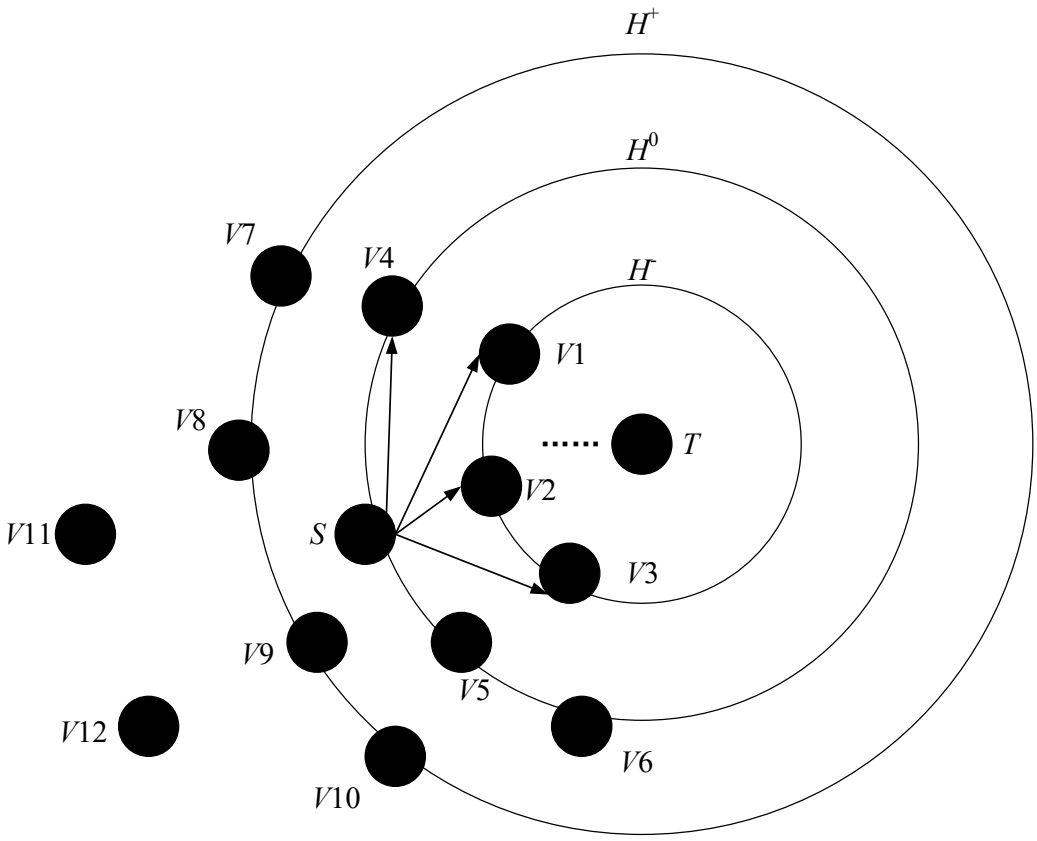


Figure 3. Selection of remaining energy nodes.

3. E-ReInForMIF Algorithm Steps

3.1. E-ReInForMIF Algorithm Flow

This research combines an EC fault-tolerant machine with node residual energy sorting and selection to propose a new routing algorithm, namely the E-ReInForMIF algorithm. The overall process for the E-ReInForMIF algorithm is shown in Figure 4.

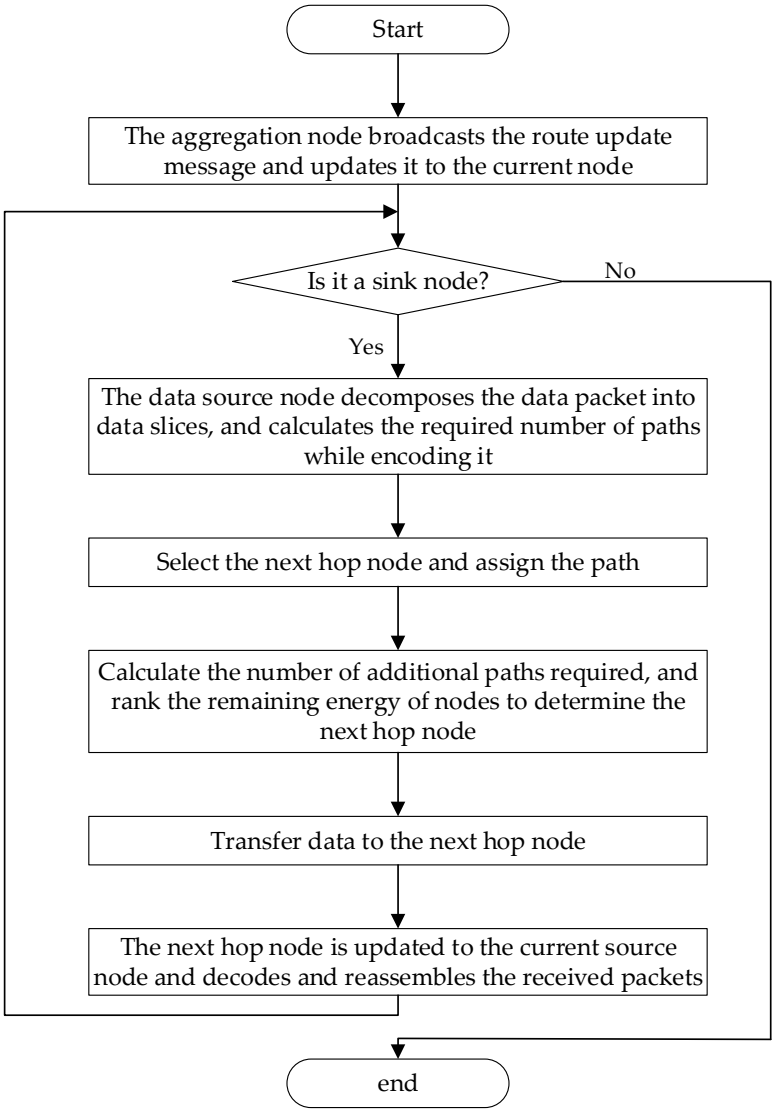


Figure 4. E-ReInForMIF algorithm flow.

3.2. Specific Steps of the E-ReInForMIF Algorithm

(1) The sink node broadcasts the updated content selected by the entire path node to each other node periodically. Each node adjusts its hop count based on the received message and broadcasts the message to other neighboring nodes. In the path, each node knows the jump to the sink node by itself and its surrounding nodes;

(2) The sending node processes the bM byte data packets, arranges them into M data slices of b bytes, and encodes them;

(3) Calculate the transmission path: according to the source node data reliability parameter r_s, channel error rate e_s, and hop count h_s, use the following formula:

$$P(r_s, e_s, h_s) = \frac{\log(1 - r_s)}{\log\left[1 - (1 - e_s)^{h_s}\right]} \tag{3}$$

prepare for reliable data transmission through calculation and analysis;

(4) Compare the energy consumption of each neighbor node transmitting data packets with the energy consumption of the source node to the sink node hop. According to the comparison between the former and the latter, they are classified into three categories: less 1, the same, and more 1. The corresponding combinations are H^-,

H^0, and H^+. The incremental ordering of each aggregation node is determined by the calculation $C_{N_iN_j}$ of each node in the transmission path;

(5) Select the next hop node from three sets according to the data transmission energy consumption requirements. The source node first selects the next hop node at H^-. If $1 - e_s$ is smaller than the calculated number of paths, additional paths need to be added when the source node transmits data. The additional number of paths is:

$$P = \frac{\log(1 - r_s)}{\log\left[1 - (1 - e_s)^{h_s}\right]} - (1 - e_s) \tag{4}$$

After the number of redundant paths is determined, select nodes in the order of H^-, H^0, and H^+ according to the three node sets sorted out previously. When P is greater than the number of nodes included in H^-, select nodes from H^0 and H^+ in turn until the selected number of paths meets the calculated probability value. If the required number cannot be reached, select again from the node set H^- and continue to cycle this operation until the selected number of paths meets the probability requirements. Each selected node continues to create its path number and is added to the additional path value. The calculated total external path number is the total external path number created by all selected nodes. Finally, the node transmits the created path to each array, and then transmits the residual energy value of the node, as well as the encoded data slice, to all neighboring nodes through the created path;

(6) After receiving the data slice, the neighbor node decodes the data; the entire encoding and decoding process is performed using a system encoding method;

(7) At this point, consider the neighbor node as the source node, calculate the number of new transmission paths again, and cycle through the previous operations until the data is transmitted to the sink node.

In the E-ReInForMIF algorithm, the nodes provide the number of data packets lost during transmission. Considering the reliability of the number of paths created, energy surplus and transmission energy consumption of the next hop nodes are analyzed to improve the overall reliability of network data transmission.

4. Experimental Simulation Result

The E-ReInForMIF algorithm takes the number of data packets received by the destination node and the integrity of the received data packets as measurement results, namely, the packet reception rate and accuracy. The higher the reception rate, the smaller the data packet loss rate in transmission. The greater the accuracy, the more complete the data received. The higher the data packet reception rate and accuracy, the higher the reliability of data transmission. This article compares the improved algorithm with three algorithms, which are:

Method 1: Based on the ReInForM algorithm, the data is directly allocated to the transmission path without considering the energy balance of the path nodes;

Method 2: Based on the ReInForM algorithm, distribute source data packets evenly into the transmission path without considering packet coding;

Method 3: Based on the ReInForM algorithm, add FEC coding to provide a certain fault tolerance rate.

The experiment was conducted on matlab2016 b; the network structure of this experiment adopts a multi-path structure, with a network coverage area of 600 m × 600 m that is monitored in real-time. During the entire simulation run of 1000–2000 times, the data packet reception rate and accuracy rate of the above three algorithms were compared with those of the proposed algorithm in this article. It is proved that the proposed algorithm is the best of the four algorithms. The relevant parameters of the wireless sensor network and the performance indicators of the nodes are shown in Table 2.

Table 2. WSNs-related parameter settings.

Parameter Type	Parameter Value
Number of network nodes	100
Node initial energy	9900 J
Node residual energy	9700 J
Transmitter amplifier energy consumption (E_{amp})	500 pj/bit/m^2
Destination node energy consumption (E_{fuse})	8J
Radio energy consumption (E_{elec})	250 J/bit
Forwarding each data packet energy consumption	2 J
Node maximum transmission distance	100 m
Data packet length	128 B
Control packet length	64 bit
Expected reliability	0.9
Channel error rate	0.4

When the network runs from 1000 to 1300 times, the data packet reception rate of the four algorithms is 1. For 1300–2000 runs, the packet reception rate of Method 3 and the proposed algorithm are still above 0.97, which has obvious advantages. For 1600–1800 runs, the data packet reception rate of the proposed algorithm in this article is slightly lower than that of Method 3 but still higher than Method 1 and Method 2. For 1800–2000 runs, the proposed algorithm is again higher than the other algorithms and remains above 0.98, as shown in Figure 5.

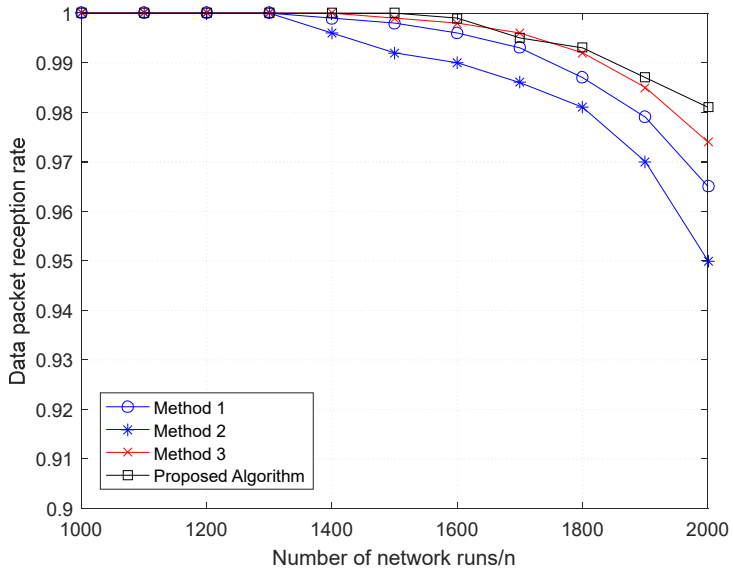


Figure 5. Packet reception rate (energy consumption parameters: E_{fuse}, E_{elec}, E_{amp}).

As can be seen from Figure 5, the greater the number of network operations, the clearer the difference in data packet reception rates among various algorithms. The Figure 6 simulation increases the network operation energy consumption by five times. Method 1 and Method 2 decrease rapidly and the overall data packet reception rate is not high. The difference between Method 3 and the proposed algorithm is not significant, and both have high data packet reception rates. When the network runs 1200–1600 times, the data packet reception rate of Method 3 and the proposed algorithm is almost the same. It can be seen that when the network runs 2000 times, the packet reception rates of Method 3 and

the proposed algorithm remain above 0.97. In other runs, the data packet reception rate of the algorithm proposed in this article is higher than that of the other three algorithms, as shown in Figure 6.

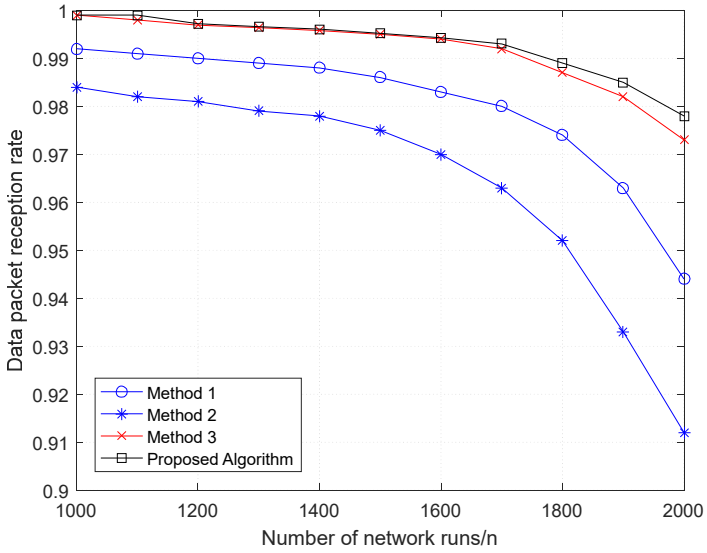


Figure 6. Packet reception rate (energy consumption parameters: $5E_{fuse}$, $5E_{elec}$, $5E_{amp}$).

In the case of low energy consumption parameters, the data packet accuracy of all four methods remains at 1 when the network runs 1000 to 1400 times. When the network runs 1400–2000 times, the data packet accuracy of all four methods has decreased, but the data acceptance rate of the proposed algorithm is always the highest, being maintained above 0.99. When the network runs 1400–1700 times, there is not much difference between the proposed algorithm and Method 3 in this article; however, it is still higher than Method 1 and Method 2. When the network runs 1800–2000 times, the data packet accuracy of the proposed algorithm is significantly higher than that of Method 3 and there is a significant difference between the two. From the simulation results, it can be seen that the data packet accuracy of all four methods remains above 0.97, as shown in Figure 7.

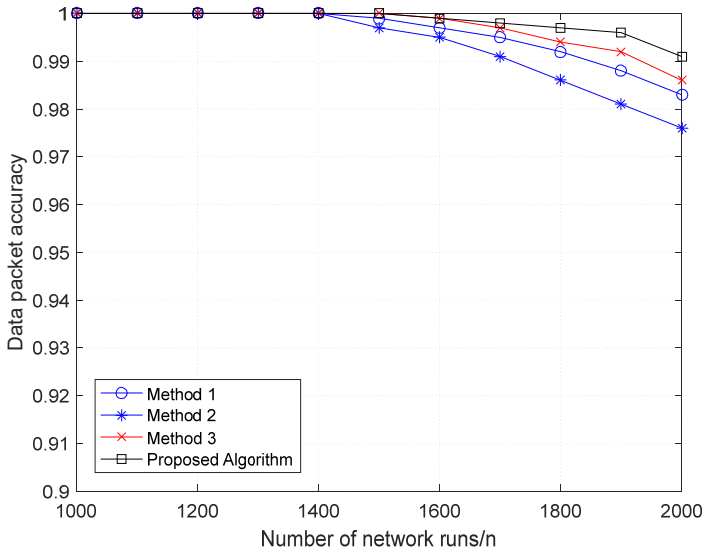


Figure 7. Data accuracy (energy consumption parameters: E_{fuse}, E_{elec}, E_{amp}).

Increasing the energy consumption parameter by five times is different from the simulation results of the data packet acceptance rate, and the difference in data packet accuracy among different algorithms is not significant. Due to the use of an EC encoding

fault-tolerant mechanism, the data accuracy of the proposed algorithm is the highest and the decrease is relatively smooth, with an accuracy rate consistently maintained above 0.985. For Method 3, the addition of FEC encoding has achieved a certain fault tolerance rate, but the fault tolerance rate is lower than the proposed algorithm. Therefore, the data packet accuracy of Method 3 is lower than that of the proposed algorithm but significantly higher than Methods 1 and 2. As shown in Figure 8.

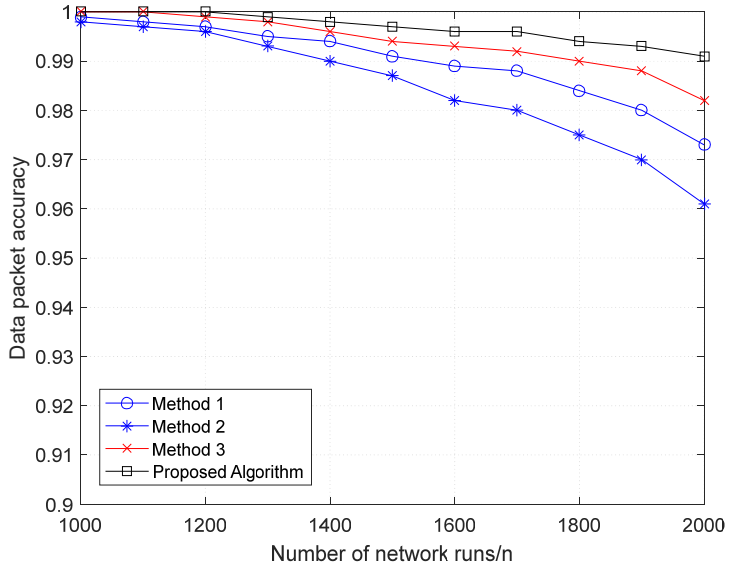


Figure 8. Data accuracy (energy consumption parameters: $5E_{fuse}$, $5E_{elec}$, $5E_{amp}$).

The energy efficiency of the path is calculated by analyzing the energy value required by the destination node to receive data and the energy value of the original node to send data. Method 2 has the lowest energy efficiency. The difference between the proposed algorithm and Method 3 in this paper is not significant when the number of runs on the network is 1000–1700 times. When the network runs 1700 times, the proposed algorithm in this article is slightly lower than Method 3. After running the network 1700–2000 times, it can be seen that the proposed algorithm has the highest energy efficiency. It is kept above 350, as shown in Figure 9.

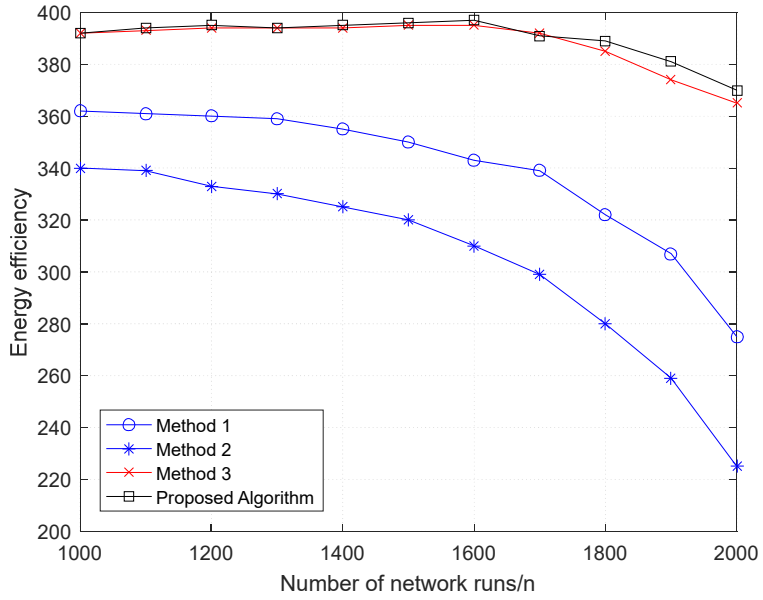


Figure 9. Energy efficiency analysis.

From the above simulation results, it can be seen that the proposed algorithm is significantly superior to Method 1 and Method 2 in terms of data packet reception rate, accuracy, and energy efficiency. Compared with the three-phase method, the algorithm proposed in this paper does not have significant advantages in packet reception rate and energy efficiency. However, with the addition of an EC encoding fault-tolerant mechanism, the proposed method significantly outperforms Method 3 in terms of data packet accuracy. Overall, the proposed algorithm is the best of the four algorithms and is a highly reliable and low-energy-consuming routing algorithm.

5. Conclusions

By analyzing the reliability of data transmission in wireless sensor networks, the performance of wireless sensor networks can be measured. This paper proposes the E-ReInForMIF algorithm, which improves the multipath routing algorithm by combining the EC fault-tolerant machine and node residual energy sorting selection. The encoded data is transmitted through the calculated path. During transmission, the energy consumption and residual energy of the nodes in the transmission path are always considered and a certain amount of data packet loss is provided to ensure the accuracy of data transmission. After transmitting to the destination node, the destination node decodes the encoded data. In the whole transmission process, the method of node residual energy ranking is used to quantitatively evaluate the energy consumption of nodes, so that the whole algorithm can improve the efficiency of data transmission, reduce energy consumption, and improve the reliability of data transmission. This article compares and analyzes the improved algorithm with traditional methods. From the simulation results, it can be seen that the algorithm proposed in this article has higher data packet acceptance and accuracy rates than the other three methods and also has higher energy efficiency than the other three methods. Therefore, the E-ReInForMIF algorithm in this article is better. Due to its excellent reliability advantage, it can be applied to the detection of all toxic gas parameters in public places such as schools and factories. Detection of temperature and pressure in pipelines such as water supply, heating, and oil supply is essential to prevent the occurrence of disasters. In the future, with the deepening of research on wireless sensor networks, they can also be applied to the healthcare industry. Remote detection of patients' conditions will enable them to receive timely treatment even at a distance from the hospital, achieving remote medical control.

Author Contributions: Conceptualization, Q.W.; methodology, Q.W.; software, H.H.; validation, J.Q., Q.W. and J.G.; writing—original draft preparation, X.L. and C.Y.; writing—review and editing, C.Y.; project administration, Q.W.; funding acquisition, X.L. and Q.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by HeiLongJiang Postdoctoral Foundation (Grant number LBH-Z22200) and Heilongjiang Provincial Natural Science Foundation of China (Grant number YQ2022F014).

Acknowledgments: The authors acknowledge HeiLongJiang Postdoctoral Foundation (Grant number LBH-Z22200) and Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ahmed, S.; Ahmed, I.; Walid, O.; Ahmed, M.K. Compressive sensing based secure data aggregation scheme for IoT based WSN applications. *PLoS ONE* **2021**, *16*, 10160–10187.
2. Nouf, A.; Tahani, G. A Novel Security Architecture for WSN-Based Applications in Smart Grid. *Smart Cities* **2022**, *5*, 633–649.
3. Lavanya, S.; Prasanth, A.; Jayachitra, S.; Shenbagarajan, A. A Tuned classification approach for efficient heterogeneous fault diagnosis in IoT-enabled WSN applications. *Measurement* **2021**, *183*, 109771–109787. [CrossRef]
4. Mustafa, A.S.; Ismail, B.; Abdelhafid, A.; Pascal, L. Enhanced efficient outlier detection and classification approach for WSNs. *Simul. Model. Pract. Theory* **2022**, *120*, 102618–102635.

5. He, C.B.; Qin, Q.; Li, M.Z. Variable Factor Based Non Uniform Clustering Routing Protocol for WSNs. *Inf. Comput.* **2022**, *34*, 11–13.
6. Zhao, F.; Wei, K.; Chen, H.B. A data aggregation node selection algorithm based on average residual energy in sensor networks. *Comput. Appl. Res.* **2012**, *29*, 3096–3098.
7. Elsway, A.A.; Khedr, A.M.; Oruba, A.; Walid, O. Energy-aware disjoint dominating sets-based whale optimization algorithm for data collection in WSNs. *J. Supercomput.* **2022**, *79*, 10227–10249. [CrossRef]
8. Xu, J.Q.; Ding, Y.G.; Zhao, H.; Zhang, X.Y.; Bi, F.W. Energy Balanced Reliable Routing Metrics in Wireless Sensor Networks. *Comput. Eng.* **2009**, *35*, 87–89.
9. Zhuang, Y.; Shen, J. ReInForm-CS routing algorithm for CS based wireless sensor networks. *Comput. Syst. Appl.* **2012**, *21*, 140–144.
10. Wang, D.D.; Chai, Q.L.; Cai, X.J. Research and Simulation of ReInForM Routing Algorithm Based on Energy Selection. *Comput. Simul.* **2011**, *28*, 174–177.
11. Zhang, K.L. Research and Implementation of Error Tolerant Mechanism for Erasure Codes in Distributed Systems. *Comput. Mod.* **2012**, *8*, 47–50.
12. Zhang, Y.H.; Yan, L.J. Low energy routing protocol based on the remaining energy of all nodes. *Microcomput. Appl.* **2021**, *37*, 20–23.
13. Pristupa, P.V.; Sushchenko, S.P.; Mikheev, P.A.; Moiseev, A.N. Efficiency Analysis of the Forward Error Correction at the Transport Protocol Level. *Autom. Remote Control* **2022**, *83*, 1059–1077. [CrossRef]
14. Suelen, L.; Ricardo, M.; Carlos, M.; Francisco, V. Combining Network Coding and Retransmission Techniques to Improve the Communication Reliability of Wireless Sensor Network. *Information* **2021**, *12*, 184 . [CrossRef]
15. Sun, C.L.; Wang, Z.Z.; Lu, D.W.; Cao, L.; Yue, Y.G.; Ding, H.H.; Hu, Z.Y. An Energy Efficient and Reliable Multipath Transmission Strategy for Mobile Wireless Sensor Networks. *Comput. Intell. Neurosci.* **2022**, *80*, 3804–3819. [CrossRef] [PubMed]
16. Zhu, B.T.; Ebrahim, B.; Nguyen, H.H.; Robert, B.; Jerome, H. Improved Soft-k-Means Clustering Algorithm for Balancing Energy Consumption in Wireless Sensor Networks. *IEEE Internet Things J.* **2021**, *8*, 4868–4881. [CrossRef]
17. Jiang, Y.; Wu, M. Research on Reliability of Data Transmission Based on Network Coding in WSN. *Comput. Technol. Dev.* **2013**, *23*, 148–151.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Domain-Aware Adaptive Logarithmic Transformation

Xuelai Fang † and Xiangchu Feng *,†

School of Mathemathics and Statistics, Xidian University, Xi'an 710071, China
* Correspondence: xcfeng@mail.xidian.edu.cn
† These authors contributed equally to this work.

Abstract: Tone mapping (TM) aims to display high dynamic range scenes on media with limited visual information reproduction. Logarithmic transformation is a widely used preprocessing method in TM algorithms. However, the conventional logarithmic transformation does not take the difference in image properties into account, nor does it consider tone mapping algorithms, which are designed based on the luminance or gradient-domain features. There will be problems such as oversaturation and loss of details. Based on the analysis of existing preprocessing methods, this paper proposes a domain-aware adaptive logarithmic transformation AdaLogT as a preprocessing method for TM algorithms. We introduce the parameter p and construct different objective functions for different domains TM algorithms to determine the optimal parameter values adaptively. Specifically, for luminance-domain algorithms, we use image exposure and histogram features to construct objective function; while for gradient-domain algorithms, we introduce texture-aware exponential mean local variance (EMLV) to build objective function. Finally, we propose a joint domain-aware logarithmic preprocessing method for deep-neural-network-based TM algorithms. The experimental results show that the novel preprocessing method AdaLogT endows each domain algorithm with wider scene adaptability and improves the performance in terms of visual effects and objective evaluations, the subjective and objective index scores of the tone mapping quality index improved by 6.04% and 5.90% on average for the algorithms.

Keywords: high dynamic range; tone mapping; adaptive logarithmic transformation; preprocessing

1. Introduction

The dynamic range of images is defined as the logarithm of the ratio of the maximum to the minimum luminance [1]. Through high dynamic range (HDR) images, we can restore the human eye's perception of scenes as much as possible [2]. Since the dynamic range of natural scenes often exceeds the display range of low dynamic range (LDR) images, traditional display devices cannot directly display HDR images well. Therefore, how to map HDR images to traditional displays and show them well has become one of the current research hotspots in image processing.

Tone mapping (TM) compresses the dynamic range of an image, mapping high contrast, wide gamut HDR images onto conventional display devices. In general, tone mapping algorithms consist of two parts: preprocessing and tone mapping. The pipeline is shown in Figure 1.

The TM algorithms' intentions can be classified as scene reproduction, best subjective quality and visual system simulator [3]. For scene reproduction, a variety of data processing methods are used in tone mapping operators (TMO), among which the logarithmic transformation is a simple and widely used one. Studies have shown that TM algorithms are closely related to human visual system (HVS) perception. The Weber–Fechner law [4] shows the sensitivity of HVS to luminance variations: the response of HVS in most of the luminance range has logarithmic characteristics, namely, there is a logarithmic relationship between the perceived luminance and physical luminance [3]. Thus, many TM algorithms

choose to compute in the logarithmic domain to ensure consistency between perceived luminance and scene luminance.

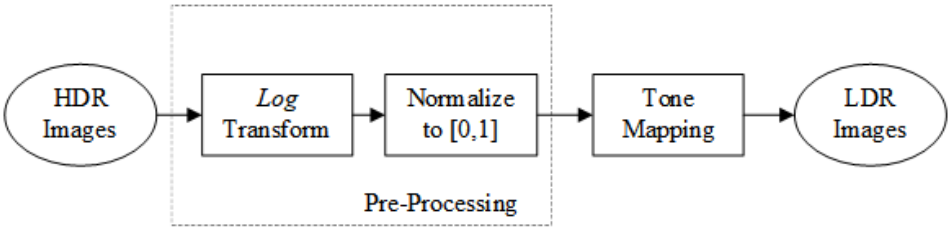


Figure 1. The pipeline of TM algorithms.

The TM algorithms process images from different domains. Traditional TM algorithms can be divided into luminance-domain methods and gradient-domain methods [2]. The layer decomposition method is representative of the luminance-domain algorithm. Durand et al. [5] proposed a single-scale decomposition of HDR images in the luminance domain using bilateral filter instead of Gaussian filter. Farbman et al. [6] constructed an edge-preserving filter based on the weighted least square (WLS) method and used multi-scale decomposition to further enhance the discrimination of low/high-frequency information in the image. The results show that the WLS algorithm suppresses the halo problem of Durand's method well. Paris et al. [7] proposed a tone mapping method based on the image Laplace pyramid with better halo reduction and detail retention properties. Liang et al. [8] introduced l_0, l_1 priors for different layers in the image decomposition process. Yang et al. [9] adaptively selected two appropriate gamma functions to adjust the brightness of dark and light areas, respectively. However, the dynamic range compression may lead to a little loss of visual naturalness. Beyond that, Dargo [10] first proposed an adaptive logarithmic tone mapping curve. Zhao et al. [11,12] proposed the effective TMOs by using localized contrast correction and Retinex [13]. Mantiuk et al. [14] used the contrast perturbation of the HVS model as the weight to construct the tone mapping operator, and Khan et al. [15] adjusted the image luminance histogram based on the just noticeable difference (JND) and used a look-up table to construct the mapping.

On the other hand, the gradient-domain algorithm performs dynamic range compression and detail enhancement by manipulating image gradients. Fattal et al. [16] constructed a compression function using multi-scale Gaussian pyramid to compress the large gradient of the image while keeping the small gradient unchanged or enhanced, which has the advantages of detail preservation and almost no halo effect. Bhat et al. [17] proposed a unified framework for gradient-domain image processing. Shibata et al. [18] combined the gradient-domain algorithm with the luminance-domain algorithm to avoid oversaturation and gradient reversal using luminance constraints. In addition, deep neural network (DNN)-based algorithms [19–22] have also been emerging in recent years, achieving significant advantages.

Many of the above algorithms use conventional logarithmic transformation LogT or its variants for preprocessing, without considering the diversity of natural scenes and various luminance ranges of different scenes. These preprocessing methods are also not tuned for the different domains TM algorithms, resulting in problems such as oversaturation and loss of details in the mapped images. Based on the analysis of various existing preprocessing methods, this paper proposes a domain-aware adaptive logarithmic transformation AdaLogT as a unified TM preprocessing method. We introduce the parameter p and construct different objective functions for luminance and gradient domains to determine the optimal parameter value of p. Specifically, for luminance-domain algorithms, we use image exposure and histogram features to construct the objective function to maximize the layered performance of the luminance-domain algorithms. For gradient-domain algorithms, texture-aware exponential mean local variance (EMLV) [23] is introduced to build the objective function to ensure the maximization of the input gradient information. Based on these, we propose a joint domain-aware logarithmic preprocessing method for DNN-based TM algorithms. The experimental results show that the proposed preprocessing

method endows each domain algorithm with wider scene adaptability and improves the performance in terms of visual effects and objective evaluations.

The rest of this paper is organized as follows. Section 2 analyzes the related preprocessing algorithms and proposes an adaptive logarithmic transformation model AdaLogT. Section 3 describes the objective functions corresponding to the luminance-domain and gradient-domain TM methods. Section 4 proposes a joint domain-aware logarithmic transformation for the DNN-based TM methods. Then, Section 5 presents the experimental results of subjective and objective comparisons with existing methods. Finally, Section 6 concludes this paper and outlooks for further work.

2. Related Work and Adaptive Logarithm Transformation Model

The luminance range of HDR images is approximately 0.0005 cd/m^2 to 10,000 cd/m^2 [24]. It is necessary to normalize the preprocessing so that the image pixel values fall within a specific range, reducing the computational complexity and ensuring the effectiveness of TM algorithms. In this section, we first review classic preprocessing methods. The summary of these research is shown in Table 1.

Let the input image be I, and image $\bar{I}$ in logarithmic transformation can be expressed as:

$$\bar{I} = log(I + \epsilon) \tag{1}$$

where $\epsilon = 1 \times 10^{-4}$. Considering the difference in dynamic range of different images, the normalization as follows is used on (1):

$$\widetilde{I} = \frac{\bar{I} - \bar{I}_{min}}{\bar{I}_{max} - \bar{I}_{min}} \tag{2}$$

where $\bar{I}_{min}$ and $\bar{I}_{max}$ represent the minimum and maximum pixel values of $\bar{I}$, respectively. $\widetilde{I}$ indicates the image after logarithmic transformation and normalization. Then, the range of pixel values are normalized to $[0, 1]$. This method is simple enough and has a good display effect. Liang [8] and other works [16,25,26] use it as a preprocessing step in the TM algorithms. We denote this method as the traditional logarithmic transformation LogT.

Stockham [27] recommends that the image should satisfy the following logarithmic relationship for image processing and display needs:

$$\widetilde{I} = \frac{log(I + 1)}{log(I_{max} + 1)} \tag{3}$$

Dargo [10] proposed an adaptive logarithmic tone curve for tone mapping. A bias power function is introduced to adaptively vary logarithmic bases. The algorithm changes the mapping of scene brightness and contrast by different logarithmic bases. Equation (4) present the tone mapping function:

$$\widetilde{I} = \frac{I_{dmax} \cdot 0.01}{log_{10}(I_{max} + 1)} \frac{log(I + 1)}{log(2 + ((\frac{I}{I_{max}})^{\frac{log(b)}{log(0.5)}} \cdot 8)} \tag{4}$$

where I_{dmax} is used as a scalefactor to adapt the output to its intended display. Generally, the reference value of I_{dmax} for displays is set at 100 cd/m^2. Adjusting the bias function parameter b is equivalent to adjusting the base of the logarithmic function, thus changing the overall effect of the result.

Gu [28] found that appealing results could be obtained by appropriately amplifying the input luminance. According to the dynamic range of conventional scenes, the following logarithmic transformation and normalization are given:

$$\widetilde{I} = \frac{log(I \cdot 10^6 + 1)}{log(I_{max} \cdot 10^6 + 1)}, \tag{5}$$

Recently, Vinker [21] proposed adaptive curve-based compression (ACC) for preprocessing of DNN algorithms, which maps and normalizes the input image through the following transformations:

$$\widetilde{I} = \frac{log(\lambda \cdot \frac{I}{I_{max}} + \varepsilon)}{log(\lambda + \varepsilon)}, \tag{6}$$

where λ is the scaling factor and the selection rule of λ is to minimize the following cross-entropy:

$$\arg\min_{\lambda} - \sum_{l} H_l(\widetilde{I}) log(H_l(LDR)), \tag{7}$$

where $H(\cdot)$ represents the histogram. $H(\widetilde{I})$ as a function of λ denotes the histogram of $\widetilde{I}$, and $H(LDR)$ represents the histogram of native LDR images. $H(LDR)$ is obtained by averaging the histogram of 900 high-quality images in the DIV2k [29] dataset. All histograms use 20 bins indexed by l.

Inspired by the above works, we construct a unified TM algorithms' preprocessing format named adaptive logarithmic transformation AdaLogT:

$$\widetilde{I} = \frac{log(I \cdot 10^p + 1)}{log(I_{max} \cdot 10^p + 1)} \triangleq AdaLogT(I; p), \tag{8}$$

where I_{max} represents the maximum value of image I. $\widetilde{I}$ is strictly limited to the range $[0, 1]$ after normalization.

Equations (2), (3), and (5) can be expressed in the form of Equation (8). In fact, Equation (5) corresponds to the special case of $p = 6$ in Equation (8). For Equation (2), we have

$$\widetilde{I} = log(I + \epsilon) = log(\frac{1}{\epsilon} \cdot I + 1) + log(\epsilon), \tag{9}$$

For $\widetilde{I}_{min}$, $\widetilde{I}_{max}$, there are

$$\widetilde{I}_{min} = log(I_{min} + \epsilon) = log(\frac{1}{\epsilon} \cdot \widetilde{I}_{min} + 1) + log(\epsilon), \tag{10}$$

$$\widetilde{I}_{max} = log(I_{max} + \epsilon) = log(\frac{1}{\epsilon} \cdot \widetilde{I}_{max} + 1) + log(\epsilon), \tag{11}$$

Since $\widetilde{I}_{min} = 0$, bringing Equations (9)–(11) into Equation (8) has

$$\begin{aligned} \widetilde{I} &= \frac{log(\frac{1}{\epsilon} \cdot I + 1) - log(\frac{1}{\epsilon} \cdot I_{min} + 1)}{log(\frac{1}{\epsilon} \cdot I_{max} + 1) - log(\frac{1}{\epsilon} \cdot I_{min} + 1)} \\ &= \frac{log(\frac{1}{\epsilon} \cdot I + 1)}{log(\frac{1}{\epsilon} \cdot I_{max} + 1)}, (0 \leqslant \widetilde{I} \leqslant 1) \end{aligned} \tag{12}$$

Therefore, Equation (8) is a generalized form of Equations (2) and (5). The parameter p enables us to adaptively obtain a suitable log-normalized transformation for different input images, which represents the order of magnitude of image amplification as shown in Figure 2. Using the parameter p to amplify the input luminance is equivalent to performing the corresponding global compression mapping, which has the properties of enhancing the contrast of low-luminance while compressing the dynamic range of high-luminance for all pixels of the image.

With the increase in the parameter p, the range of enhancement area reduced, and the amplitude of enhancement increased. Meanwhile, the suppression effect of the highlighted area is enhanced, and the overall luminance of the image is improved. When p is too large, the dynamic range of the original low-luminance region is also compressed, and the image details will be suppressed, resulting in the lack of contrast. For HDR images where most of the data is located in the low-luminance region and a small part of the data have the characteristics of very high luminance, the selection of parameter p is essentially a

trade-off between the detail enhancement region and the dynamic range compression region. From Figure 2, we can select the range of p as $[-5, 10]$. For different input images, the parameter p is selected adaptively according to image information and the domain of the TM algorithm. Specific selection strategies will be given in the following sections.

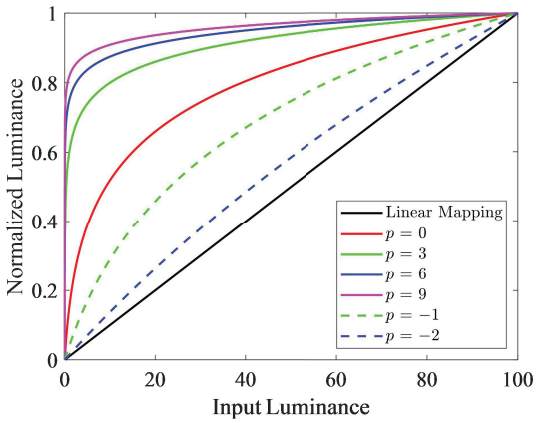


Figure 2. Mapping curves with different p values.

Table 1. Summary of tone mapping preprocessing research.

Researcher	Expression	Advantage	Disadvantage
Stockham [27]	$\frac{log(I+1)}{log(I_{max}+1)}$	Strictly mapped to the interval $[0, 1]$	The luminance compression is excessive; The lost of high contrast content.
Dargo [10]	$\frac{I_{dmax} \cdot 0.01}{log_{10}(I_{max}+1)} \frac{log(I+1)}{log(2+((\frac{I}{I_{max}})^{\frac{log(b)}{log(0.5)}} \cdot 8)}$	Well-suited to the specific image content	Parameter b needs to be adjusted for different images; Local contrast reduction
Gu [28]	$\frac{log(I \cdot 10^6+1)}{log(I_{max} \cdot 10^6+1)}$	Enhancing low-light areas of the image; Improving the overall brightness of the image	Overexposure may occur
Vinker [21]	$\frac{log(\lambda \cdot \frac{I}{I_{max}}+\varepsilon)}{log(\lambda+\varepsilon)}$	Adaptive searching for appropriate mapping curves	High computational complexity; Not strictly normalized to the interval $[0, 1]$

3. Domain-Aware Objective Function

Traditional TM algorithms can be classified into luminance-domain and gradient-domain methods broadly. The luminance-domain methods use layer decomposition, histogram [15,30,31], HVS [32–34], etc. to deal with image luminance. These types of methods consider how to compress the HDR image luminance to the display range of traditional display devices. The gradient-domain methods focus on the preservation of image contrast and gradient, directly acting on the image gradient to achieve overall dynamic range compression. Based on their different focus directions, we propose different objective functions in the luminance domain and gradient domain to guide the selection of the parameter p, which are called luminance-domain-aware AdaLogT methods and gradient-domain-aware AdaLogT methods, respectively. The overall flowchart of the proposed method is shown in Figure 3.

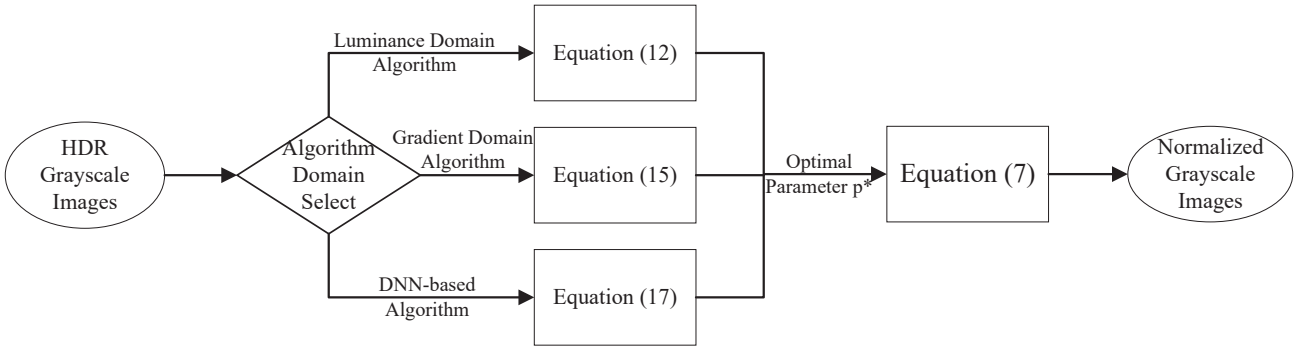


Figure 3. Overall flowchart of the proposed method.

3.1. Luminance-Domain-Aware AdaLogT Method

The grayscale mean of image pixels reflects the exposure degree of the image [35–37]. Direct observation shows that the distribution of HDR image pixels at each luminance level is uneven [38], as shown in Figure 4a. On the other hand, histogram equalization states that if the pixels of an image can be distributed evenly over all possible gray levels, the image will have high contrast and richer details. In AdaLogT, different parameters p can adjust the luminance distribution of the image without changing the overall shape of the image histogram, thereby adjusting the exposure level of the image. Figure 4 shows the image grayscale histogram under different parameters. With the logarithmic transformation of different p values, the image histogram gradually extends to other luminance levels.

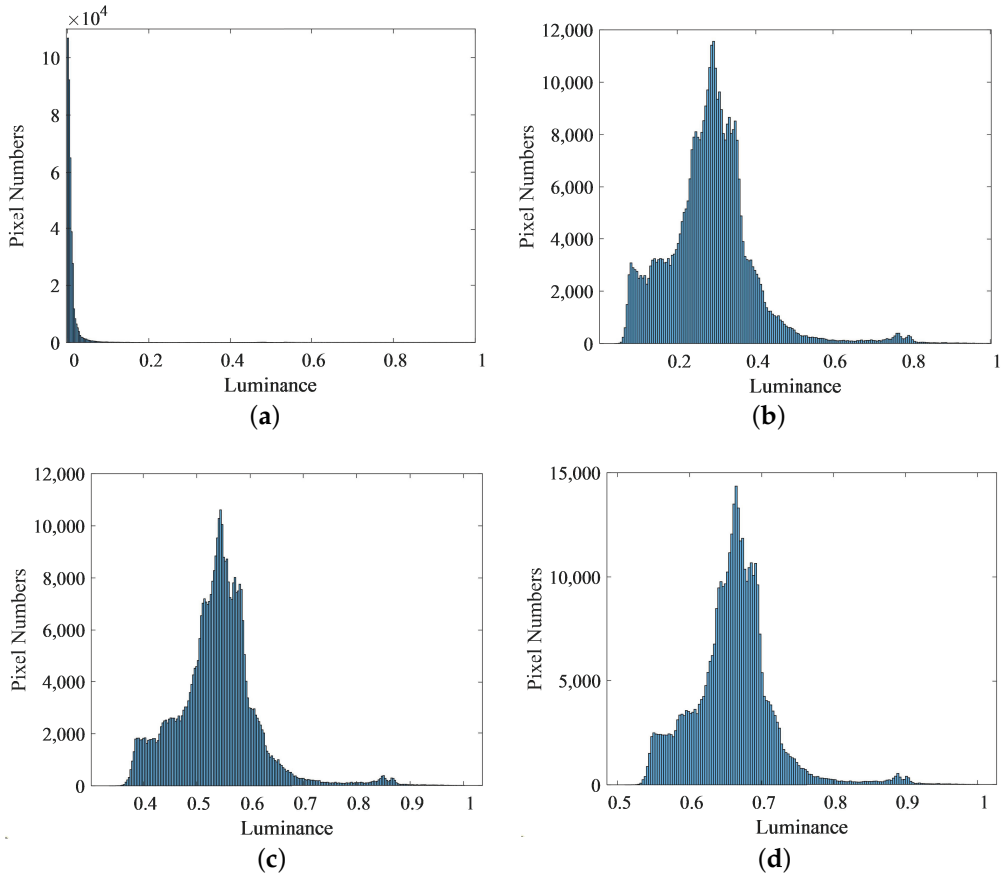


Figure 4. Image histogram distribution under different parameters p: (**a**) radiance map; (**b**) $p = 3$; (**c**) $p = 6$; (**d**) $p = 9$.

Image skewness $s(\cdot)$ measures the symmetry of the image distribution concerning the mean value, and the degree of its approximation to 0 reflects the degree of symmetry of the distribution. Therefore, we introduce image skewness to ensure the symmetry of the image distribution after adjustment. Thus, the objective function and corresponding optimization problem for luminance-domain TM algorithms are given as follows:

$$\begin{aligned} p^* &= \arg\min_p \alpha \cdot \left\| \widetilde{I} - T \right\|_F^2 + (1-\alpha) \cdot \left| s(\widetilde{I}) \right|, \\ s.t.\ \widetilde{I} &= log(I \cdot 10^p + 1)/log(I_{max} \cdot 10^p + 1), \\ s(\widetilde{I}) &= \mathbb{E}(\widetilde{I} - \mu)^3/\sigma^3, \end{aligned} \tag{13}$$

where $\|\cdot\|_F$ indicates Frobenius norm. $s(\widetilde{I})$ is the skewness of image $\widetilde{I}$, $|s(\widetilde{I})|$ means the absolute value of $s(\widetilde{I})$, and $\mathbb{E}(\cdot)$ denotes the expectation. μ and σ are respectively the mean and variance of $\widetilde{I}$. T represents the exposure level of the target image, and α is the weight to balance exposure and skewness. This paper defaults to $T = 0.5$ and $\alpha = 0.8$.

We can calculate the above optimization problem with the trichotomy method to obtain the optimal parameter p, as shown in Algorithm 1. Figure 5 shows the grayscale histograms of some images before and after adaptive logarithmic transformation. By comprehensively considering exposure and skewness, the AdaLogT images have better display luminance and contrast, which is reflected in the histogram, that is, the pixels are distributed to as many grayscale levels as possible.

Algorithm 1: Trichotomy method for optimum value

input : Grayscale high dynamic range image I, Exposure level T, Lower bound l, Upper bound L, weight α.

output: p^*

$f(I) := \alpha \cdot \left\| I - T \right\|_F^2 + (1-\alpha) \cdot \left| s(I) \right|$;
$I_p := log(I \cdot 10^p + 1)/log(I_{max} \cdot 10^p + 1)$;
while $l <= L$ **do**
    $l_{mid} = (l + L) >> 1$, $L_{mid} = (l_{mid} + L) >> 1$;
    Compute $I_{l_{mid}}, I_{L_{mid}}, f(I_{l_{mid}}), f(I_{L_{mid}})$;
    if $f(I_{l_{mid}}) <= f(I_{L_{mid}})$ **then**
        $L = L_{mid} - 1$
    else
        $l = l_{mid} + 1$
$p^* = l$

(**a**)

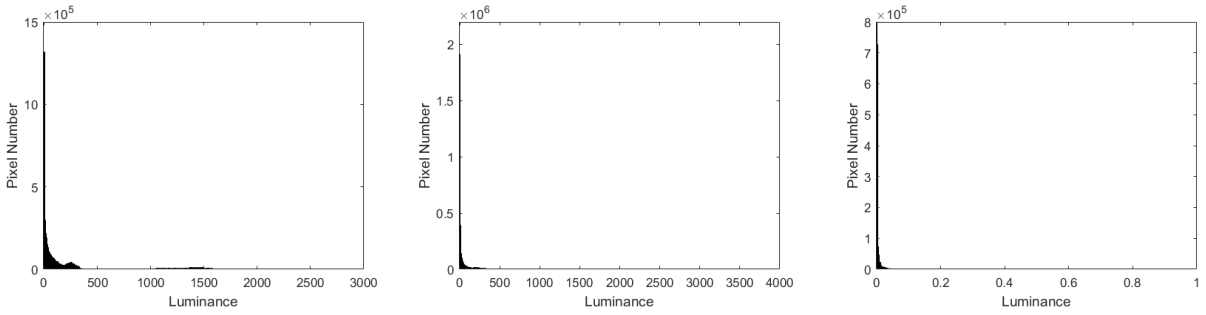


Figure 5. *Cont.*

(**b**)

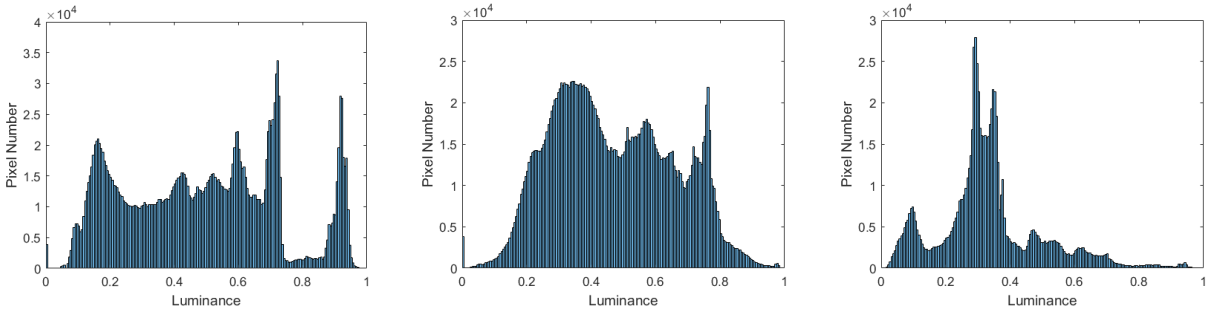


Figure 5. Histogram before and after adaptive logarithmic transformation of different images: (**a**) radiance map; (**b**) AdaLogT.

3.2. Gradient-Domain-Aware AdaLogT Method

For gradient-domain algorithms, the enhancement of image details by preprocessing is the most critical. The gradient-domain algorithm represented by [16] reconstructs the image by solving the Poisson equation:

$$\Delta f = div(G) \tag{14}$$

where Δ is the Laplace operator. $div(\cdot)$ denotes the divergence. f is the output image to be reconstructed, and G is the guided gradient field calculated according to the log-transformed image gradients. The guided gradient field maintains the order relationship of the original image gradients and compresses the large gradients while enhancing the small gradients. However, the enhancement of small gradients depends on the input image gradients, which leads to the problem that the reconstructed image is too dark to distinguish details due to the small gradients of the input image detail part.

Based on this observation, the key to selecting the logarithmic transformation parameters of gradient-domain algorithms is to ensure the image has good detail performance. Therefore, the mean of exponential mean local variance (EMLV) [23] is introduced as the measure of image detail, denoted as M_g:

$$M_g = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{1}{|\Omega|}\sum_{\Omega}\nabla\widetilde{I}_i\right|^{\gamma}, \tag{15}$$

where $\widetilde{I}_i$ represents the ith pixel of image $\widetilde{I}$, and Ω is a 3×3 neighborhood of $\widetilde{I}_i$. $|\Omega|$ means the number of pixels in Ω, and N denotes the total number of pixels in $\widetilde{I}$. $|\nabla\widetilde{I}_i| = \sqrt{(\partial_x\widetilde{I}_i)^2 + \partial_y\widetilde{I}_i)^2}$, ∂_i denotes the partial derivative with respect to the direction i. γ determines the sensitivity to the gradient of $\widetilde{I}$, and γ is taken as 0.5 in this paper.

Figure 6 shows the change of M_g after changing the parameter p in different images. As the parameter p increases, the M_g value of the log-transformed image presents a single peak that first increases and then decreases. In further experiments, when p is small, the image luminance is low, and texture details are lost. While the image luminance and contrast decrease when p is too high. Therefore, we suggest that the parameter p be selected to maximize M_g after transformation to ensure the maximization of input gradient information. We give the optimization problem corresponding to the objective function of the gradient-domain TM algorithms:

$$\begin{aligned} &p^* = \arg\max_{p}\frac{1}{N}\sum_{i=1}^{N}\left|\frac{1}{|\Omega|}\sum_{\Omega}\nabla\widetilde{I}_i\right|^{\gamma}, \\ &s.t.\ \widetilde{I} = log(I\cdot 10^p + 1)/log(I_{max}\cdot 10^p + 1), \end{aligned} \tag{16}$$

Due to the unimodality of the objective function, we can use a zero-order optimization method such as the Fibonacci method to calculate the optimal value of p.

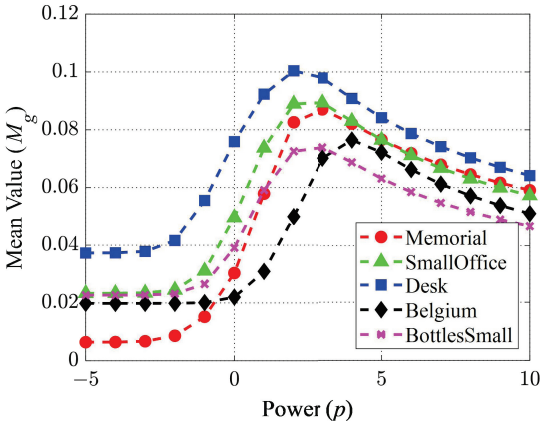


Figure 6. The relationship between the mean value of M_g and p.

4. *AdaLogT* Method for DNN-Based TM Algorithms

Many algorithms, such as the DNN-based TM algorithm, do not solely consider image luminance or gradient information. DNN algorithms are data-driven and learn the main features of TM process through a large number of samples, which include but are not limited to image luminance and gradient, etc. Some of its convolution operations may contain the function of the average operator, while others may contain the function of the difference operator. Therefore, the objective functions for a single domain may not enhance all the information required by the algorithm.

In Vinker [21] , a log-normalized preprocessing method for DNN-based TM algorithms was proposed. For Equations (6) and (7), the following two issues need further study and discussion.

(1) Normalization. $\widetilde{I}_{max} = 1$, but $\widetilde{I}_{min} = \frac{log(\epsilon)}{log(\lambda+\epsilon)} \neq 0$ when $I_{min} = 0$. In other words, Equation (6) does not strictly map the input luminance to $[0, 1]$. If we modify Equation (6) to:

$$\widetilde{I} = \frac{log(\lambda \cdot \frac{I}{I_{max}} + \varepsilon) - log(\epsilon)}{log(\lambda + \varepsilon) - log(\epsilon)}, \tag{17}$$

Then $\widetilde{I} \in [0, 1]$.In this case $\frac{\lambda}{\epsilon \cdot I_{max}} = 10^p$, the selection of λ is transformed into the problem of selection of p.

(2) Computational complexity. Equation (7) uses the mean of the luminance histograms of 900 LDR images in the DIV2k [29] dataset as reference. Ideally, the calculation of the histogram means should use the distance between distributions, such as earth mover's distance (EMD) [39], which is computationally expensive. Specifically, Vinker uses the stochastic search method [40] to find suitable values within 1 to 1×10^9 and uses a floating point type with a high degree of computational accuracy, which needs to be continually performed. Depending on the variation of the mapping curve with different parameters in Figure 2, there is less gain in increased accuracy as it takes a large parameter change to make a significant difference to the curve. Figure 7 gives a comparison of ACC and AdaLogT execution times and shows that ACC has a far greater computational complexity than AdaLogT.

Using the analysis in Section 3, we know that since the DNN contains both the luminance domain and the gradient domain, the corresponding objective function should have the form of joint domain perception:

$$\begin{aligned} p^* = \arg\min_p \alpha \cdot (\left\| \widetilde{I} - T \right\|_F^2) + (1-\alpha) \cdot \left| s(\widetilde{I}) \right| \\ - \frac{1}{N} \sum_{i=1}^{N} \left| \frac{1}{|\Omega|} \sum_{\Omega} \nabla \widetilde{I}_i \right|^{\gamma}, \\ s.t.\ \widetilde{I} = log(I \cdot 10^p + 1)/log(I_{max} \cdot 10^p + 1), \end{aligned} \tag{18}$$

Equation (18) integrates image luminance and gradient features with wide perceptual range, which can be solved by methods such as step-by-step method. Compared with Equation (7), the computational cost of Equation (18) is greatly reduced.

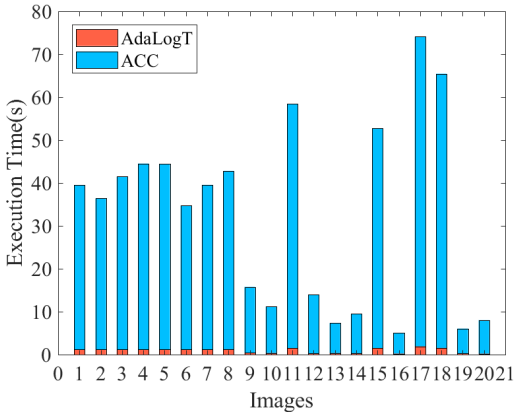


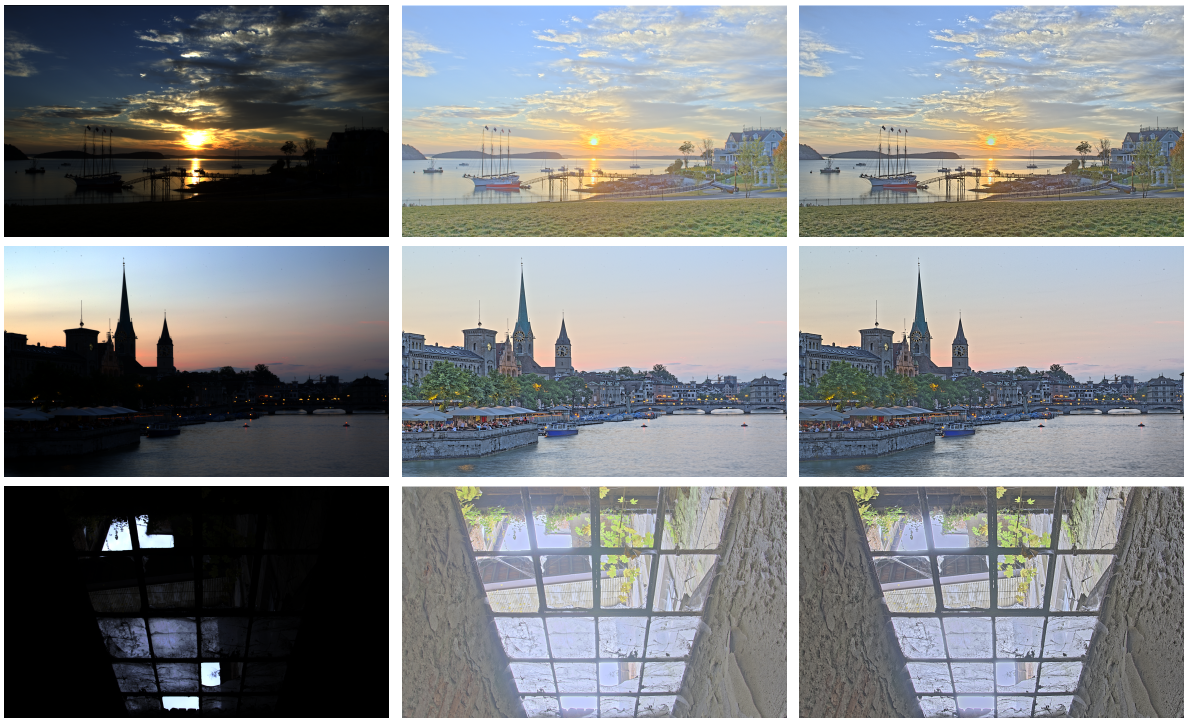
Figure 7. Comparison of ACC andAdaLogT image execution time.

5. Experimental Results and Analysis

The state-of-the-art TM algorithms used for comparison in the experiment are the luminance-domain algorithm Gu [28], the gradient-domain algorithm Fattal [16], and the deep neural network algorithm Vinker [21]. The source codes of Gu [28] and Vinker [21] are obtained from the authors' homepage, and we use the default parameters of the programs. Additionally, the network pre-training parameters given by Vinker [21] are used as the default. The Fattal [16] algorithm is implemented by 'LuminanceHDR ' (https://qtpfsgui.sourceforge.net/, accessed on 13 September 2022) and gamma is set to 2.2. All these experiments were run on a HP Workstation Z680 with Intel Xeon E5630 CPU, NIVDIA GeForce 2080Ti GPU and 32 GB memory. To fully consider the differences in different scenes, we perform experiments on a large number of HDR images and randomly select 20 images for experimental analysis.

5.1. Luminance-Domain Algorithm

The preprocessing method proposed by Gu [28] provides a better display for brighter scenes. However, for indoor and outdoor dark scenes, there will be problems where the background luminance of output images does not match the real scene, and the overall contrast is reduced. The luminance-aware AdaLogT better considers the background luminance of different scenes and enhances the subjective visual effect. Figure 8 shows the comparison of the two methods. A subjective experiment is conducted based on the results of 20 HDR images. We invite ten people to evaluate the experimental results, 6 males and 4 females, six of whom have a research area in image processing. The rating scale is from 1 (worst) to 10 (best) in steps of 1. The results are displayed on Samsung S32R750UEC 32 inch (4096×2160). Compared with the mean and standard deviation of Gu's method (6.84, 1.55), our method (7.40, 1.15) achieved an 8% improvement.



(**a**) (**b**) (**c**)

Figure 8. Comparison of the results of the luminance-domain TM algorithm in different preprocessing methods: (**a**) radiance map; (**b**) Gu [28]; (**c**) AdaLogT.

We also select the tone-mapped image quality index (TMQI) [41] for objective evaluation. TQMI evaluates images from multiple perspectives. This method measures the structural fidelity and naturalness scores of tone-mapped results. Then, it comprehensively gives a final score ranging from 0 to 1. A larger value of TMQI represents better result achieved by the TM algorithm. A scatter plot is used to visualize the TMQI scores of different preprocessing methods on experimental images. As shown in Figure 9, AdaLogT achieves better results in most images.

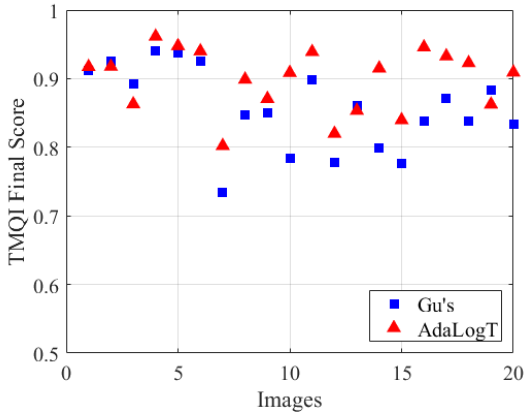


Figure 9. TMQI final score of the luminance-domain algorithm in different preprocessing methods.

Table 2 shows the mean TMQI scores of 20 experimental images before and after the adoption of AdaLogT, where the highest score is given in bold. We observe that AdaLogT improves the adaptability of the TM algorithm to different scenes. The appropriate exposure choice brings great advantages to the image display, effectively improving the naturalness score of TMQI and resulting in a higher TMQI final score.

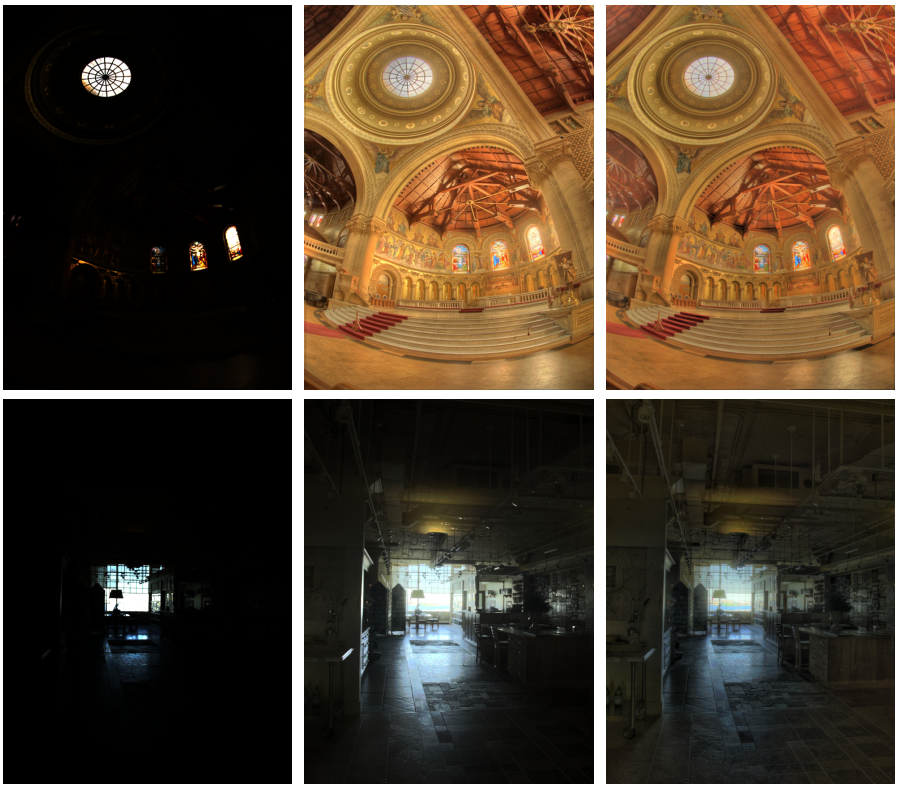
Table 2. Mean TMQI scores of luminance-domain algorithm.

Preprocessing	Structure	Naturalness	Final
Gu's	0.8273	0.4098	0.8562
AdaLogT	0.8305	0.6346	0.8983

5.2. Gradient-Domain Algorithm

The gradient-domain algorithm [16] based on the gradient-domain-aware AdaLogT method has achieved good results in quantitative evaluation and objective assessment.

Figure 10 shows the comparison of TM results for bright and dark scenes, where the first column is the HDR radiance map, the second column is the TM results under LogT, and the third column is the TM results using AdaLogT. When the input image is dark, the result under LogT is dim and the details are vague or even indistinguishable. Our method corrects image exposure and achieves a balance between detail preservation and image naturalness. The structure and local details of the image are better preserved in the result, and the visual effect is more consistent with the human eye's perception of the scene. In the quantitative user evaluation, the Fattal method and ours achieve scores of (6.73, 1.13) and (7.16,1.07), respectively.



(**a**) (**b**) (**c**)

Figure 10. Comparison of the results of the gradient-domain TM algorithm in different preprocessing methods: (**a**) radiance map; (**b**) LogT; (**c**) AdaLogT.

Figure 11 is the scatter plot of the gradient-domain algorithm results under different preprocessing methods. Our algorithm achieves equal or better scores in the vast majority of images. Table 3 presents the mean TMQI scores under preprocessing methods LogT and AdaLogT. The gradient-domain objective function pays attention to image details and textures so that the TM results have better visual brightness and detail preservation. Therefore, compared with the result of LogT, the result of AdaLogT has a higher TMQI structure score, and has also achieved significant improvement in the naturalness index.

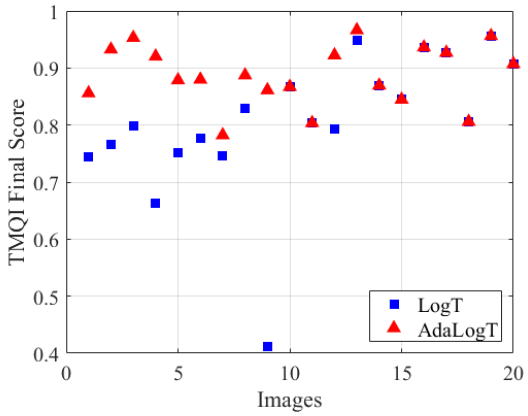


Figure 11. TMQI final score of the gradient-domain algorithm in different preprocessing methods.

Table 3. Mean TMQI scores of gradient-domain algorithm.

Preprocessing	Structure	Naturalness	Final
LogT	0.8104	0.2577	0.8072
AdaLogT	0.8647	0.5272	0.8879

5.3. DNN-Based TM Algorithm

Vinker [21] achieved good results by building a generative adversarial network to perform unpaired data training. Compared with its proposed adaptive curve-based compression (ACC), the objective function for the DNN-Based TM algorithm comprehensively considers image luminance and gradient characteristics, improving the subjective and objective quality of TM results. ACC uses an average of 900 image histograms as the primary target for parameter selection. The image exposure, skewness, and gradient priors are used in this paper to make the results have the same or even better display effect, as shown in Figure 12. The subjective scores of Vinker and our method are (7.11, 1.16) and (7.37, 1.06). According to the quantitative subjective evaluation, we have made certain progress in the subjective effect.

Figure 13 shows the TMQI quality scores of the two preprocessing methods for each experimental image, and Table 4 shows the mean TMQI scores of the TM algorithm [21] under different preprocessing methods. For the DNN-Based algorithm with complex features, the proposed method outperforms the ACC method in most images. Moreover, the enhancement of image gradients ensures that the TM algorithm results have better structure-preserving properties, achieving higher TMQI structure and naturalness scores.

Table 4. Mean TMQI scores of DNN-based TM algorithm.

Preprocessing	Structure	Naturalness	Final
ACC	0.8587	0.5398	0.8872
AdaLogT	0.8798	0.6383	0.9129

Figure 12. Comparison of the results of the DNN-based TM algorithm in different preprocessing methods: (**a**) radiance map; (**b**) ACC [21]; (**c**) AdaLogT.

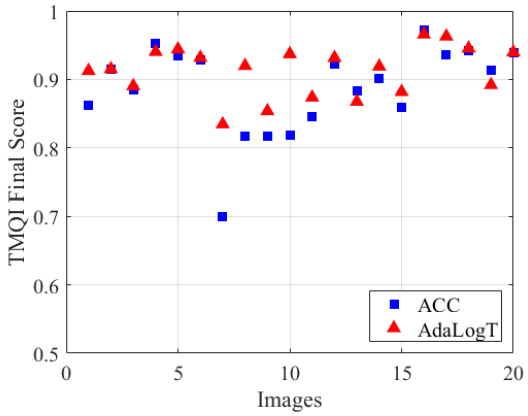


Figure 13. TMQI final score of the DNN-based TM algorithm in different preprocessing methods.

6. Conclusions

This paper proposed an adaptive logarithmic normalization transformation, AdaLogT, for TM algorithms in order to compensate for the defects caused by the preprocessing. Based on the analysis of classical preprocessing methods, the parameter p was introduced in order to obtain the appropriate log-normalized curve. The optimal parameter p was calculated by proposed objective functions. Considering the TM algorithms based on luminance or gradient domain, the objective functions based on luminance and gradient-domain features were constructed, respectively. Furthermore, a joint domain-aware objective function was presented for DNN-based TM algorithms. The proposed preprocessing algorithm ensures that the image luminance conforms to the HVS perception of the scene brightness. State-of-the-art luminance, gradient-domain, and DNN-based algorithms were selected for the experiments, which used different preprocessing methods. The experiments were conducted by combining subjective qualitative and objective quantitative. The results show

that the proposed algorithm achieves the best subjective quantitative scores with TMQI quality scores, which indicates that the method improves the subjective effect and objective quality of the images. Further work includes the improvement of the optimization method for optimal parameter p and the utilization of folded concave functions for more general tone mapping preprocessing.

Author Contributions: Conceptualization, X.F. (Xuelai Fang) and X.F. (Xiangchu Feng); Supervision, X.F. (Xiangchu Feng); Writing—original draft, X.F. (Xuelai Fang); Writing—review and editing, X.F. (Xuelai Fang) and X.F. (Xiangchu Feng). All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Nature Science Foundation of China under Grant 61772389 and Grant 61972264.

Data Availability Statement: All experimental data are from publicly available data sets. The URL for the datasets are: https://ivc.uwaterloo.ca/database/TMQI/TMQI-Database.html, https://pfstools.sourceforge.net/hdr_gallery.html, and https://qualinet.github.io/databases/image/tone-mapped-image-quality-database/(accessed on 21 October 2022).

Acknowledgments: We would like to express our sincere gratitude to the three reviewers for their insightful and valuable feedback, which has helped us improve our work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. DiCarlo, J.M.; Wandell, B.A. Rendering high dynamic range images. In *Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications*; SPIE: Bellingham, WA, USA, 2000; Volume 3965, pp. 392–401.
2. Reinhard, E.; Heidrich, W.; Debevec, P.; Pattanaik, S.; Ward, G.; Myszkowski, K. *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*; Morgan Kaufmann: Burlington, MA, USA, 2010.
3. Eilertsen, G.; Mantiuk, R.K.; Unger, J. A comparative review of tone-mapping algorithms for high dynamic range video. In *Computer Graphics Forum*; Wiley Online Library: New York, NY, USA, 2017; Volume 36, pp. 565–592.
4. Gibbon, J. Scalar expectancy theory and Weber's law in animal timing. *Psychol. Rev.* **1977**, *84*, 279. [CrossRef]
5. Durand, F.; Dorsey, J. Fast bilateral filtering for the display of high-dynamic-range images. In Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, San Antonio, TX, USA, 23–26 July 2002; pp. 257–266.
6. Farbman, Z.; Fattal, R.; Lischinski, D.; Szeliski, R. Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM Trans. Graph. (TOG)* **2008**, *27*, 1–10. [CrossRef]
7. Paris, S.; Hasinoff, S.W.; Kautz, J. Local laplacian filters: Edge-aware image processing with a laplacian pyramid. *ACM Trans. Graph.* **2011**, *30*, 68. [CrossRef]
8. Liang, Z.; Xu, J.; Zhang, D.; Cao, Z.; Zhang, L. A hybrid l1-l0 layer decomposition model for tone mapping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4758–4766.
9. Yang, K.F.; Li, H.; Kuang, H.; Li, C.Y.; Li, Y.J. An adaptive method for image dynamic range adjustment. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 640–652. [CrossRef]
10. Drago, F.; Myszkowski, K.; Annen, T.; Chiba, N. Adaptive logarithmic mapping for displaying high contrast scenes. In *Computer Graphics Forum*; Wiley Online Library: New York, NY, USA, 2003; Volume 22, pp. 419–426.
11. Zhao, L.; Li, G.; Wang, J. Tone Mapping Method Based on the Least Squares Method. *Electronics* **2022**, *12*, 31. [CrossRef]
12. Zhao, L.; Sun, R.; Wang, J. Three-Stage Tone Mapping Algorithm. *Electronics* **2022**, *11*, 4072. [CrossRef]
13. Land, E.H. The retinex theory of color vision. *Sci. Am.* **1977**, *237*, 108–129. [CrossRef]
14. Mantiuk, R.; Daly, S.; Kerofsky, L. Display adaptive tone mapping. In *ACM SIGGRAPH 2008 Papers on—SIGGRAPH '08*; ACM Press: New York, NY, USA, 2008 ; pp. 1–10.
15. Khan, I.R.; Rahardja, S.; Khan, M.M.; Movania, M.M.; Abed, F. A tone-mapping technique based on histogram using a sensitivity model of the human visual system. *IEEE Trans. Ind. Electron.* **2017**, *65*, 3469–3479. [CrossRef]
16. Fattal, R.; Lischinski, D.; Werman, M. High dynamic range compression. In Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, San Antonio, TX, USA, 23–26 July 2002; pp. 249–256.
17. Bhat, P.; Zitnick, C.L.; Cohen, M.; Curless, B. Gradientshop: A gradient-domain optimization framework for image and video filtering. *ACM Trans. Graph. (TOG)* **2010**, *29*, 1–14. [CrossRef]
18. Shibata, T.; Tanaka, M.; Okutomi, M. Gradient-domain image reconstruction framework with intensity-range and base-structure constraints. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2745–2753.
19. Rana, A.; Singh, P.; Valenzise, G.; Dufaux, F.; Komodakis, N.; Smolic, A. Deep tone mapping operator for high dynamic range images. *IEEE Trans. Image Process.* **2019**, *29*, 1285–1298. [CrossRef]

20. Panetta, K.; Kezebou, L.; Oludare, V.; Agaian, S.; Xia, Z. Tmo-net: A parameter-free tone mapping operator using generative adversarial network, and performance benchmarking on large scale hdr dataset. *IEEE Access* **2021**, *9*, 39500–39517. [CrossRef]
21. Vinker, Y.; Huberman-Spiegelglas, I.; Fattal, R. Unpaired learning for high dynamic range image tone mapping. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 20–25 June 2021; pp. 14657–14666.
22. Cao, X.; Lai, K.K.; Smith, M.R.; Yanushkevich, S. Adversarial and adaptive tone mapping operator: Multi-scheme generation and multi-metric evaluation. *J. Electron. Imaging* **2021**, *30*, 043020. [CrossRef]
23. Xu, J.; Hou, Y.; Ren, D.; Liu, L.; Zhu, F.; Yu, M.; Wang, H.; Shao, L. Star: A structure and texture aware retinex model. *IEEE Trans. Image Process.* **2020**, *29*, 5022–5037. [CrossRef]
24. Azimi, M.; Boitard, R.; Nasiopoulos, P.; Pourazad, M.T. Visual color difference evaluation of standard color pixel representations for high dynamic range video compression. In Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), Kos, Greece, 28 August–2 September 2017; pp. 1480–1484.
25. Duan, J.; Qiu, G.; Chen, M. Comprehensive fast tone mapping for high dynamic range image visualization. In *Pacific Graphics*; Citeseer: Macau, China, 2005.
26. Zhang, Z.; Han, C.; He, S.; Liu, X.; Zhu, H.; Hu, X.; Wong, T.T. Deep binocular tone mapping. *Vis. Comput.* **2019**, *35*, 997–1011. [CrossRef]
27. Stockham, T.G. Image processing in the context of a visual model. *Proc. IEEE* **1972**, *60*, 828–842. [CrossRef]
28. Gu, B.; Li, W.; Zhu, M.; Wang, M. Local edge-preserving multiscale decomposition for high dynamic range image tone mapping. *IEEE Trans. Image Process.* **2012**, *22*, 70–79. [PubMed]
29. Agustsson, E.; Timofte, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 126–135.
30. Boschetti, A.; Adami, N.; Leonardi, R.; Okuda, M. High dynamic range image tone mapping based on local histogram equalization. In Proceedings of the 2010 IEEE International Conference on Multimedia and Expo, Beijing, China, 26–28 October 2010; pp. 1130–1135.
31. Khan, I.R.; Aziz, W.; Shim, S.O. Tone-mapping using perceptual-quantizer and image histogram. *IEEE Access* **2020**, *8*, 31350–31358. [CrossRef]
32. Reinhard, E.; Stark, M.; Shirley, P.; Ferwerda, J. Photographic tone reproduction for digital images. In Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, San Antonio, TX, USA, 23–26 July 2002; pp. 267–276.
33. Li, Z.; Zheng, J. Visual-salience-based tone mapping for high dynamic range images. *IEEE Trans. Ind. Electron.* **2014**, *61*, 7076–7082. [CrossRef]
34. Barai, N.R.; Kyan, M.; Androutsos, D. Human visual system inspired saliency guided edge preserving tone-mapping for high dynamic range imaging. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 1017–1021.
35. Mertens, T.; Kautz, J.; Van Reeth, F. Exposure fusion. In Proceedings of the 15th Pacific Conference on Computer Graphics and Applications (PG'07), Maui, HI, USA, 29 October–2 November 2007; pp. 382–390.
36. Mertens, T.; Kautz, J.; Van Reeth, F. Exposure fusion: A simple and practical alternative to high dynamic range photography. In *Computer Graphics Forum*; Wiley Online Library: New York, NY, USA, 2009; Volume 28, pp. 161–171.
37. Guo, C.; Li, C.; Guo, J.; Loy, C.C.; Hou, J.; Kwong, S.; Cong, R. Zero-reference deep curve estimation for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1780–1789.
38. Miao, D.; Zhu, Z.; Bai, Y.; Jiang, G.; Duan, Z. Novel tone mapping method via macro-micro modeling of human visual system. *IEEE Access* **2019**, *7*, 118359–118369. [CrossRef]
39. Panaretos, V.M.; Zemel, Y. Statistical aspects of Wasserstein distances. *Annu. Rev. Stat. Its Appl.* **2019**, *6*, 405–431. [CrossRef]
40. Storn, R.; Price, K. Differential Evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341. [CrossRef]
41. Yeganeh, H.; Wang, Z. Objective quality assessment of tone-mapped images. *IEEE Trans. Image Process.* **2012**, *22*, 657–667. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Improved Reconstruction Algorithm of Wireless Sensor Network Based on BFGS Quasi-Newton Method

Xinmiao Lu [1,*], Cunfang Yang [1], Qiong Wu [1,2], Jiaxu Wang [1], Yuhan Wei [1], Liyu Zhang [1], Dongyuan Li [1] and Lanfei Zhao [1]

[1] School of Measurement-Control Technology and Communications Engineering, Harbin University of Science and Technology, Harbin 150080, China
[2] Heilongjiang Network Space Research Center, Harbin 150090, China
* Correspondence: lvxinmiao0611@hrbust.edu.cn; Tel.: +86-187-0462-8841

Abstract: Aiming at the problems of low reconstruction rate and poor reconstruction precision when reconstructing sparse signals in wireless sensor networks, a sparse signal reconstruction algorithm based on the Limit-Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) quasi-Newton method is proposed. The L-BFGS quasi-Newton method uses a two-loop recursion algorithm to find the descent direction d_k directly by calculating the step difference between m adjacent iteration points, and a matrix H_k approximating the inverse of the Hessian matrix is constructed. It solves the disadvantages of BFGS requiring the calculation and storage of H_k, reduces the algorithm complexity, and improves the reconstruction rate. Finally, the experimental results show that the L-BFGS quasi-Newton method has good experimental results for solving the problem of sparse signal reconstruction in wireless sensor networks.

Keywords: sparse signal reconstruction; L-BFGS quasi-Newton method; two-loop recursion algorithm; reconstruction rate

1. Introduction

A wireless sensor network (WSN) is a wireless network composed of a large number of sensor nodes. Each sensor node needs to collect a large amount of data and transmit it to the sink node. Traditional data acquisition and transmission methods will cause huge energy consumption and reduce the service life of nodes [1]. If the collected data is compressed before transmission, the energy consumption can be greatly reduced and the node life can be prolonged. However, the traditional Nyquist sampling frequency is too high, resulting in excess energy consumption. To solve the problem that Nyquist's sampling frequency is too high. Jose et al. proposed to apply the compressed sensing (CS) [2] theory to wireless sensor networks. The CS theory shows that as long as the signal is sparse or can be sparsely represented, the original signal can be reconstructed undistorted from the measurement of a linear combination far below Nyquist sampling frequency. The original signals of wireless sensor networks are sparse, as such, and CS theory is widely used in the reconstruction of sparse original signals in wireless sensor networks.

At present, the main reconstruction algorithms studied include greedy iterative algorithms [3], non-convex optimization algorithms [4], and convex optimization algorithms [5]. The most classical greedy iterative algorithm is the matching pursuit (MP) algorithm [6]. The algorithm uses residuals to reconstruct sparse signals. The algorithm is relatively simple, and the reconstruction rate is fast, but the sparsity of signals in a certain transformation domain is required, and the reconstruction accuracy is poor. It is suitable for cases with a large amount of data and low reconstruction accuracy requirements, but it is not suitable for fields with high reconstruction accuracy requirements. Gorodnitsky proposed a sparse signal reconstruction algorithm based on the wavelet transform [7]. Daubechies proposed a sparse reconstruction algorithm based on logic and iterative weighted least

squares minimization [8]. Both algorithms belong to non-convex optimization algorithms, which can improve the reconstruction accuracy to a certain extent but whose improvement effect is limited. Daubechies et al. proposed convex optimization algorithms to reconstruct sparse signals in wireless sensor networks [9]. This method transforms the sparse signal recovery problem into a convex optimization problem and solves the minimum value of the objective function. The results obtained are globally optimal, and the reconstruction accuracy is good and applicable to the realm with high reconstruction accuracy. It requires fewer measurements, but the computational complexity is very high. A typical convex optimization algorithm is the basis pursuit (BP) algorithm [10]. The BP algorithm has high reconstruction accuracy but a low reconstruction rate. William et al. proposed a gradient-based sparse recovery method [11], which has high reconstruction accuracy and a fast first-order convergence rate.

Ma Duxiang proposed to apply Newton's method to a sparse signal reconstruction algorithm [12]. Newton's search direction is used to replace the steepest descent direction and obtain the second-order convergence rate, which improves the reconstruction rate of the sparse signal to a certain extent. Newton's method requires the inverse of the Hessian matrix, which requires large computational quantities and can improve the reconstruction rate but is limited. The computational quantities are the complexity and time of the algorithm. The longer the algorithm runs, the greater the computational quantities. Chen Fenghua used the BFGS [13] algorithm to solve the problem of large computational quantities in wireless sensor networks. The BFGS algorithm replaces the inverse of the Hessian matrix with an approximate matrix H_k by iteration, which reduces the amount of computation. However, the BFGS algorithm needs to calculate and store the $n \times n$ matrix H_k. When dimension n is large, the reduced computation amount is limited. Therefore, the L-BFGS [14] quasi-Newton algorithm is proposed in this paper to solve the problem of sparse signal reconstruction in wireless sensors based on compressed sensing. The L-BFGS algorithm does not need to calculate and store H_k directly but only needs to store m (m<<n) vector pairs (s_k,u_k). This solves the shortcoming of the BFGS quasi-Newton algorithm, which must calculate and store the matrix H_k directly; the size of H_k is $n \times n$. In Section 5, we prove the feasibility of using the L-BFGS quasi-Newton algorithm to solve the problem of sparse signal reconstruction in wireless sensors based on compressed sensing.

2. Signal Reconstruction Algorithm Based on Newton Method

In WSNs, it is known that sparse signals $\overline{x} \in R^n$, and the matrix $A \in R^{m \times n}$ (m<<n) [13]. $y \in R^m$ is the observation vector. They satisfy the following relation:

$$y = A\overline{x}. \tag{1}$$

How to reconstruct the original signal $\overline{x}$ from a linear system is the core problem of sparse signal reconstruction using compressed sensing in wireless sensor networks. Equation (1) is underdetermined and has infinite sets of solutions.

In the compressed sensing problem, if the original signal $\overline{x}$ is known to be sparse under a certain transformation, the Equation (1) can be transformed into an L_0 norm minimization problem, and the signal is reconstructed by solving the L_0 norm minimization problem:

$$\min_x \|x\|_0, s.t. Ax = y, \tag{2}$$

the L_0 norm here is the number of nonzero elements of the vector.

Research shows that sparse signals can be accurately reconstructed by solving the linear optimization of the L_0 norm minimization problem. However, the L_0 norm problem is NP-hard. In other words, the solution to the L_0 norm minimization problem needs to enumerate all the permutations and combinations of non-zero values in the original signal. Under certain conditions, the L_1 norm minimization problem is the optimal convex approximation of the L_0 norm minimization problem [15], and the solution to the L_1 norm

minimization problem is simple. Therefore, solving the L_0 norm minimization problem is transformed into solving the L_1 norm minimization problem:

$$\min_x \|x\|_1, s.t. Ax = y. \tag{3}$$

The L_1 norm minimization problem is a convex optimization problem. There are many methods to solve convex optimization problems, and Newton's method is one of them. Newton's method uses the first-order matrix and second-order Hessian matrix to approximate the objective function quadratic, and the result is faster than other convex optimization algorithms.

In practice, it is difficult to directly solve the linear programming problem (3). Since matrix A is very large and complex to calculate. To simplify the calculation, problem (3) can be transformed into an L_1 regularized least squares problem [16].

$$\min_x \lambda\|x\|_1 + \frac{1}{2}\|y - Ax\|_2^2, \tag{4}$$

where λ is a regularization parameter.

Since $\|x\|_1$ is a nonsmooth convex function for x, the subgradient basic approach is not effective in solving the problem (4). The L_1 minimization problem (3) can be solved by solving a suitable smooth form of problem (4).

According to the literature [17], using the Nesterov smoothing technique, $\|x\|_1$ is smoothed into the following smooth function:

$$f(x) = \sum_{i=1}^{n} H(x(i)),$$

where:

$$H(u) = \begin{cases} \frac{u^2}{2\tau}, & |u| \le \tau; \\ |u| - \frac{\tau}{2}, & |u| > \tau \end{cases},$$

H(u) is the Huber penalty function, τ is a smooth parameter, see literature [18] for relevant research.

The smooth function $f(x)$ is convex, the gradient $\nabla f\ (x)$ is Lipchitz continuous, and its components are:

$$\nabla f(x(i)) = \begin{cases} \frac{x(i)}{\tau}, & |x(i)| \le \tau \\ sign(x(i)), & |x(i)| > \tau \end{cases} \quad (i = 1, 2, 3, \ldots\ldots, n),$$

which is:

$$sign(v) = \begin{cases} 1, & x > 0; \\ 0, & x = 0; \\ -1, & x < 0. \end{cases}$$

For ease of calculation, let $F(x) = \lambda\|x\|_1 + \frac{1}{2}\|y - Ax\|_2^2$. Convert the L_1 regularized least squares problem (4) into an unconstrained smooth convex programming problem:

$$\min F(x) = \min_x \lambda\|x\|_1 + \frac{1}{2}\|y - Ax\|_2^2, \tag{5}$$

where $F(x)$ is differentiable when Newton's method is adopted, the search direction of $F(x)$ is:

$$d_k = -\left[\nabla^2 F(x)\right]^{-1} \nabla F(x). \tag{6}$$

The gradient of $F(x)$ is:

$$\nabla F(x) = \lambda \nabla f(x) + A^T(Ax - y).$$

$\nabla F(x)$ is Lipchitz continuous, for the convenience of record, let:

$$s_k = x_{k+1} - x_k,$$

$$g_k = \nabla F(x_k),$$

$$u_k = \nabla F(x_{k+1}) - \nabla F(x_k) = g_{k+1} - g_k.$$

x_k is the iteration point of step k, x_{k+1} is the iteration point of step $k+1$, s_k is the iteration difference of adjacent iteration points, g_k is the gradient of $F(x)$, u_k is the gradient difference of $F(x)$.

Then, the search direction of Newton's method can be simplified as follows:

$$d_k = -\left[\nabla^2 F(x_k)\right]^{-1} g_k.$$

When solving the problem (5), the iteration formula of Newton's method is:

$$x_{k+1} = x_k + d_k.$$

When the Newton method is used to solve the sparse signal reconstruction problem in wireless sensor networks, its advantage is that it has a second-order convergence rate and that the reconstruction rate is higher than that of other convex optimization algorithms. The disadvantage of Newton's method is that it needs the inverse of the second-order Hessian matrix and a large amount of computation, which will reduce the reconstruction rate to a certain extent. In addition, it cannot guarantee that the search direction is the descending direction of the objective function $F(x)$ at x when $\nabla^2 F(x_k)$ is not positive definite.

Based on the existing problems of the Newton method, this paper proposes the L-BFGS quasi-Newton method to solve the problem of sparse signal reconstruction in wireless sensor networks. In Section 5, it is proven whether the L-BFGS quasi-Newton method can solve the sparse reconstruction problem of wireless sensor networks.

3. Principles

The L-BFGS quasi-Newton method is an improvement of the BFGS quasi-Newton method. The L-BFGS quasi-newton method uses the information of the objective function value and the first derivative to construct a suitable H_k to approximate the inverse of the Hessian matrix without directly calculating and storing the matrix H_k. The complexity of the L-BFGS algorithm is lower than that of the BFGS algorithm, and the convergence speed of the Newton method is maintained. The constructed H_k is positive definite, which ensures that the search direction is the descending direction of the target function $F(x)$ at x.

Before starting the L-BFGS algorithm, we first recall the famous BFGS method. An important problem in the BFGS quasi-Newton method is constructing a suitable matrix B_k, which makes B_k is an approximate matrix of the Hessian matrix. B_k needs to meet the following conditions:

(1) For all k, B_k is symmetric and positive definite, ensuring that the direction generated by the algorithm is the descending direction of the objective function $F(x)$ at x_k;

(2) The updating rule of a matrix B_k is relatively simple.

In this paper, the search direction d_k is obtained by solving the following linear equation:

$$B_k d_k = -g_k.$$

The updating rule of the matrix B_k is:

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{u_k u_k^T}{u_k^T s_k}. \tag{7}$$

When a suitable B_k has been constructed, H_k satisfies the condition $B_{k+1} = H_{k+1}^{-1}$. The Sherman-Morrison Woodbury [19] formula is used twice to obtain the updated formula of the matrix H_k is:

$$H_{k+1} = (I - \frac{s_k u_k^T}{u_k^T s_k}) H_k (I - \frac{u_k s_k^T}{u_k^T s_k}) + \frac{s_k s_k^T}{u_k^T s_k}, \tag{8}$$

where H_k is the approximation of the inverse of the Hessian matrix and I is the identity matrix of order n, H_k should satisfy the following conditions:

(1) H_k satisfies the Quasi-Newton equation:

$$B_{k+1} s_k = y_k \text{ or } H_{k+1} y_k = s_k;$$

(2) H_k is a positive definite symmetric matrix;

(3) The update from H_k to H_{k+1} is a low-rank update.

The specific process of the BFGS algorithm is shown in the literature [13]. It is known that the curvature condition $u_k^T s_k > 0$ can guarantee the positive definiteness of H_{k+1} when H_k is positive definite. From Formula (8), an efficient recursive procedure is derived to compute the matrix-vector product H_k. For completeness, we describe the L-BFGS recursive procedure for H_k in Algorithm 1.

Algorithm 1: L-BFGS two-loop recursion

1. Input: $s_k = [s_{k-m}, s_{k-m+1}, ..., s_{k-1}]$ and $u_k = [u_{k-m}, u_{k-m+1}, ..., u_{k-1}]$;
2. $q \leftarrow g_k$
3. **for** $i = k-1$: $k-m$ **do**
4. $\quad \alpha_i \leftarrow \rho_i s_i^T q$
5. $\quad q = q \leftarrow \alpha_i u_i$
6. **end**
7. $r \leftarrow H_k^{(0)} q$
8. **for** $i = k-1$: $k-m$ **do**
9. $\quad \beta \leftarrow \rho_i u_i^T d_k$
10. $\quad r \leftarrow r + s_i(\alpha_i - \beta)$
11. **end** $r = -H_k g_k$
12. Return the result

The two-loop recursive scheme can find the descent direction only through a simple calculation. Hence, it is cheap to generate the search direction by using Algorithm 1.

4. Algorithm

In the L-BFGS method, H_k no longer needs to be calculated and stored directly. Instead, the m latest vectors (s_k,u_k) saved in the memory are used to update the H_k, which only needs to store matrices of the size $m \times n$, saving a lot of computation cost and improving the reconstruction rate. The specific step of the L-BFGS algorithm is Algorithm 2.

In the L-BFGS algorithm, the Wolfe line search is used to obtain the step size α_k. Wolfe line search belongs to a kind of inexact linear search. When using Wolfe line search to find the step size α_k, α_k is required to satisfy the following:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \rho_1 \alpha_k g_k^T d_k. \tag{9}$$

To make (9), flexible, we employ the non-monotone reduction condition on the:

$$g(x_k + \alpha_k d_k)^T d_k \geq \rho_2 g_k^T d_k, \tag{10}$$

the parameterρ_1 and ρ_2 satisfies the conditions ($0 < \rho_1 < \rho_2 < 1$).

Algorithm 2: Limit memory BFGS Algorithm(L-BFGS)

1.Given: $\varepsilon > 0$, $k = 0$, $\sigma_1 \in (0, \frac{1}{2})$, $\rho_1 \in (0,1)$, $t \in (0,1)$;
2. Initialization : $x_0 \in R^n$; $H_0 = I$.
3. **while** $\|g_k\| > \varepsilon$ **do**
4. Calculate the gradient g_k
5. Compute $d_k = -H_k g_k$ from Algorithm 1
6. **while** $f(x_k + \alpha_k d_k) - f(x_k) \le \rho_1 \alpha_k g_k^T d_k$ **do**
7. $\alpha = t\alpha$
8. **end**
9. $\alpha_k = \alpha$, $x_{k+1} = x_k + \alpha_k d_k$
10. Update H_{k+1} by s_k and u_k
11. Set $k = k + 1$
12. **end**

Finally, compared with the BFGS algorithm, the L-BFGS algorithm does not need to calculate and store H_k directly; it only needs to store matrices of the size $m \times d$ and calculate H_k by matrices of storage. The high reconstruction rate is improved by discarding some vectors and sacrificing a little reconstruction accuracy. Therefore, compared with the BFGS quasi-Newton method, the L-BFGS quasi-Newton method greatly reduces the computational complexity and makes the reconstruction rate of a sparse signal faster.

5. Verify the Feasibility of the L-BFGS Algorithm

According to Section 2, the sparse signal reconstruction problem of wireless sensor networks can be transformed into the L_1 norm minimization problem. When the solution is the optimal solution to the L_1 norm minimization problem, the sparse signal can be accurately reconstructed. In other words, when algorithm 2 meets global convergence, it is feasible.

First, we give some useful inequalities in the following lemma.

Lemma 1. *Let B_k be symmetric and positive definite, and B_{k+1} is determined by the L-BFGS correction Equation (7), then the necessary and sufficient condition for B_{k+1} symmetric and positive definite is $u_k^T s_k > 0$.*

Lemma 1 shows that, if the initial matrix B_0 is symmetric and positive definite and maintains $u_k^T s_k > 0$ ($\forall k \ge 0$) during the iteration. Then, the matrix sequence $\{ B_{k+1} \}$ generated by the L-BFGS correction formula is symmetric and positive definite, so that the equation $B_k d_k = -g_k$ has a unique solution d_k**.** It is easy to deduce that

$$d_k^T B_k d_k = -g_k^T d_k.$$

Since $d_k^T B_k d_k > 0$, so $g_k^T d_k < 0$. d_k is the descending direction of the target function *F(x)* at x_k.

Lemma 2. *If the Wolfe search criterion is adopted in the L-BFGS corrected quasi-Newton algorithm, there is $u_k^T s_k > 0$.*

Proof. Since $u_k^T s_k = (g_{k+1} - g_k)^T \alpha_k d_k = \alpha_k (g_{k+1}^T d_k - g_k^T d_k)$, according to step 5 of algorithm 2, we have

$$g_{k+1}^T d_k \ge \rho_2 g_k^T d_k,$$

it is easy to deduce that

$$u_k^T s_k \ge \alpha_k (\rho_2 - 1) g_k^T d_k.$$

Since $\rho_2 < 1$ and $g_k^T d_k < 0$, so $u_k^T s_k > 0$. B_{k+1} is positively definite. The search direction generated by the algorithm is guaranteed to be the descending direction of the objective function $F(x)$ at x_k, making the algorithm work normally.□

Lemma 3. *If the smooth function of the L_1 norm $\|x\|_1$ obtained by the smoothing technique is:*

$$f(x) = \sum_{i=1}^{n} H(x(i)),$$

where:

$$H(x(i)) = \begin{cases} \frac{x^2(i)}{2\tau}, & |x(i)| \le \tau, \\ |x(i)| - \frac{\tau}{2}, & |x(i)| > \tau, \end{cases}$$

then it can be proven that $f(x) \le \|x\|_1$.

Proof. when $|x(i)| \le \tau, f(x) = \sum_{i=1}^{n} \frac{x^2(i)}{2\tau} = \frac{1}{2\tau}\sum_{i=1}^{n} x^2(i) \le \frac{1}{2\tau}\tau\sum_{i=1}^{n}|x(i)| = \frac{1}{2}\|x\|_1 \le \|x\|_1$, when $|x(i)| > \tau, f(x) = \sum_{i=1}^{n}(|x(i)| - \frac{\tau}{2}) = \|x\|_1 - \frac{n\tau}{2} \le \|x\|_1$, we have that: $f(x) \le \|x\|_1$.□

From Lemma 3 and the mechanism of Algorithm 2, we immediately obtain the following theorem.

Theorem 1. *Let $\{x_k\}$ denote the sequence generated by L-BFGS quasi-Newton method iteration. When the smoothing parameter of $\|x\|_1$ satisfies $\tau_k \to 0$, then $\{x_k\}$ is a bounded sequence. Let x^* denote any limit point of $\{x_k\}$, then x^* is the optimal solution to the L_1 norm minimization problem.*

Prove the smooth parameter τ_k for the step k, make $x_k^* = \arg\min_x f(x)$ denotes the minimum point of the objective function $F_k(x)$ at step k, and z denote any vector satisfying $Az = y$.

Because $x_k^* = \arg\min_x F_k(x)$, so $F_k(x_k^*) \le F_k(z)$, also because $Az = y$, so:

$$F_k(z) = \lambda f_k(z) + \frac{1}{2}\|Av - y\|_2^2 = \lambda f_k(z). \tag{11}$$

And from Lemma 3, $f_k(z) \le \|z\|_1$, then:

$$F_k(x_k^*) \le F_k(z) = \lambda f_k(z) \le \lambda\|z\|_1. \tag{12}$$

Since $F_k(x) = \lambda f_k(x) + \frac{1}{2}\|Ax - y\|_2^2$, $\|Ax - y\|_2^2 \ge 0$, so:

$$F_k(x) \ge \lambda f_k(x). \tag{13}$$

In combination with Equations (11) and (12), the smoothness of $f_k(x)$ is:

$$\|x_k\|_1 \le f_{\tau_k}(x_k) + \frac{n\tau_k}{2} \le \frac{1}{\lambda}F_k(x_k) + \frac{n\tau_k}{2} \le \frac{1}{\lambda}\lambda\|z\|_1 + \frac{n\tau_k}{2} = \|z\|_1 + \frac{n\tau_k}{2}.$$

In addition, when the smoothing parameter $\tau_k \to 0$, $\{x_k\}$ is a bounded sequence. On the other hand, it can also be proved that the limit point x^* of the sequence $\{x_k\}$ is feasible. The limit point x^* is the KKT point of the L_1 norm minimization problem. Since $\|x\|_1$ is convex and problem (3) is a convex programming problem, the KKT condition is sufficient for optimality, so x^* is the optimal solution to the $\|x\|_1$ norm minimization problem (3). It can be seen that the sparse signal of the wireless sensor network can be accurately reconstructed by the L-BFGS quasi-Newton method.

Corollary 1. *Let the L_1 norm minimization problem have a unique optimal solution, and let $\{x_k\}$ represent the sequence generated by an iteration of the L-BFGS method. When $k \to \infty$ and $\tau_k \to 0$, the x_k converges to x*, where $x^* = arg\ min\{\|x\|_1 : Ax = y\}$.*

6. L-BFGS Quasi-Newton Method Steps

The L-BFGS quasi-Newton method is developed from the BFGS quasi-Newton method. The basic idea of the BFGS quasi-Newton method is to use the identity matrix I to gradually approximate the H matrix, which avoids directly calculating the inverse of the Hessian matrix. However, H_k should be stored each time. H_k is very large, and the reconstruction rate of the algorithm is lost by direct calculation. According to Equation (8), each iteration of H_k is obtained by iterating curvature information (s_k,u_k). Since H_k cannot be stored easily, then store all the curvature information (s_k,u_k). This kind of storage saves memory space and improves the speed of the algorithm.

When the number of iterations is very large, all (s_k,u_k) cannot be saved. The reconstruction rate can be improved by discarding some original (s_k,u_k). Assuming that the number of storage vectors set is m. When the iteration is $m + 1$ times, (s_1,u_1) will be thrown away, and when the iteration is $m + 2$ times, (s_2,u_2) will be thrown away. By parity of reasoning, only the latest m-group (s_k,u_k) is retained. In this way, although the reconstruction accuracy is lost, the memory is saved, the algorithm complexity is reduced, and the reconstruction rate is improved. So the L-BFGS algorithm can be understood as another optimization of the BFGS algorithm.

When the L-BFGS quasi-Newton method is used to solve the sparse signal reconstruction problem in wireless sensor networks, the specific steps are as follows:

Step 1: First initialize the variable, select the initial point x_0, and define the operation error ε. The general value of ε is between 0 and 1, and the initial value of the H matrix is the identity matrix I of order n;

Step 2: Let $k = 1$ and calculate the gradient g_k at this time;

Step 3: Calculate g_{k+1}. If the norm of g_{k+1} satisfies the operating error defined, namely $\|g_{k+1}\| \leq \varepsilon$, it indicates that the cut-off condition is met and the algorithm is terminated. The iteration point x_{k+1} is the optimal solution. If the cut-off conditions are not met, it is necessary to go to step 4 and continue to solve the optimal solution;

Step 4: Calculate the search direction d_k and the step factor α_k, and update x_{k+1}; to ensure that the search direction is correct. There are many methods to calculate the step factor α_k. Here, the inexact Wolfe line search is used to solve the step factor α_k. The step size can be guaranteed to be larger than 0. In the correct search direction, the x_{k+1} is closer and closer to the optimal solution;

Step 5: If $k \geq m$, keep the latest curvature information (s_k,u_k), and delete (s_{k-m},u_{k-m}). The L-BFGS quasi-Newton method will retain m bits of data. When it exceeds m bits, it will delete the curvature information before m bits to improve the calculation speed. This is also the difference between the BFGS quasi-Newton method;

Step 6: Let $k = k + 1$; transfer to step 3. Start iterating until the optimal solution is obtained and the algorithm stops.

Special remarks: In step 1, for the initial value x_0, the initial value x_0 of iteration should be selected near the root to ensure the convergence of the iterative process. The most common way is to choose the initial value by using Newton's convergence theorem. In step 4, the correctness of the search direction d_k should be ensured. If the search direction is not the descending direction at x, the iterative point will deviate more and more from the optimal point, and the sparse signal cannot be accurately reconstructed. There are many ways to calculate the descending direction, d_k. In this paper, a two-loop recursive algorithm is adopted to find the search direction, which can ensure the accuracy of the search direction and the two-loop recursive algorithm is simple. In step 5, if $k \geq m$ is satisfied, calculate the new curvature information and discard the curvature information before m-bit

(s_{k-m},u_{k-m}). By giving up small reconstruction accuracy in exchange for a faster reconstruction rate. If $k \geq m$ is not satisfied, directly calculate (s_k,u_k). In step 6, let $k = k + 1$ and

start a new iteration; no initialization is required. The result of the previous iteration is its initial value, and the iteration continues until the optimal solution is obtained.

The flow chart of the L-BFGS quasi-Newton method is shown in Figure 1 as follows:

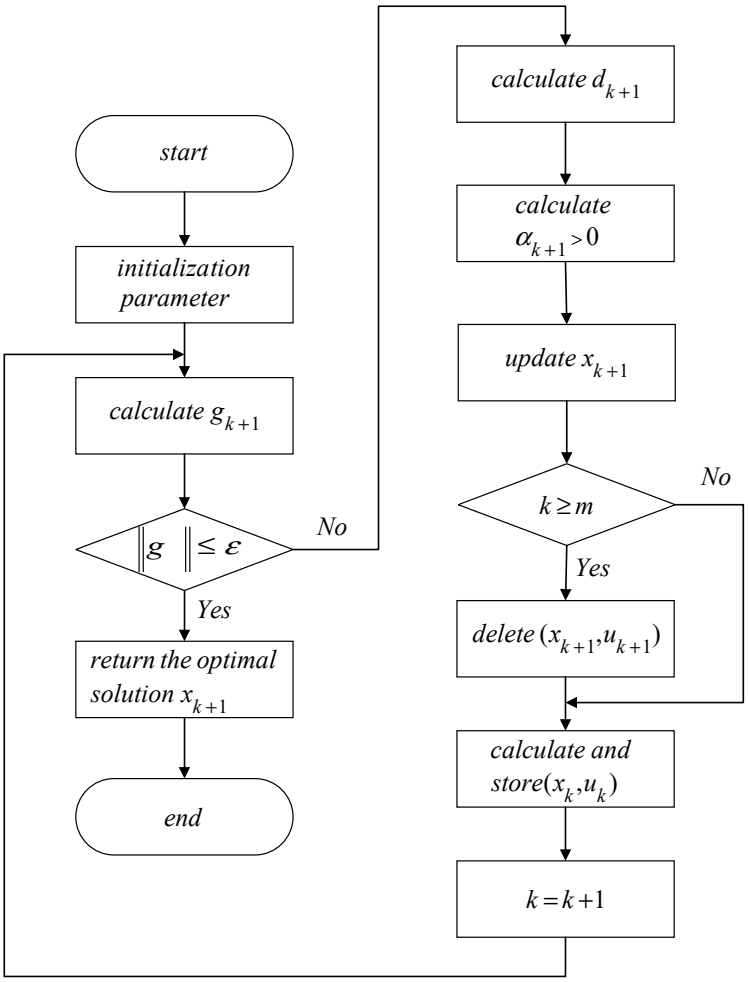


Figure 1. The flow chart of L-BFGS quasi-Newton method.

7. Experimental Simulation

To verify the effectiveness of the L-BFGS quasi-Newton method in wireless sensor networks for signal reconstruction, simulation experiments are carried out. The Gaussian matrix can satisfy the constraint isometry with high probability. In the experiment, the measurement matrix $\Phi \in R^{M \times N}$ with Gaussian distribution is used as the observation matrix, and the Gaussian sparse signal $x \in R^N$ with variable sparsity is used as the original signal. The value range of sparsity K is [1,70].

The simulation experiment in this paper takes the reconstruction accuracy and reconstruction time of each algorithm for the original signal as the main evaluation criteria, and the unit of time is a second (s). In the simulation experiment, each experiment is carried out 1000 times, and the average reconstruction accuracy and average reconstruction time of the 1000 experiments are taken as the final reconstruction accuracy and reconstruction time.

This paper mainly compares the reconstruction accuracy and reconstruction time of the BP algorithm, the NSHTP [20] algorithm, the BFGS algorithm, and the L-BFGS algorithm under conditions of sparsity. When the data changes, the difference in reconstruction accuracy is small. This paper mainly analyzes the reconstruction accuracy of two sets of data: n = 512, m = 256, and n = 256, m = 128. The reconstruction time is greatly affected by the data size. To better compare the reconstruction time of each algorithm, four groups of

data are selected to compare and test: n = 128, m = 64; n = 256, m = 128; n = 512, m = 256; n = 1024, m = 512.

First, there are two simulation diagrams about the reconstruction accuracy. In Figure 2, n = 256 and m = 128. As can be seen from the figure, there is little difference in reconstruction accuracy between the four algorithms. The reconstruction rate of the four algorithms is high. The BP algorithm has the worst reconstruction accuracy. The reconstruction accuracy of the BP algorithm starts to decrease when K = 35. The BP algorithm cannot reconstruct the signal when K = 62. The BFGS algorithm has the best reconstruction accuracy, and the difference between the proposed algorithm and the BFGS algorithm is very small. Both algorithms begin to reduce the reconstruction accuracy when K = 40 and cannot reconstruct the signal when K = 72. In general, the reconstruction accuracy of the BFGS algorithm is slightly higher than that of the proposed algorithm, but the reconstruction accuracy of the proposed algorithm is higher than that of the NSHTP algorithm and the BP algorithm.

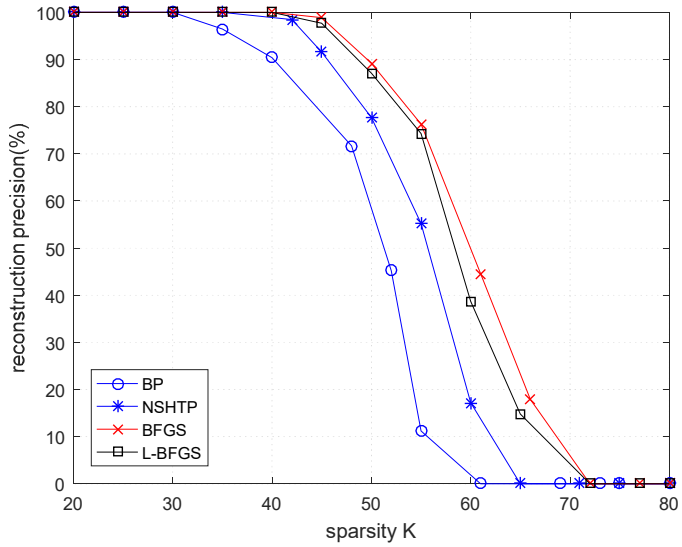


Figure 2. Reconstruction precision of each algorithm under different sparsity (n = 256, m = 128).

In Figure 3, n = 512 and m = 256. At this time, the reconstruction accuracy is slightly different for n = 256 and m = 128; however, the reconstruction accuracy of all algorithms has deteriorated. It shows that the larger n and m, the worse the reconstruction accuracy. The difference between the proposed algorithm and the BFGS algorithm is still very small and within an acceptable range.

Secondly, there are four simulation diagrams about the reconstruction rate. In Figure 4, n = 128 and m = 64. It can be seen from Figure 4 that the reconstruction time of the BP algorithm is higher than the other three algorithms, which require the longest reconstruction time and the lowest reconstruction rate. The reconstruction time of the NSHTP algorithm and BFGS algorithm is between the BP algorithm and the proposed algorithm, and the proposed algorithm requires the shortest reconstruction time and the highest reconstruction rate. The reconstruction time of the BFGS algorithm is twice that of the proposed algorithm, and the reconstruction rate of the proposed algorithm is higher than that of the BFGS algorithm. As can be seen from the figure, when the data is short, the reconstruction rate of all algorithms is high, and the sparsity has little influence on the reconstruction time. There is no obvious mutation point when the sparsity increases in the reconstruction time.

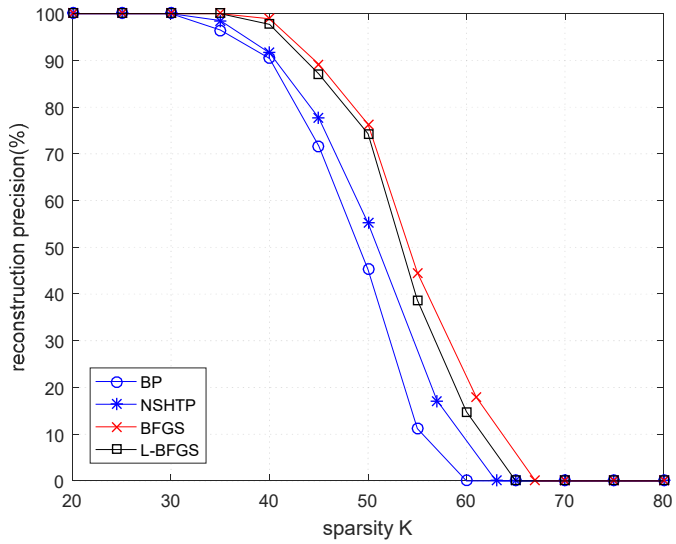


Figure 3. Reconstruction precision of each algorithm under different sparsity (n = 512, m = 256).

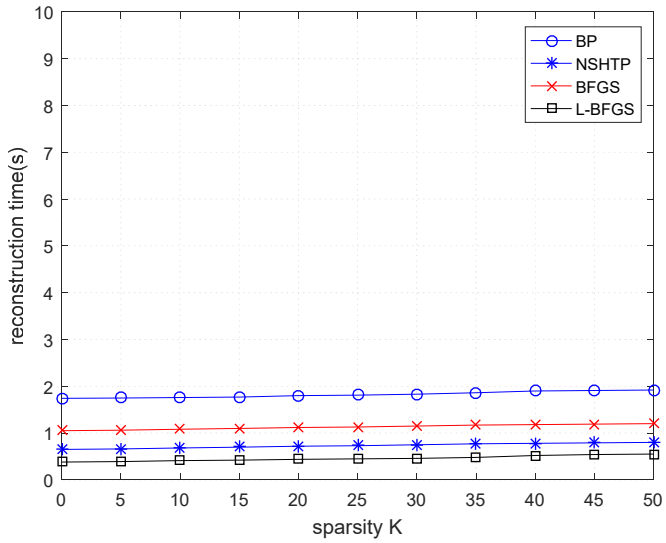


Figure 4. Reconstruction time of each algorithm under different sparsity (n = 128, m = 64).

In Figure 5, n = 256 and m = 128. The reconstruction time by the four algorithms is significantly higher than that when n = 128 and m = 64. It indicates that with the increase in data length, the time for signal reconstruction is longer and the reconstruction rate is lower. As can be seen from Figure 5, the proposed algorithm has the shortest reconstruction time and the fastest reconstruction rate. The reconstruction time of the BP algorithm is the longest, and the reconstruction rate is the worst. The reconstruction times of the NSHTP algorithm and the BFGS algorithm are between the BP algorithm and the proposed algorithm. In general, when the signal length increases, the reconstruction rate of all algorithms will be affected, but the reconstruction rate of the proposed algorithm is the fastest, which is significantly higher than the BFGS algorithm.

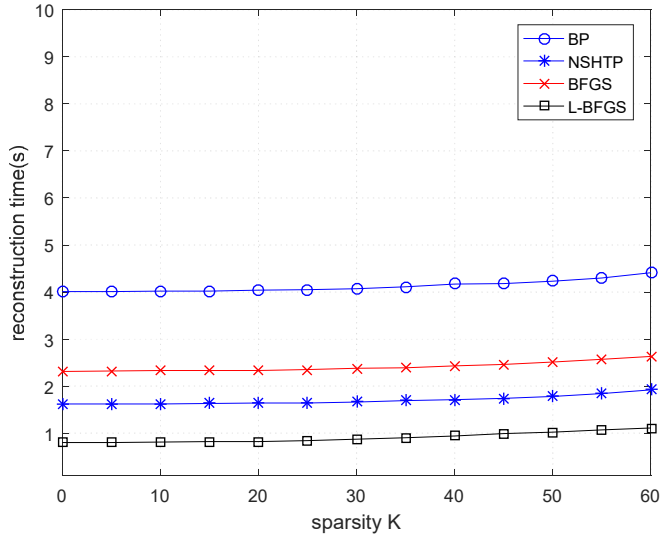


Figure 5. Reconstruction time of each algorithm under different sparsity (n = 256, m = 128).

In Figure 6, n = 512 and m = 256. All algorithms require longer reconstruction times, which get longer as the length of the data increases. The proposed algorithm still has the shortest reconstruction time and the fastest reconstruction rate. The BP algorithm still has the worst reconstruction rate. When K = 40, the BP algorithm has a small mutation point, and when K = 45, the NSHTP algorithm also has a small mutation point. Both the proposed algorithm and the BFGS algorithm have no obvious mutation point. It indicates that when the data length increases, the sparsity has little influence on the BP algorithm and the NSHTP algorithm but no obvious influence on the proposed algorithm and the BFGS algorithm. In general, when n = 512 and m = 256, the proposed algorithm has the shortest reconstruction time and the fastest reconstruction rate among the four algorithms. It is less affected by sparsity, and the reconstruction rate is significantly higher than the BFGS algorithm.

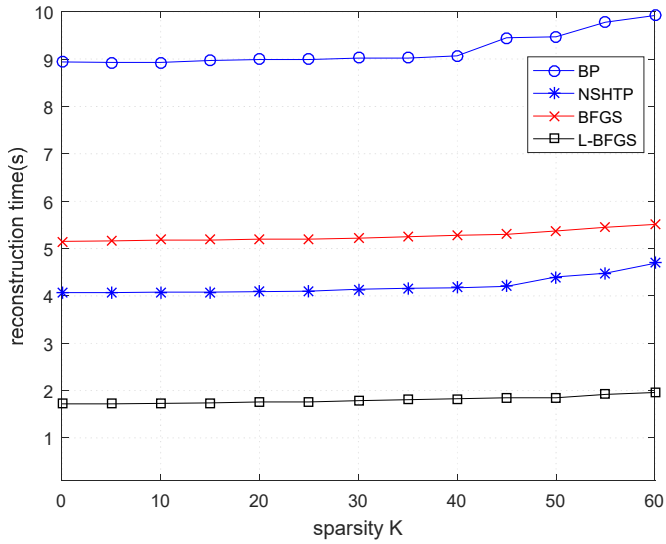


Figure 6. Reconstruction time of each algorithm under different sparsity (n = 512, m = 256).

In Figure 7, n = 1024 and m = 512.The exact reconstruction time of all algorithms becomes longer, but the reconstruction time of the proposed algorithm is the shortest and the reconstruction rate is the fastest. The BP algorithm has the longest reconstruction time and the worst reconstruction rate. As shown in Figure 7, with the increase in sparsity, the BP algorithm has a mutation point when K = 30. When K = 35, the NSHTP algorithm

has a mutation point, and the BFGS algorithm has a small mutation point. When K = 45, small mutation points appear in the proposed algorithm. It indicates that when n = 1024 and m = 512, sparsity has an impact on all algorithms. The BP algorithm and the NSHTP algorithm have a greater impact, and the proposed algorithm and the BFGS algorithm have a smaller impact. In general, the reconstruction rate of the proposed algorithm is the fastest and most stable, which is significantly better than the BFGS algorithm in the reconstruction rate.

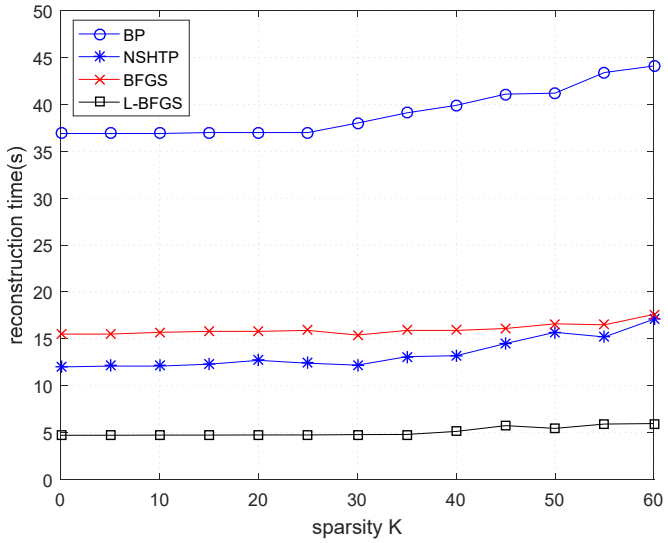


Figure 7. Reconstruction time of each algorithm under different sparsity (n = 1024, m = 512).

Finally, by comparing n = 128, m = 64; n = 256, m = 128; n = 512, m = 256; n = 1024, m = 512, the reconstruction time of four groups of data was simulated. It can be found that the larger the length of data, the longer the time for signal reconstruction, and the worse the rate for signal reconstruction. When n = 512, m = 256, and K = 25, the signal reconstruction time of the BP algorithm is 9.11 s, the NSHTP algorithm is 4.23 s, the BFGS algorithm is 5.17 s, and the proposed algorithm is 1.93 s. When n = 1024, m = 512, and K = 25, the signal reconstruction time of the BP algorithm becomes 37.25 s, the NSHTP algorithm becomes 12.57 s, the BFGS algorithm becomes 15.93 s, and the proposed algorithm becomes 4.84 s. It can be seen that when the data length and matrix size are doubled, the reconstruction time of the BP algorithm is more than 4 times the growth of the data, the reconstruction times of the NSHTP algorithm and BFGS algorithm have been tripled, and the proposed algorithm becomes 2.5 times the growth of the data. It shows that when n and m grow, the reconstruction rate of the proposed algorithm is the most stable and fastest. Through the simulation of reconstruction accuracy and reconstruction rate, it can be seen that the L-BFGS algorithm is slightly worse than the BFGS algorithm in reconstruction accuracy, but the reconstruction rate is significantly higher than the BFGS algorithm. In general, the L-BFGS algorithm is superior to the BFGS algorithm and is more suitable for sparse signal reconstruction in wireless sensor networks.

8. Conclusions

Aiming at the problem of a low reconstruction rate when the Newton method is used to reconstruct sparse signals in wireless sensor networks, this paper proposes to use the L-BFGS quasi-Newton algorithm to improve the reconstruction rate. The L-BFGS does not need to store and calculate H_k directly; only two matrices of size $m \times n$ need to be stored. This algorithm, by discarding some vectors and sacrificing some reconstruction accuracy, greatly reduces the reconstruction time and improves the reconstruction rate compared with the BFGS algorithm. At the same time, the simulation experiment is carried out to compare the reconstruction accuracy and reconstruction time of the proposed algorithms,

the BFGS algorithm, the NSHTP algorithm, and the BP algorithm. Experiments show that the reconstruction accuracy of the proposed algorithm is higher than that of the BP algorithm and the NSHTP algorithm but slightly lower than that of the BFGS algorithm. However, the reconstruction rate of the proposed algorithm is the fastest and most stable among the four algorithms, which is two times higher than the BFGS algorithm. Therefore, the proposed algorithm has a high reconstruction rate and certain advantages in terms of reconstruction accuracy. It is a better reconstruction algorithm than the BFGS algorithm.

Author Contributions: Conceptualization, X.L.; methodology, X.L.; software, Y.W.; validation, X.L., L.Z. (Liyu Zhang), Q.W. and D.L.; writing—original draft preparation, C.Y., L.Z. (Lanfei Zhao) and J.W.; writing—review and editing, C.Y.; project administration, X.L.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014.

Acknowledgments: The authors acknowledge the Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Benelhouri, A.; Idrissi, S.H.; Antari, J. Evolutionary routing based energy-aware multi-hop scheme for lifetime maximization in heterogeneous WSNs. *Simul. Model. Pract. Theory* **2022**, *116*, 102471–102487. [CrossRef]
2. AlZobi, F.I.; AlZubi, A.A.; Yurii, K.; Alharbi, A.; Alanazi, J.M.; Smadi, S. An Optimal Scheme for WSN Based on Compressed Sensing. *Comput. Mater. Contin.* **2022**, *72*, 1053–1069. [CrossRef]
3. Deng, Q.; Zeng, H.; Zhang, J.; Tian, S.; Cao, J.; Li, Z.; Liu, A. Compressed sensing for image reconstruction via back-off and rectification of greedy algorithm. *Signal Process.* **2019**, *157*, 280–287. [CrossRef]
4. Sorokovikov, P.; Gornov, A. Combined non-convex optimization algorithms based on differential evolution, harmony search, firefly, and L-BFGS methods. *IOP Conf. Series: Mater. Sci. Eng.* **2021**, *1047*, 012077. [CrossRef]
5. Traoré, C.; Pauwels, E. Sequential convergence of AdaGrad algorithm for smooth convex optimization. *Oper. Res. Lett.* **2021**, *49*, 452–458. [CrossRef]
6. Alahari, R.K.; Satya, P.K.; Kishan, R. Low Complexity FFT Factorization for CS Reconstruction. *Int. J. Eng. Adv. Technol.* **2020**, *9*, 438–442.
7. Mohimani, G.H.; Babaie-Zadeh, M.; Gorodnitsky, I.; Jutten, C. Sparse Recovery using Smoothed ℓ;0; (SL0): Convergence Analysis. *CoRR* **2010**, *1001*, 5073–5087.
8. Alaifari, R.; Daubechies, I.; Grohs, P.; Thakur, G. Reconstructing Real-Valued Functions from Unsigned Coefficients with Respect to Wavelet and Other Frames. *J. Fourier Anal. Appl.* **2017**, *23*, 1480–1494. [CrossRef]
9. Daubechies, I.; Friese, M.; Mol, C. An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint. *Commun. Pure Appl. Math.* **2004**, *57*, 1413–1457. [CrossRef]
10. Chen, S.S.; Donoho, D.L.; Saunders, M.A. Atomic Decomposition by Basis Pursuit. *SIAM J. Sci. Comput.* **1998**, *20*, 33–61. [CrossRef]
11. Hager, W.W.; Phan, D.D.; Zhang, H.C. Gradient-Based Methods for Sparse Recovery. *Soc. Ind. Appl. Math.* **2011**, *4*, 146–165. [CrossRef]
12. Ma, D.X.; Zhang, M.H.; Meng, X. Fast smooth l_0 norm method for compressed sensing signal reconstruction. *Sci. Technol. Eng.* **2013**, *13*, 2377–2381.
13. Chen, F.H.; Li, S.A. BFGS correction Quasi-Newton method for large-scale signal recovery. *Chin. J. Math.* **2015**, *35*, 727–734.
14. Lu, L.B.; Wang, K.P.; Tan, H.D.; Li, Q.K. Three-dimensional magnetotelluric inversion using L-BFGS. *Acta Geophys. Off. J. Inst. Geophys. PAS Pol. Acad. Sci.* **2020**, *68*, 1049–1066. [CrossRef]
15. Jin, D.; Yang, Y.; Ge, T.; Wu, D. A Fast Sparse Recovery Algorithm for Compressed Sensing Using Approximate l0 Norm and Modified Newton Method. *Materials* **2019**, *12*, 1227. [CrossRef] [PubMed]
16. Yang, H.; Yang, X.; Zhang, F.; Ye, Q. Robust Plane Clustering Based on L1-Norm Minimization. *IEEE Access* **2020**, *8*, 29489–29500. [CrossRef]
17. Nam, N.M.; An, N.T.; Rector, R.B.; Sun, J. Nonsmooth Algorithms and Nesterov's Smoothing Technique for Generalized Fermat–Torricelli Problems. *SIAM J. Optim.* **2014**, *24*, 1815–1839. [CrossRef]
18. Duan, C.; Liu, Y.; Xing, C.; Wang, Z. Infrared and Visible Image Fusion Using Truncated Huber Penalty Function Smoothing and Visual Saliency Based Threshold Optimization. *Electronics* **2021**, *11*, 33. [CrossRef]

19. Yang, H.; Zhong, J.; Ma, B.L. The Sherman-Morrison-Woodbury Formula of Matrix Core Inverse and its application. *J. Jiangxi Univ. Sci. Technol.* **2021**, *42*, 98–102.
20. Meng, N.; Zhao, Y.-B. Newton-Step-Based Hard Thresholding Algorithms for Sparse Signal Recovery. *IEEE Trans. Signal Process.* **2020**, *68*, 6594–6606. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Detecting Human Falls in Poor Lighting: Object Detection and Tracking Approach for Indoor Safety

Xing Zi, Kunal Chaturvedi, Ali Braytee, Jun Li and Mukesh Prasad *

School of Computer Science, Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney 2007, Australia
* Correspondence: mukesh.prasad@uts.edu.au

Abstract: Falls are one the leading causes of accidental death for all people, but the elderly are at particularly high risk. Falls are severe issue in the care of those elderly people who live alone and have limited access to health aides and skilled nursing care. Conventional vision-based systems for fall detection are prone to failure in conditions with low illumination. Therefore, an automated system that detects falls in low-light conditions has become an urgent need for protecting vulnerable people. This paper proposes a novel vision-based fall detection system that uses object tracking and image enhancement techniques. The proposed approach is divided into two parts. First, the captured frames are optimized using a dual illumination estimation algorithm. Next, a deep-learning-based tracking framework that includes detection by YOLOv7 and tracking by the Deep SORT algorithm is proposed to perform fall detection. On the Le2i fall and UR fall detection (URFD) datasets, we evaluate the proposed method and demonstrate the effectiveness of fall detection in dark night environments with obstacles.

Keywords: fall detection; YOLOv7; dual illumination estimation; CNN; Deep SORT

1. Introduction

The World Health Organization states that falling has been ranked as the second leading cause of accidental death [1]. A fall is described as a rapid change from a normal state to a reclined or extended position of the whole body. It can be caused by discomfort or unsteadiness while standing [2]. Recent studies indicate that the death rate among the elderly due to falls is nearly three times higher than that of younger age groups [3,4]. Most older people, especially those living independently without a carer, fear severe injuries or even fatalities from falling because of delayed assistance. This demonstrates the importance of early warning and convenient management of falls. It is foreseeable that with practical and timely detection and warning mechanisms, the rate of severe injuries and even fatalities in falls, especially among the elderly, will be significantly reduced [5,6].

Most of the current fall detection and warning systems employ sensors such as barometers, accelerometers, gyroscopes, and inertial sensors [7,8]. These sensors are usually used in wearable devices, including a smartwatch, shoes, and necklace, to monitor users' body parameters to detect a fall. However, they have the drawback of requiring the devices to be put on the user's body, which makes it uncomfortable and infeasible to be worn constantly. Furthermore, sensor-based fall detection systems are expensive and have data privacy and security concerns.

With the advent of vision-based systems, computer vision has witnessed a growing trend in its applications. Notably, significant amounts of research are focused on vision-based systems to monitor any anomalies in body movement for fall detection [5,9–13]. Such methods can be divided into two categories based on the data used, namely, (1) RGB-based detection and (2) depth-based detection [14]. In this paper, we perform fall detection on an RGB-based dataset using a deep learning framework. Deep learning networks, in

particular, convolutional neural networks (CNNs), have emerged as a popular computation framework in computer vision [15–19]. Many deeper and more complicated networks are being developed to enable CNNs to deliver near-human accuracy in many applications, such as classification [20,21], detection [22–24], and segmentation [25,26]. A CNN-based deep learning approach on RGB-based datasets has been established as state-of-the-art for fall detection [12,13,27,28]. Although these methods enable flexible automatic feature extraction from images, they use only one keyframe, which may not be sufficient to adequately classify falls in videos. For example, a fall can be summarized using distinctive keyframes based on variations in position from standing or sitting to lying down [29]. Specifying the location of such keyframes for falls would be too tedious and prone to errors. Therefore, there is a need for a frame-to-frame object tracking mechanism to characterize the current state of the body.

This paper presents a novel deep-learning-based tracking-by-detection approach to detect falls in videos. Tracking-by-detection methods consist of: (1) an object detection algorithm that gives the object's bounding box coordinates for each frame, and (2) a tracking algorithm that decides if the newly detected object can be connected to the predicted position of existing tracks. The quality of both the detection algorithm and the data association algorithm heavily influences the effectiveness of object tracking [30]. Specifically, the proposed method integrates Simple Online and Realtime Tracking with a Deep association metric (Deep SORT) algorithm [31] with YOLOv7 [32] for fall detection. Performance is highly dependent on the light in the environment. In the real-world setting, fall detection can be challenging due to light changes and poor illumination. To improve the overall efficiency, we combine the deep-learning-based tracking-by-detection with an image enhancement algorithm. Our proposed hybrid method significantly reduces the false positive rates in the fall detection task.

The main contributions of this paper are as follows:

- A novel deep-learning-based approach for vision-based fall detection is proposed, integrating YOLOv7 for object detection and the Deep SORT algorithm for tracking and trajectory analysis.
- The proposed method incorporates dual illumination estimation, utilizing a Retinex-based image enhancement algorithm, to effectively tackle the issue of inconsistent lighting conditions and exposure levels.
- The effectiveness and superiority of the proposed approach over current state-of-the-art methods are extensively demonstrated, offering a robust solution for vision-based fall detection.

2. Related Works

Elderly care requires continuous monitoring by health care staff, which is costly, time-consuming, and even considered an intrusive process that disrupts individuals' routines and privacy [33]. Thus, an automatic fall detection system is required in healthcare services to provide cost-effective, efficient, and timely care to the person affected. Researchers have exhaustively explored many methods of fall detection to reduce the number of injuries caused by falls. These systems are broadly classified based on the device used to detect the fall, including wearable sensors, environment sensors, and vision-based sensors. In wearable devices, one or more sensors are put on various body parts of the person to detect and identify falls from activities of daily living (ADL). It detects any rapid increase in negative acceleration when the individual suddenly changes its position from standing to lying down [34,35]. For environment-based sensors, signals from acoustic [36], infrared [37], vibration [38] and pressure [39] are collected and analyzed to detect falls. These devices are less intrusive than wearable sensor-based devices and require less interaction with the individual, which reduces privacy and security concerns. However, environment-based sensors are prone to false alarms due to changes in the external environment. For example, different noises or other falling items in the room can impact the sensors' performance [40].

In recent years, the development of vision-based sensors has led to several computer vision-based approaches being proposed for fall detection. Vision-based approaches extract data from sensors which can be RGB-based, depth-based, thermal-based, or even a combination of these. The extracted data is then utilized by computer vision algorithms to detect any unusual changes related to the body trajectories, postures, or shapes to identify falls. The most common approach in computer vision is based on hand-crafted feature extraction [9,11]. Sehairi et al. [11] use a finite state machine (FSM) on a human silhouette extracted by background subtraction technique to determine if a fall happened. Albawendi et al. [9] perform fall detection based on hand-crafted features extracted from projection histogram, motion information, and human shape variation. Recently, deep learning methods have gained popularity over hand-crafted feature-based methods due to their ability to extract important features in high dimensional data independently.

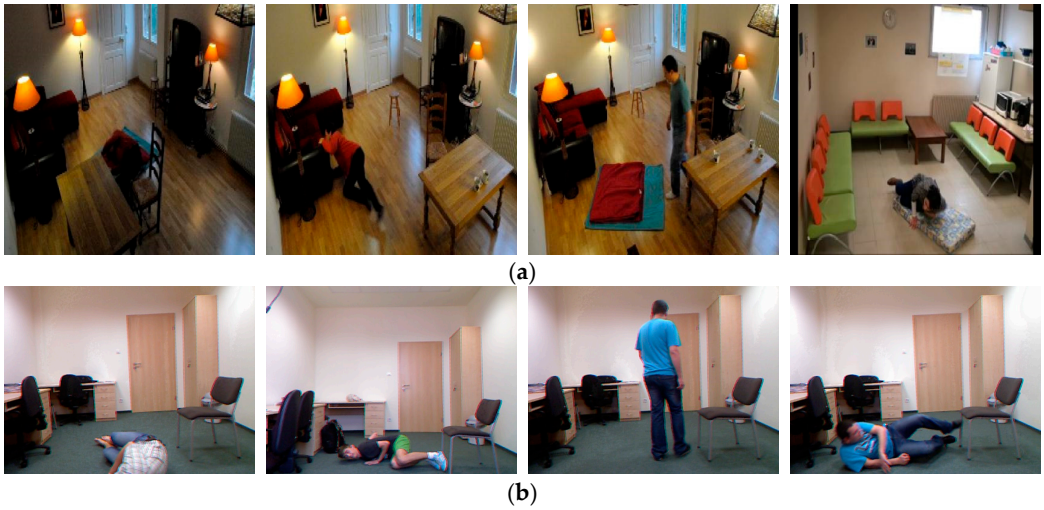
Deep learning has been successfully applied in fall detection, demonstrating high performance. Lu et al. [12] incorporate a combination of a three-dimensional convolutional neural network (3D CNN) and a long short-term memory (LSTM) in their fall prevention method. Han et al. [13] improve the fall detection speed using a two-stream approach with the lightweight MobileVGG. Recently, human skeleton joint coordinates extracted from RGB data have also been used to detect falls [10,28]. The authors [10] propose a spatiotemporal network based on CNNs, GRUs, and fully connected layers to classify falls and ADL. The existing methods for fall detection involve either a combination of various tasks, such as foreground and background separation or human skeleton joint coordinates. In contrast, our proposed method uses spatio-temporal features from RGB data using a deep-learning-based tracking-by-detection approach.

A considerable amount of research has been conducted in the field of tracking-by-detection approach [41–44]. These methods use spatiotemporal data to extract relevant features and then use them as inputs to distinguish between the identified objects. Finally, a tracker follows the object during the video flow. Despite the promising results obtained by tracking-by-detection methods, these methods can still suffer from colour bias or complex underexposed conditions [45]. Therefore, this paper presents a two-stage fall detection framework. First, we perform exposure correction of the video frames using a dual illumination estimation method [46]. The method adaptively corrects the frames based on exposure conditions, including overexposed, underexposed, and partially over- and underexposed conditions. Second, we use the YOLOv7 algorithm [26] to detect the subject's activity and combine the appearance information from object-detection CNNs with a fast, improved Deep SORT [25] based multiple object tracking (MOT) method to extract motion features and compute trajectories.

3. Methodology

3.1. Experimental Data

In this paper, we use two publicly available datasets, i.e., the Le2i Fall Detection dataset [47] and the UR Fall Detection dataset (URFD) [48]. The Le2i dataset is composed of falls captured by narrow angle cameras. The dataset includes videos of various actors falling and not falling in different illumination scenarios. The videos vividly depict real-world scenes of people falling while performing their daily activities. These scenes are captured in a variety of environments, including homes, workplaces, and pantries; it makes for more accurately simulated realistic video sequences of falls during everyday activities, as shown in Figure 1. The UR Fall Detection dataset [48] contains 60 videos taken by two cameras placed at different angles. Figure 1a,b shows some example video frames from the Le2i and URFD dataset, respectively.



(**a**)

(**b**)

Figure 1. Some example video frames from the publicly available datasets (**a**) Le2i and (**b**) URFD dataset.

3.2. Annotation

Our annotate method detects the person class in each frame using a pre-trained YOLOv5 model, and then stores the bounding box coordinates in a CSV table. In the next step, we examine each frame and correct the annotation results for fall and upright, as shown in Table 1. Figure 2 shows the detection results of YOLOv5 and Manual post-correction.

Table 1. Example bounding box coordinates and annotations.

File Name	Xmin	Ymin	Xmax	Ymax	YOLOv5 Annotations	Manually Corrected Annotations
000018.jpg	120	101	187	200	person	upright
000019.jpg	109	113	194	198	person	upright
000020.jpg	108	116	195	197	person	fall
000021.jpg	107	125	198	200	person	fall
000022.jpg	107	125	198	200	person	fall
000023.jpg	107	125	198	200	person	fall

3.3. Proposed Framework

The proposed framework is shown in Figure 3, which employs a two-stage framework to detect falls in low-light indoor environments. The object detection model is used as the main component of the fall detection system to further elaborate on the proposed technique. In the first stage, the captured video frames are pre-processed to improve the visual quality of the footage. This is accomplished using a dual illumination estimation-based exposure correction technique field [40]. The goal of this step is to adjust the brightness, contrast, and detail of the video frames so that later stages can detect falls more accurately. On the enhanced video frames, the second stage implements a deep-learning-based approach for person tracking and detection. This is accomplished using a technique known as people tracking by detection, which involves using an object detection model to track the movements of people in video footage. The algorithm can detect when a person falls by analysing the motion patterns of the tracked objects. To detect falls in low-light indoor environments, the proposed fall detection system utilizes a sophisticated combination of exposure correction and deep-learning-based people tracking by detection algorithms. By improving the visual quality of the video footage and leveraging advanced object detection techniques, the system can improve the accuracy and reliability of fall detection.

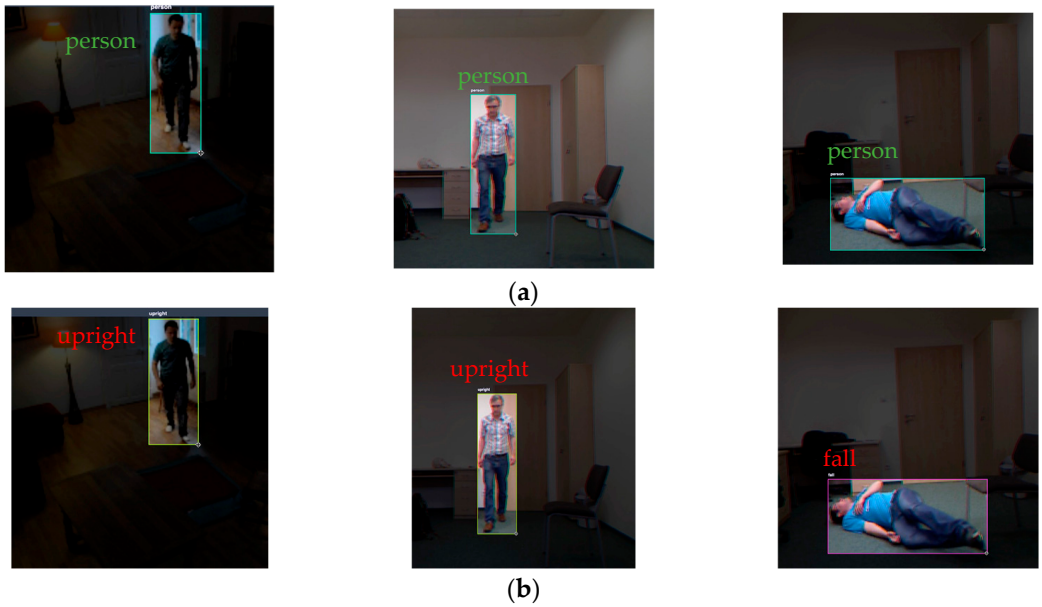


Figure 2. The detection results of YOLOv5 and Manual post-correction. (**a**) YOLOv5 pre-trained model. (**b**) Manual post-correction.

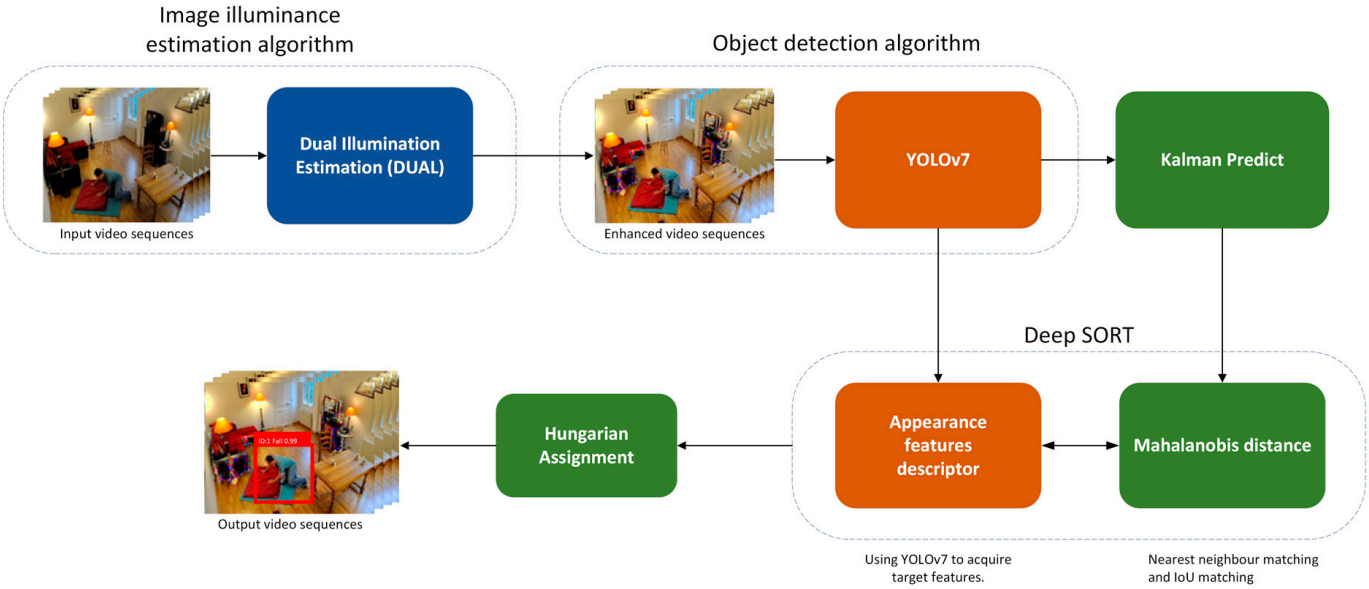


Figure 3. Schematic representation of the proposed framework.

3.3.1. Object Detection

RetinaNet was proposed by Lin et al. [49] in 2018, which is based on three parts: the ResNet backbone, the feature pyramid network (FPN), and the object classification and bounding box regression subnetwork. Deeper neural networks capture more information, and the ResNet backbone uses identity mappings to ensure that performance does not degrade as the network's depth increases. The FPN brings the benefits of multi-scale fusion by capturing coarse to fine semantic information, allowing for more feature details to be obtained for small object detection tasks and thus improving accuracy. The RetinaNet model runs the classification and regression sub-networks in parallel before attaching them to the FPN module. A sigmoid activation function is used to predict the classification at each location by the classification module. The regression sub-network outputs coordinate values of shape 4 via a fully connected layer. In object detection tasks [50–52], an imbalance between positive and negative samples is a major cause of classification difficulties. The

Focal Loss method solves this problem by adjusting the weights of the difficult and easy-to-classify samples.

Redmon et al. [53] proposed You Only Look Once (YOLO), which combines object detection and regression into a single task. The input image is segmented into multiple grid cells to accomplish this. The case where the object's centre point is located on the grid is seen as the anchor box responsible for detecting the object class. The YOLOv5 [54] and YOLOv7 [32] models used in this paper are optimized from YOLOv3 [55]. The reparameterization convolution method is used in YOLOv7. This method fuses the convolutional layers and batch normalization into a single convolutional module, which greatly improving the model inference speed.

The main advantage of one-stage models used in the proposed method is the high computational efficiency and better inference speeds over two-stage models. In real-time, vision-based systems for fall detection need to be faster and more efficient. Furthermore, the one-stage models address the class imbalance problem in fall detection to improve the detection accuracy.

3.3.2. Object Tracking

The Simple Online and Realtime Tracking (SORT) algorithm [30], proposed in 2016, is a straightforward and quick method for multiple object tracking (MOT). The correlation between the preceding and following frames is processed using Kalman filtering and then measured using the Hungarian algorithm [56]. The Deep SORT algorithm is an extension of the SORT algorithm. The Deep SORT algorithm's neural network weights training on the pedestrian dataset as best suited for human fall detection. By matching the extracted features to the object's nearest neighbors, it is well adapted for improving object tracking [57] and detection in obstacle-type environments [31]. The feature extraction network can significantly improve the Deep SORT algorithm's robustness to obstacles and object loss.

Using object-tracking methods in conjunction with fall detection systems is a promising approach, especially given the limitations of current visual method studies, which frequently overlook the possibility of increased false-negative detection rates. Deep SORT, fortunately, can help address this issue by predicting the potential location of the next frame and calculating correlations, providing an early warning if a person becomes obscured from view. It is possible to improve the accuracy and reliability of fall detection systems by incorporating this approach, thereby improving the safety and well-being of individuals at risk of falling. It is notable to combine object tracking methods with fall detection systems. Most current visual method studies do not account for the potential increase in false-negative detection rates. Deep SORT provides early warning if a person is obscured by predicting the potential location of the next frame and calculating correlations.

3.3.3. Dual Illumination Estimation (DUAL)

Zhang et al. [40] proposed the dual illumination estimation algorithm for dark light image enhancement. This method is based on the core concept of Retinex. The colour of an object is determined by its ability to reflect light of three different wavelengths. The support for this concept is based on colour constancy. Dual illumination enhancement demonstrates that uneven lighting conditions do not affect the colour of the object. When there is underexposure or underexposure, the colour of the object does not change. The illumination is estimated in both the forward and reverse directions to recover a properly exposed image.

The objective of this project is to create the image I by using the image light mapping method L to multiply each pixel value in the existing image I'. When an overexposed image is reversed, it produces an underexposed image. This can be interpreted as locating the

overexposed portion of the current image by entering a forward and reverse image. The generated image I is defined in Equation (1) below,

$$I = I' \times L \tag{1}$$

Here, I' is calculated according to Equation (2), where $I_{inv} = 1 - I$ is the formula for the inverted image. At this point, the illuminance map L_{inv} for the current state is estimated. As a result of calculating the underexposure correction, the image I_{inv} can be obtained.

$$I' = 1 - \left(I_{inv} \times I_{inv}^{-1}. \right) \tag{2}$$

$$L'_p = max I^c_p, \ \forall c \ \in \{r, g, b\} \tag{3}$$

where c is the colour channel; p is the pixel and I^c_p is expressed as the colour channel c in pixel point p. Before estimating the illumination of the image, it needs to extract the maximum channel value for each pixel in the RGB tree-channel image. It is possible to compose an initialized illuminance L'_p by iteratively obtaining each maximum value as shown in Equation in (3). The desired illumination map L is obtained using the objective defined in Equation (4).

$$\underset{L}{\operatorname{argmin}} \sum_p \left(\left(L_p - L'_p \right)^2 + \lambda \left(w_{x,y} (\vartheta_x L)^2_p + w_{y,p} (\vartheta_y L)^2_p \right) \right) \tag{4}$$

In real-world conditions, the performance of deep learning models suffers due to videos captured under suboptimal lighting conditions caused by dim or uneven light. For example, in an indoor setting, a video captured at night has dark or under-exposed regions due to insufficient illumination. This may result in the proposed systems failing to monitor individuals' activities, resulting in high false-negative rates. Figure 4 shows the results from Le2i FDD obtained by using a dual estimation algorithm.

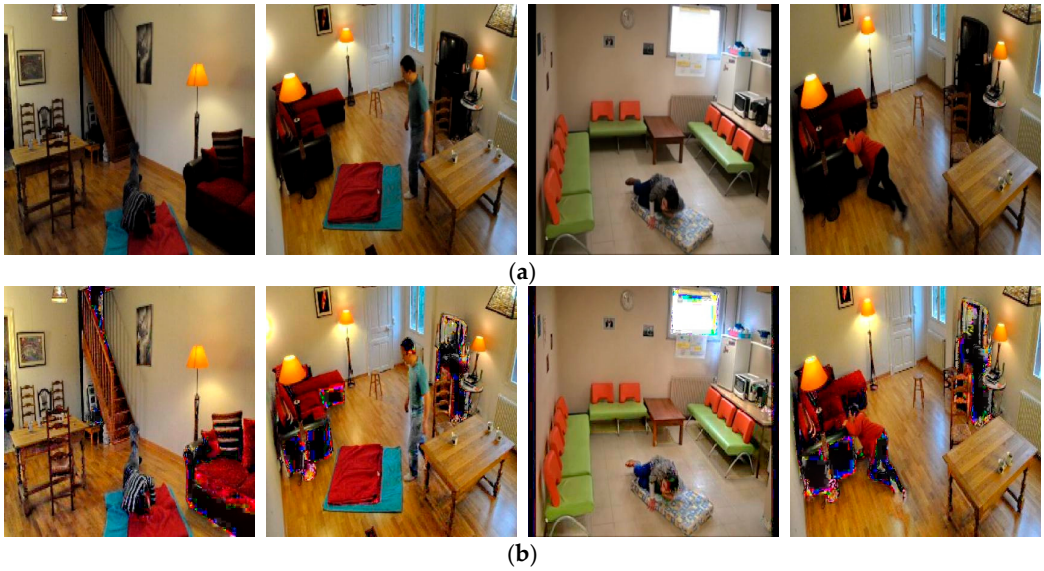


Figure 4. Examples of images from Le2i Fall Detection dataset. (**a**) The original images. (**b**) The images after processing by the DUAL illumination estimation.

4. Experimental Evaluation

4.1. Experimental Setup

This paper uses RetinaNet, YOLOv5, and YOLOv7 object detection algorithms based on a PyTorch framework to evaluate the proposed approach. Experimental results are conducted on a Tesla P100 GPU with 16,280 Mb video memory. The initial learning rate

for YOLO series models is set to 0.1, gradually decreasing to 0.01 during training, whereas RetinaNet uses a learning rate of 0.00025. As shown in Table 2, the batch size for YOLOv7 and RetinaNet is set to 8, and for YOLOv5, the batch size is set to 32. All the models are trained for 100 epochs each. For the YOLOv7/w DUAL, we use the same model parameters as YOLOv7. McNemar's significance test is used to statistically validate the performance of the object detection models. Next, the Deep SORT tracking algorithm is applied to the best performing object detection model. Specifically, we use the performance metrics such as accuracy, 0.5 mAP, and precision score for evaluation.

Table 2. Details of the parameters of the model trained.

Parameters	The Proposed Method	YOLOv7	YOLOv5	RetinaNet
Learning Rate	0.01–0.1	0.01–0.1	0.001–0.01	0.00025
Batch Size	8	8	32	8
Epochs	100	100	100	100
Data Enhancement	Yes	-	-	-

4.2. Results

Table 3 shows the results of McNemar's significance test (p values indicate the results obtained by the best performing model) on the Le2i Fall Detection dataset where YOLOv7 is statistically different than the results by RetinaNet and YOLOv5. Table 4 shows the results from different object detection algorithms on the Le2i Fall Detection dataset (Le2i FDD). YOLOv7 outperforms YOLOv5 and the RetinaNet with an accuracy of 90.5%, and mAP of 0.966. Here, the RetinaNet model performs the worst with an accuracy of 59.02%, and mAP of 0.842. Then, we apply Deep SORT to the best performing object detection model, i.e., YOLOv7, which is trained and tested on enhanced video frames obtained by performing dual illumination estimation. Here, the proposed method (YOLOv7 + Deep SORT/w DUAL) gives an accuracy of 94.5% and mAP of 0.986, which is a significant improvement over the object detection methods. Moreover, YOLOv7 + Deep SORT/w DUAL is compared to the existing state-of-the-art (SOTA) methods on Le2i FDD. Table 4 shows that the proposed approach achieves the highest accuracy, outperforming the SOTA methods by Poonsri et al. [58] and Chamle et al. [59], which have the fall accuracy of 91.38% and 79.41%, respectively. Poonsri et al. [58] and Chamle et al. [59] have used background-based subtraction techniques. However, due to insufficient illumination conditions in videos of Le2i FDD, background subtraction techniques used in these methods incorrectly extract other objects as a human silhouette. This results in high false-positive rates for fall detection. The proposed method is compared to the traditional method proposed by Poonsri et al. [48] and Chamle et al. [49] based on their annotated images and the result presented. Furthermore, the visual results using the YOLOv7 and the proposed method are shown in Figure 5. When the environment light is not insufficiently illuminated, as shown in the first row of Figure 6, the YOLOv7 + Deep SORT misclassifies the fall as upright. However, the proposed method correctly classifies it as a fall, as shown in the second row of Figure 5.

Table 3. p-Values of the McNemar's significance test on Le2i dataset. Here, $p < 0.05$ is statistically significant.

Method	RetinaNet	YOLOv5	YOLOv7
YOLOv7	<0.001	0.0095	1

Table 4. Performance of the different state-of-the-art methods on Le2i dataset.

Method	Accuracy (%)	0.5 mAP	Precision of Fall
Poonsri et al. [58]	91.38	-	0.886
Chamle et al. [59]	79.31	-	0.794
RetinaNet [49]	59.02	0.842	0.775
YOLOv5 [54]	86.0	0.947	0.896
YOLOv7 [32]	90.5	0.966	0.935
The proposed method	94.5	0.986	0.970

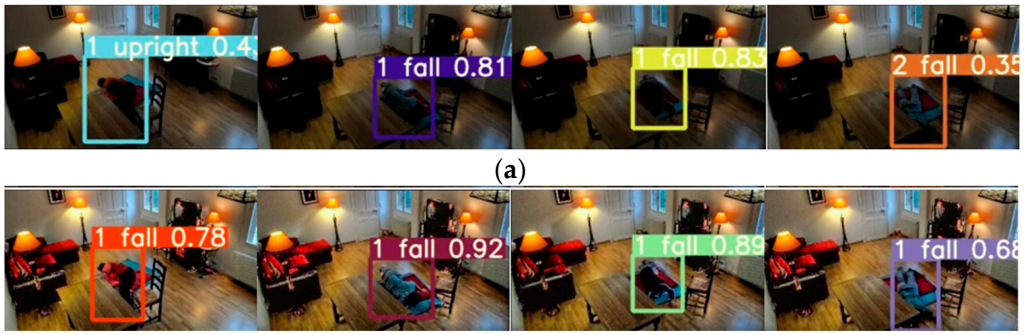


Figure 5. Visual results of object tracking on Le2i Fall Detection dataset. (**a**) YOLOv7 + Deep SORT method. (**b**) The proposed method.

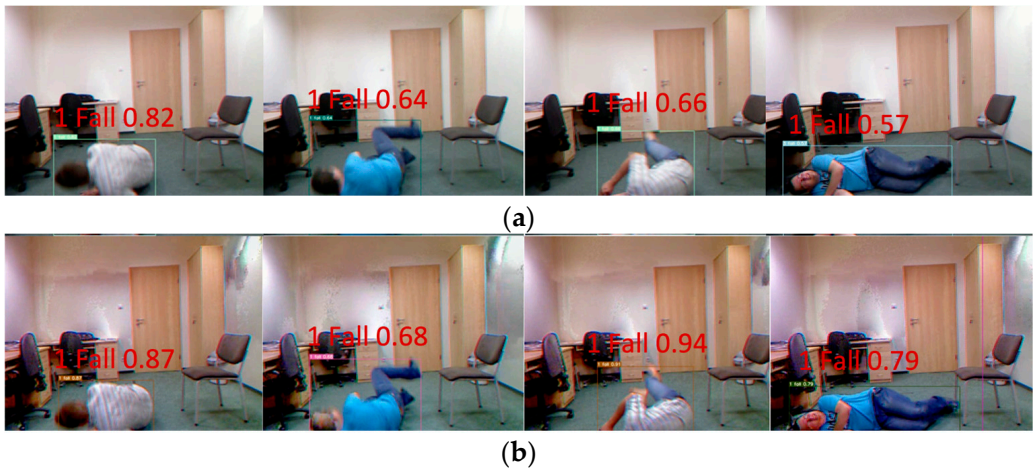


Figure 6. Visual results of object tracking on UR Fall Detection dataset (**a**) YOLOv7 + Deep SORT method (**b**) the proposed method.

Table 5 shows the results of McNemar's significance test on UR-Fall dataset where the performance of YOLOv7 is statistically significant ($p < 0.05$) over RetinaNet, but it is not statistically different from that of YOLOv5. Table 6 shows the different object detection algorithms on the UR Fall detection dataset. YOLO models achieve high results on metrics such as accuracy and mAP, with YOLOv7 performing the best among all the models. However, RetinaNet did not perform well on the UR-Fall dataset, reporting the lowest accuracy of 40.9% as well as the lowest mAP of 0.464. Similarly, to the experiments on Le2i FDD, YOLOv7 is integrated with the Deep SORT tracking algorithm to report the results on the UR-Fall dataset. The proposed method outperforms all the methods in terms of accuracy, mAP, and precision. As shown in the second row of Figure 6, the visual results on the UR-Fall dataset using the proposed method highlights the improved performance on the enhanced video frames as compared to the YOLOv7 + Deep SORT.

Table 5. *p*-Values of the McNemar's significance test on UR-Fall dataset.

Method	RetinaNet	YOLOv5	YOLOv7
YOLOv7	<0.001	0.085	1

Table 6. Testing Performance of different models in UR Fall detection dataset.

Method	Accuracy (%)	0.5 mAP	Precision of Fall
RetinaNet [49]	40.9	0.464	0.818
YOLOv5 [54]	89.8	0.925	0.881
YOLOv7 [32]	92.4	0.944	0.893
The proposed method	93.2	0.960	0.920

5. Conclusions

This paper proposes a vision-based fall detection system with an improved deep-learning-based tracking-by-detection method. The proposed method integrates dual illumination estimation to the YOLOv7 + Deep SORT tracking algorithm to enhance fall detection performance under suboptimal lighting conditions caused by dim or uneven light. The proposed method also incorporates exposure correction for fall detection in videos. The performance of the proposed method is validated on two fall detection datasets, namely, Le2i FDD and UR-Fall datasets. For future experiments, we aim to implement a self-learning framework that automatically adapts to false alarms and adds correct results to help the current fall detection systems to improve their performance.

Author Contributions: Conceptualization, X.Z. and M.P.; methodology, X.Z. and K.C.; software, X.Z. and K.C.; validation, X.Z., K.C., A.B., J.L. and M.P.; formal analysis, X.Z., K.C. and A.B.; investigation, X.Z. and K.C.; resources, A.B., J.L. and M.P.; data curation, X.Z. and K.C.; writing—original draft preparation, X.Z.; writing—review and editing, X.Z., K.C., A.B., J.L. and M.P.; visualization, K.C., A.B. and J.L.; supervision, J.L. and M.P.; project administration, A.B. and M.P.; funding acquisition, J.L. and M.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. World Health Organization (WHO). Falls. 2021. Available online: https://www.who.int/news-room/fact-sheets/detail/falls (accessed on 17 July 2022).
2. Alam, E.; Sufian, A.; Dutta, P.; Leo, M. Vision-based human fall detection systems using deep learning: A review. *Comput. Biol. Med.* **2022**, *146*, 105626. [CrossRef] [PubMed]
3. Burns, E.; Kakara, R. Deaths from Falls Among Persons Aged ≥ 65 Years—United States, 2007–2016. *MMWR. Morb. Mortal. Wkly. Rep.* **2018**, *67*, 509–514. [CrossRef]
4. Kelsey, J.L.; Procter-Gray, E.; Hannan, M.T.; Li, W. Heterogeneity of Falls Among Older Adults: Implications for Public Health Prevention. *Am. J. Public Health* **2012**, *102*, 2149–2156. [CrossRef]
5. Vishnu, C.; Datla, R.; Roy, D.; Babu, S.; Mohan, C.K. Human Fall Detection in Surveillance Videos Using Fall Motion Vector Modeling. *IEEE Sensors J.* **2021**, *21*, 17162–17170. [CrossRef]
6. Mubashir, M.; Shao, L.; Seed, L. A survey on fall detection: Principles and approaches. *Neurocomputing* **2013**, *100*, 144–152. [CrossRef]
7. Saleh, M.; Jeannes, R.L.B. Elderly Fall Detection Using Wearable Sensors: A Low Cost Highly Accurate Algorithm. *IEEE Sensors J.* **2019**, *19*, 3156–3164. [CrossRef]
8. Wang, X.; Ellul, J.; Azzopardi, G. Elderly Fall Detection Systems: A Literature Survey. *Front. Robot. AI* **2020**, *7*. [CrossRef]
9. Albawendi, S.; Lotfi, A.; Powell, H.; Appiah, K. Video Based Fall Detection using Features of Motion, Shape and Histogram. In Proceedings of the 11th PErvasive Technologies Related to Assistive Environments Conference, Corfu, Greece, 26–29 June 2018; pp. 529–536. [CrossRef]
10. Yadav, S.K.; Luthra, A.; Tiwari, K.; Pandey, H.M.; Akbar, S.A. ARFDNet: An efficient activity recognition & fall detection system using latent feature pooling. *Knowl. Based Syst.* **2022**, *239*, 107948. [CrossRef]

11. Sehairi, K.; Chouireb, F.; Meunier, J. Elderly fall detection system based on multiple shape features and motion analysis. In Proceedings of the 2018 IEEE International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, 2–4 April 2018; pp. 1–8. [CrossRef]
12. Lu, N.; Wu, Y.; Feng, L.; Song, J. Deep Learning for Fall Detection: Three-Dimensional CNN Combined with LSTM on Video Kinematic Data. *IEEE J. Biomed. Health Inform.* **2018**, *23*, 314–323. [CrossRef]
13. Han, Q.; Zhao, H.; Min, W.; Cui, H.; Zhou, X.; Zuo, K.; Liu, R. A Two-Stream Approach to Fall Detection with MobileVGG. *IEEE Access* **2020**, *8*, 17556–17566. [CrossRef]
14. Khraief, C.; Benzarti, F.; Amiri, H. Elderly fall detection based on multi-stream deep convolutional networks. *Multimed. Tools Appl.* **2020**, *79*, 19537–19560. [CrossRef]
15. Li, J.; Han, L.; Zhang, C.; Li, Q.; Liu, Z. Spherical Convolution Empowered Viewport Prediction in 360 Video Multicast with Limited FoV Feedback. *ACM Trans. Multimed. Comput. Commun. Appl.* **2023**, *19*, 1–23. [CrossRef]
16. Feng, Q.; Feng, Z.; Su, X. Design and Simulation of Human Resource Allocation Model Based on Double-Cycle Neural Network. *Comput. Intell. Neurosci.* **2021**, *2021*, 7149631. [CrossRef] [PubMed]
17. Liu, H.; Liu, M.; Li, D.; Zheng, W.; Yin, L.; Wang, R. Recent Advances in Pulse-Coupled Neural Networks with Applications in Image Processing. *Electronics* **2022**, *11*, 3264. [CrossRef]
18. Qin, X.; Liu, Z.; Liu, Y.; Liu, S.; Yang, B.; Yin, L.; Liu, M.; Zheng, W. User OCEAN Personality Model Construction Method Using a BP Neural Network. *Electronics* **2022**, *11*, 3022. [CrossRef]
19. Lu, H.; Zhu, Y.; Yin, M.; Yin, G.; Xie, L. Multimodal Fusion Convolutional Neural Network with Cross-Attention Mechanism for Internal Defect Detection of Magnetic Tile. *IEEE Access* **2022**, *10*, 60876–60886. [CrossRef]
20. Zhou, W.; Wang, H.; Wan, Z. Ore Image Classification Based on Improved CNN. *Comput. Electr. Eng.* **2022**, *99*, 107819. [CrossRef]
21. Huang, C.-Q.; Jiang, F.; Huang, Q.-H.; Wang, X.-Z.; Han, Z.-M.; Huang, W.-Y. Dual-Graph Attention Convolution Network for 3-D Point Cloud Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–13. [CrossRef]
22. Xu, S.; He, Q.; Tao, S.; Chen, H.; Chai, Y.; Zheng, W. Pig Face Recognition Based on Trapezoid Normalized Pixel Difference Feature and Trimmed Mean Attention Mechanism. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 3500713. [CrossRef]
23. Zhou, X.; Zhang, L. SA-FPN: An effective feature pyramid network for crowded human detection. *Appl. Intell.* **2022**, *52*, 12556–12568. [CrossRef]
24. Shi, Y.; Xu, X.; Xi, J.; Hu, X.; Hu, D.; Xu, K. Learning to Detect 3D Symmetry from Single-View RGB-D Images With Weak Supervision. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, 1–15. [CrossRef] [PubMed]
25. Yang, D.; Zhu, T.; Wang, S.; Wang, S.; Xiong, Z. LFRSNet: A robust light field semantic segmentation network combining contextual and geometric features. *Front. Environ. Sci.* **2022**, *10*, 1443. [CrossRef]
26. Sheng, H.; Cong, R.; Yang, D.; Chen, R.; Wang, S.; Cui, Z. UrbanLF: A Comprehensive Light Field Dataset for Semantic Segmentation of Urban Scenes. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 7880–7893. [CrossRef]
27. Galvão, Y.M.; Ferreira, J.; Albuquerque, V.A.; Barros, P.; Fernandes, B.J. A multimodal approach using deep learning for fall detection. *Expert Syst. Appl.* **2020**, *168*, 114226. [CrossRef]
28. Ramirez, H.; Velastin, S.A.; Meza, I.; Fabregas, E.; Makris, D.; Farias, G. Fall Detection and Activity Recognition Using Human Skeleton Features. *IEEE Access* **2021**, *9*, 33532–33542. [CrossRef]
29. Cheng, S.; Liu, J.; Li, Z.; Zhang, P.; Chen, J.; Yang, H. 3D error calibration of spatial spots based on dual position-sensitive detectors. *Appl. Opt.* **2023**, *62*, 933–943. [CrossRef] [PubMed]
30. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468. [CrossRef]
31. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3645–3649. [CrossRef]
32. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. 2022. Available online: http://arxiv.org/abs/2207.02696 (accessed on 14 December 2022).
33. Abbate, S.; Avvenuti, M.; Bonatesta, F.; Cola, G.; Corsini, P.; Vecchio, A. A smartphone-based fall detection system. *Pervasive Mob. Comput.* **2012**, *8*, 883–899. [CrossRef]
34. Palmerini, L.; Klenk, J.; Becker, C.; Chiari, L. Accelerometer-Based Fall Detection Using Machine Learning: Training and Testing on Real-World Falls. *Sensors* **2020**, *20*, 6479. [CrossRef] [PubMed]
35. Bagalà, F.; Becker, C.; Cappello, A.; Chiari, L.; Aminian, K.; Hausdorff, J.M.; Zijlstra, W.; Klenk, J. Evaluation of Accelerometer-Based Fall Detection Algorithms on Real-World Falls. *PLoS ONE* **2012**, *7*, e37062. [CrossRef] [PubMed]
36. Irtaza, A.; Adnan, S.M.; Aziz, S.; Javed, A.; Ullah, M.O.; Mahmood, M.T. A framework for fall detection of elderly people by analyzing environmental sounds through acoustic local ternary patterns. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 1558–1563. [CrossRef]
37. Fan, X.; Zhang, H.; Leung, C.; Shen, Z. Robust unobtrusive fall detection using infrared array sensors. In Proceedings of the 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Daegu, Republic of Korea, 16–18 November 2017; pp. 194–199. [CrossRef]
38. Muheidat, F.; Tawalbeh, L.A.; Tyrer, H. Context-Aware, Accurate, and Real Time Fall Detection System for Elderly People. In Proceedings of the 2018 IEEE 12th International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA, 31 January–2 February 2018; pp. 329–333. [CrossRef]

39. Chaccour, K.; Darazi, R.; el Hassans, A.H.; Andres, E. Smart carpet using differential piezoresistive pressure sensors for elderly fall detection. In Proceedings of the 2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Abu Dhabi, United Arab Emirates, 19–21 October 2015; pp. 225–229. [CrossRef]
40. Ren, L.; Peng, Y. Research of Fall Detection and Fall Prevention Technologies: A Systematic Review. *IEEE Access* **2019**, *7*, 77702–77722. [CrossRef]
41. Wang, S.; Sheng, H.; Yang, D.; Zhang, Y.; Wu, Y.; Wang, S. Extendable Multiple Nodes Recurrent Tracking Framework With RTU++. *IEEE Trans. Image Process.* **2022**, *31*, 5257–5271. [CrossRef] [PubMed]
42. Wu, Y.; Sheng, H.; Zhang, Y.; Wang, S.; Xiong, Z.; Ke, W. Hybrid Motion Model for Multiple Object Tracking in Mobile Devices. *IEEE Internet Things J.* **2022**, 1. [CrossRef]
43. Xiong, S.; Li, B.; Zhu, S. DCGNN: A single-stage 3D object detection network based on density clustering and graph neural network. *Complex Intell. Syst.* **2022**, 1–10. [CrossRef]
44. Lu, S.; Liu, S.; Hou, P.; Yang, B.; Liu, M.; Yin, L.; Zheng, W. Soft Tissue Feature Tracking Based on Deep Matching Network. *Comput. Model. Eng. Sci.* **2023**, *136*, 363–379. [CrossRef]
45. Zhao, L.; Lu, S.-P.; Chen, T.; Yang, Z.; Shamir, A. Deep Symmetric Network for Underexposed Image Enhancement with Recurrent Attentional Learning. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 12055–12064. [CrossRef]
46. Zhang, Q.; Nie, Y.; Zheng, W. Dual Illumination Estimation for Robust Exposure Correction. *Comput. Graph. Forum* **2019**, *38*, 243–252. Available online: http://arxiv.org/abs/1910.13688 (accessed on 21 July 2022). [CrossRef]
47. Charfi, I.; Miteran, J.; Dubois, J.; Atri, M.; Tourki, R. Optimized spatio-temporal descriptors for real-time fall detection: Comparison of support vector machine and Adaboost-based classification. *J. Electron. Imaging* **2013**, *22*, 41106. [CrossRef]
48. Kwolek, B.; Kepski, M. Human fall detection on embedded platform using depth maps and wireless accelerometer. *Comput. Methods Programs Biomed.* **2014**, *117*, 489–501. [CrossRef]
49. Lin, T.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2999–3007. [CrossRef]
50. Cheng, E.J.; Prasad, M.; Yang, J.; Khanna, P.; Chen, B.-H.; Tao, X.; Young, K.-Y.; Lin, C.-T. A fast fused part-based model with new deep feature for pedestrian detection and security monitoring. *Measurement* **2019**, *151*, 107081. [CrossRef]
51. Hong, G.-J.; Li, D.-L.; Pare, S.; Saxena, A.; Prasad, M.; Lin, C.-T. Adaptive Decision Support System for On-Line Multi-Class Learning and Object Detection. *Appl. Sci.* **2021**, *11*, 11268. [CrossRef]
52. Cheng, E.J.; Prasad, M.; Yang, J.; Zheng, D.R.; Tao, X.; Mery, D.; Young, K.Y.; Lin, C.T. A novel online self-learning system with automatic object detection model for multimedia applications. *Multimed. Tools Appl.* **2020**, *80*, 16659–16681. [CrossRef]
53. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
54. Jocher, G.; Stoken, A.; Chaurasia, A.; Borovec, J.; Kwon, Y.; Michael, K.; Liu, C.; Fang, J.; Abhiram, V.; Skalski, S.P. Ultralytics/yolov5: V6.0—YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support. *Zenodo Tech. Rep.* **2021**. [CrossRef]
55. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767. Available online: http://arxiv.org/abs/1804.02767 (accessed on 20 January 2023).
56. Huang, C.-J. Integrate the Hungarian Method and Genetic Algorithm to Solve the Shortest Distance Problem. In Proceedings of the 2012 Third International Conference on Digital Manufacturing & Automation, Guilin, China, 31 July–2 August 2012; pp. 496–499. [CrossRef]
57. Chang, L.C.; Pare, S.; Meena, M.S.; Jain, D.; Li, D.L.; Saxena, A.; Prasad, M.; Lin, C.T. An Intelligent Automatic Human Detection and Tracking System Based on Weighted Resampling Particle Filtering. *Big Data Cogn. Comput.* **2020**, *4*, 27. [CrossRef]
58. Poonsri, A.; Chiracharit, W. Fall detection using Gaussian mixture model and principle component analysis. In Proceedings of the 2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE), Phuket, Thailand, 12–13 October 2017; pp. 1–4. [CrossRef]
59. Chamle, M.; Gunale, K.G.; Warhade, K.K. Automated unusual event detection in video surveillance. In Proceedings of the 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 26–27 August 2016; pp. 1–4. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Deep Signal-Dependent Denoising Noise Algorithm

Lanfei Zhao [1], Shijun Li [1] and Jun Wang [2,*]

[1] The Higher Educational Key Laboratory for Measuring & Control Technology and Instrumentations of Heilongjiang Province, Harbin University of Science and Technology, Harbin 150080, China
[2] School of Information Engineering, Quzhou College of Technology, Quzhou 324000, China
* Correspondence: 145688@qzct.edu.cn; Tel.: +86-181-0570-1939

Abstract: Although many existing noise parameter estimations of image signal-dependent noise have certain denoising effects, most methods are not ideal. There are some problems with these methods, such as poor noise suppression effects, smooth details, lack of flexible denoising ability, etc. To solve these problems, in this study, we propose a deep signal-dependent denoising noise algorithm. The algorithm combines the model method with a convolutional neural network. We use the noise level of the noise image and the noise image together as the input of the convolutional neural network to obtain a wider range of noise levels than the single noise image as the input. In the convolutional neural network, the deep features of the image are extracted by multi-layer residuals, which solves the difficult problem of training. Extensive experiments demonstrate that our noise parameter estimation has good denoising performance.

Keywords: signal-dependent noise; noise parameter estimation; convolutional neural network; image denoising

1. Introduction

Image denoising is a classic and indispensable research topic in low-level vision tasks, which is the premise of high-level vision tasks [1]. The clear and high-quality images obtained by denoising serve the high-level vision and make completing the tasks better. This paper mainly introduces the denoising of signal-dependent noisy images.

With the continuous updating of digital image acquisition technology, equipment for acquiring images and the number of acquired images are also increasing, and people's requirements for image quality are also increasing. However, due to the imperfections of imaging systems, imaging equipment, and transmission media, many different types of noise are introduced during the formation of images in different devices, which can affect the imaging effect, and thus the quality of the image. Noise-contaminated images will have a considerable impact on subsequent image processing. Noise pollution will affect the quality of subsequent image processing, such as image recognition, segmentation, classification, and so on [2].

In this paper, we model the signal-independent noise in the image, and then input it into the denoising network to solve the problem. During training and testing, networks that take the noise level of the input image and the noise image as input have a wider range of noise levels than networks with a single noise image as the input. The success of CNN denoisers is significantly dependent on whether the distributions of synthetic and real noises are well matched. Therefore, realistic noise models are the foremost issue for blind denoising of real photographs. According to [3], signal-dependent noise can be modeled by signal-dependent Gaussian noise distribution, and the signal-dependent Gaussian model has been considered as a more appropriate alternative than other models for real raw noise modeling. In addition, it is necessary to further optimize the graph after noise modeling. Because there are two unavoidable errors in modeling, the first one is the error generated in the estimation of noise parameters. The second is when pixel values are obtained by the

local statistical feature method, which also has errors. Thus, after acquiring the preliminary picture, we use the preliminary image and the noise image together as the input into the denoising network. The two errors are reduced through the network.

At present, image denoising relies on the image degradation model, but most of the image degradation model algorithms are simple and do not match the complex noise in the real image. It leads to the less ideal denoising effect. Therefore, it is necessary to design a more reasonable model or an algorithm with better performance. This paper proposes a method combining the model method and the convolutional neural network. The preliminary image obtained by the model method is used as a part of the network's input, and the final image is output through the continuous iteration of the network.

Most image denoising is Gaussian noise denoising. The variance of the noise is constant and does not change with the position and intensity of the pixel, the noise level is the only parameter required for modeling. The model of this degradation process is generally defined as:

$$y = x + n \tag{1}$$

where x is the degraded image pixel value, y is the original image pixel value, n is Additive White Gaussian Noise (AWGN) with a standard deviation of σ. Since Gaussian white noise has only one unknown parameter, the variance of the noise, DnCNN enumerates the variance of different values in the process of training the network model [4], meaning that the network can remove AWGN in the case of unknown noise variance. However, because there are other noises in the real image, the Gaussian modeling of the image cannot achieve a good denoising effect.

Alessandro Foi [5] modeled the real noise as a signal-dependent Poisson–Gaussian noise model and proposed an estimation method for Poisson–Gaussian noise parameters. However, he trimmed the underexposure and overexposure data when experimenting. Despite achieving good performance, the method is limited to Poisson–Gaussian noise due to modeling problems. The model needs to know the Gaussian and Poisson components a and b, the model is as follows:

$$\mathrm{z}(v) = k(v) + \sigma(k(v))\zeta(v) \tag{2}$$

$$\sigma^2(y(v)) = ay(v) + b \tag{3}$$

where v is the pixel position in the domain. $z(v)$ is the observed signal, and $k(v)$ is the original signal. $\zeta(v)$ is zero-mean independent random noise with a standard deviation equal to 1. σ is a function that gives the standard deviation of the overall noise component. The variance is an expression for the Poisson Gaussian noise component from which estimates of a and b are obtained.

Liu et al. [6] proposed a generalized signal-dependent noise model.

$$g = f + f^{\gamma}u + w \tag{4}$$

$$\sigma(f) = \sqrt{f^{2\gamma}\sigma_u^2 + \sigma_w^2} \tag{5}$$

where g is the noisy pixel value, f is the noise-free pixel value, and u and w are zero-mean Gaussian variables. γ is an exponential parameter that controls signal dependence by changing the three NLF parameters γ, σ_u^2, and σ_w^2, and the noise model can represent various types of noise by changing these three parameters. Existing models usually assume that one parameter is known, and only estimate the remaining two parameters to simplify the problem. This method requires three parameters; thus, the amount of calculation is large.

The model we propose in this paper is a noise parameter estimation for real camera noise, which can be modeled with a signal-dependent Gaussian distribution [3].

$$x_p \sim N(y_p, \sigma_r^2 + \sigma_s y_p) \tag{6}$$

where x_p is the degraded image pixel value, y_p is the original image pixel value, and the noise parameters σ_s and σ_r are fixed and only change with the gain of the sensor. The main reasons for adopting this model are as follows: similar to the generalized noise model shown in [6], the exponential coefficient will make the calculation of the numerical solution difficult, and the number of unknown parameters in the Gaussian correlated noise model is less than that of the generalized signal-dependent noise model, thus the estimation difficulty will be much less. The model we use is more accurate in parameter estimation than the Poisson–Gaussian noise model, and the denoising effect is better.

In this paper, we propose a deep signal-dependent denoising noise algorithm. The algorithm further improves the noise suppression ability and improves the visual effect of the restored image. The denoising algorithm in this paper can effectively improve image quality, which plays an important role in many fields. First, we find a more reasonable model for the signal-related noise image, which has a wider range of application scenarios and a higher accuracy of parameter estimation. Second, we combine the model with the convolutional neural network to further improve the image-denoising effect. The algorithm further improves the noise suppression ability and improves the visual effect of the restored image. The denoising algorithm in this paper can effectively improve the image quality, which has a wide range of applications for the multimedia, military, medical, and other industries.

2. The Related Work

Ben et al. [3] proposed a technique for jointly denoising bursts of images taken from a handheld camera. Burst image noise parameter estimation operates on a set of successive, rapidly taken images to compute a single, noise-free result. In particular, they propose a convolutional neural network architecture for predicting spatially varying kernels that can both align and denoise frames: a synthetic data generation approach based on a realistic noise formation model. The method for burst denoising they proposed has the signal-to-noise ratio benefits of multi-image denoising and the large capacity and generality of a convolution neural network.

Liu et al. [7] proposed a segmentation-based image denoising algorithm for signal dependent noise. First, they identify the noise level function for a given single noisy image. Then, after initial denoising, segmentation is applied to the prefiltered image.

Tan et al. [8] proposed a deep convolutional neural network named "deep residual noise estimator" (DRNE) for pixel-wise noise-level estimation. The DRNE framework they designed consists of a stack of customized residual blocks without any pooling or interpolation operations. The noise level estimation graph obtained by DRNE, and the ground truth graph are used as the input of the convolutional neural network, which makes the original network achieve better results.

Talmaj et al. [9] improved KPN, and they proposed a deep neural network-based approach called Multi-Kernel Prediction Networks (MKPN) for burst image denoising. MKPN predicts kernels of not just one size but of varying sizes and performs a fusion of these different kernels resulting in one kernel per pixel. This enables MKPN to better extract information from images. MKPN has achieved good results in texture and homogeneous area denoising. Bingyang et al. [10] proposed a novel network including noise estimation module and removal module (NERNet). The noise estimation module automatically estimates the noise level map corresponding to the information extracted by symmetric dilated block and pyramid feature fusion block. To fuse noisy multi-modality image pairs accurately and efficiently, Huanqiu et al. [11] proposed a multi-modality image simultaneous fusion and denoising method.

Zhonghua et al. [12] proposed a novel model-guided boosting framework. By using the Regularization by Denoising (RED), they could apply explicit regularization equipped with powerful image denoising engine to establish the global minimization problem, making the obtained model clearly defined and well optimized. The framework enjoyed

the advantage of easily extending to the case of composite denoising via superadding a regularization term.

Gang Liu et al. [13] proposed a true wide CNN (WCNN) to reorganize several convolutional layers. Each subnetwork had its own input and output and was supervised by its own loss function to capture image features with a specific direction and scale, allowing the WCNN to have sufficient convolutional layers to capture image features while avoiding vanishing/exploding gradients. Jiechao Shen et al. [14] proposed the sparse representation-based network (SRNet). They considered combining the sparse representation with deep learning to make this traditional model more effective and efficient. They embedded the convolutional neural network (CNN) into the sparse representation framework. Laya et al. [15] proposed the Multi Scaling Aided Double Decker (MUS-ADD) convolutional neural network. It solved the disadvantages of the traditional method, which required a large number of models, etc. Shuang Xu et al. [16] built a more interpretable network. An observation model was proposed to account for modality gap between target and guidance images. Then, they formulated a deep prior regularized optimization problem, and solved it by the alternating direction method of multipliers (ADMM) algorithm. Yang Ou et al. [17] proposed a novel multi-scale weighted group sparse coding model (MS-WGSC). It better restored the structure and the edges of images contaminated by noise. Lei Zhang et al. [18] proposed the Robust Low-Rank Analysis with Adaptive Weighted Tensor (AWTD) method. It obtained the low-rank approximation of the tensor by adding adaptive weights to the unfolding matrix of the tensor. By decomposing true image into a cartoon part and texture part, Xiao Li et al. [19] proposed a fractional image denoising model with two-component regularization terms. Sunder Ali et al. [20] proposed a cascaded and recursive convolutional neural network (CRCNN) framework, which could cope with spatial variant noise and blur artifacts in a single denoising framework. Lei Zhang et al. [21] designed a novel denoising model named Kronecker Component with Low-Rank Dictionary (KCLD), which replaced the Frobenius norm with a Nuclear norm in order to capture the low-rank property better. Phan et al. [22] proposed an adaptive model that used the mean curvature of an image surface to control the strength of smoothing. A fast method for noise level estimation is proposed to improve the effectiveness of the proposed model.

In [23], Heng Yao et al. proposed an algorithm to efficiently estimate the noise level function (NLF), which is defined as the noise standard deviation with respect to image intensity. The method divided the input image into overlapping patches. The confidence levels of the noise samples and the prior of the camera response function were then employed as constraints for the recovery of the NLF. The NLF was incorporated into other denoising schemes to obtain better results than the original scheme.

Guo et al. [24] trained a convolutional blind denoising network (CBDNet) with a more realistic noise model and real-world noisy–clean image pairs. To further provide an interactive strategy to rectify denoising results conveniently, a noise estimation subnetwork with asymmetric learning to suppress under-estimation of noise level was embedded into CBDNet. They adopted an asymmetric loss by imposing more penalty on an under-estimation error of noise level, making CBDNet perform robustly when the noise model was not well matched with real-world noise.

Zhao et al. [25] proposed a simple but robust network called SDNet to improve the effectiveness and practicability of deep denoising models. Additionally, there is no need to estimate the noise level. The model learning had exploited clean–noisy image pairs, newly produced, built on a generalized signal-dependent noise model. To separate the noise from image content as fully as possible, a kind of lifting residual module was specifically proposed for discriminative feature extraction. The network emphasized separating the noise from image content via a direct, fully end-to-end residual learning strategy.

In [26], Dong et al. proposed a grouped residual dense network (GRDN). The core part of DRDN is defined as a grouped residual dense block (GRDB) and used as a building module of GRDN. They experimentally show that image-denoising performance can be

significantly improved by cascading GRDBs. Meanwhile, they developed a new generative adversarial network-based real-world noise modeling method.

3. Proposed Methods

Synthetic noise images are used in this paper. The most important noises in real images are Poisson noise (shot noise) and Gaussian noise. Some scholars have proposed a simple and practical noise model based on the original data of digital imaging sensors [3]. The signal-dependent noise model gives a function consisting of the Poisson part of the modeled photon sensing and the Gaussian part of the remaining stationary perturbations in the output data. Two parameters of a given noisy image model can be estimated by the image restoration algorithm based on local statistical features. Overall architecture of the algorithm is shown in Figure 1.

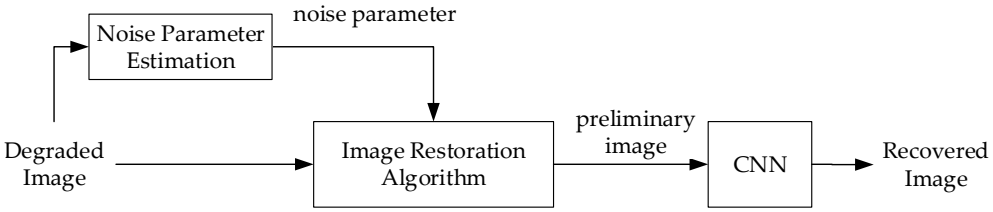


Figure 1. Overall architecture of the algorithm.

3.1. The Image Restoration Algorithm on Local Statistical Features

In this paper, the Gaussian correlation noise model shown in Equation (6) is used to model the image degradation process. The estimated values of the two parameters are obtained by estimating the parameters of the noise model, and then the preliminary image is obtained through local statistical features.

3.1.1. Noise Model Parameter Estimation

First, two important assumptions are given: first, according to [27], the statistical features of a pixel in a degraded image with signal-dependent noise can be derived from a local region centered at that pixel. Second, there are a certain number of locally homomorphic sub-blocks in the original image, and the variance of the pixel intensity in the sub-blocks is approximately 0. Assuming that the pixel blocks of y_p and its adjacent areas are homomorphic sub-blocks, the brightness of the original pixels in the homomorphic sub-blocks are y_p and $E(y_p) = y_p$. The equation represents the local mean of the homomorphic subblock image without noise as equal to the local mean of noisy image. According to the modeling formula of x_p, we can acquire:

$$D(x_p) = \sigma_r^2 + {\sigma_s}^2 y_p \tag{7}$$

$$E(x_p) = y_p \tag{8}$$

where E is local mean, D is variance of noise, x_p is the degraded image pixel value, y_p is the original image pixel value, and σ_s and σ_r are the noise parameters. The two formulas can be simplified to acquire:

$$D(x_p) = \sigma_r^2 + {\sigma_s}^2 E(x_p) \tag{9}$$

Equation (9) is derived from Equations (7) and (8). Equation (7) is a quadratic equation of σ_s and σ_r, which has an infinite number of solutions about the variables σ_s and σ_r. To obtain the unique solution of σ_s and σ_r, it is necessary to select multiple homomorphic sub-blocks, then a set of linear equations composed of binary linear equations in the form of Equation (9) can be obtained. The number of sub-equations depends on the homomorphic sub-blocks quantity. Typically, the number of image homomorphic sub-blocks are more

than two, thus the system of equations is an overdetermined system of equations. The overdetermined systems of equations are expressed.

$$\sum_{p\in H}\left[\sigma_r^2+{\sigma_s}^2E(x_p)-D(x_p)\right]^2=0 \tag{10}$$

In this paper, Equation (10) is solved by the least squares method. By taking the partial derivative of σ_s and σ_r in Equation (10), we can acquire both of the equations in Equation (11). The solutions of σ_s and σ_r are transformed into solving the system of linear equations shown below. We polluted the images of the Set12 dataset with noise $\sigma_s = 2.0 \times 10^{-2}$, $\sigma_r = 10^{-1}$, and then estimated the average value of noise parameters obtained by Equation (11) to be $\sigma_s = 1.978 \times 10^{-2}$, $\sigma_r = 1.15 \times 10^{-1}$, with an error within 0.02.

$$\begin{cases} \sum_{p\in H}\left[\sigma_r^3+{\sigma_s}^2\sigma_rE(x_p)-\sigma_rD(x_p)\right]=0 \\ \sum_{p\in H}\left[\sigma_s\sigma_r^2E(x_p)+\sigma_s^3E^2(x_p)-\sigma_sD(x_p)E(x_p)\right]=0 \end{cases} \tag{11}$$

At this point, the unknown parameters σ_s and σ_r of the model have unique solutions that satisfy the least squares method.

3.1.2. The Image Restoration Algorithm on Local Statistical Features

It can be seen from the above reasoning that the variance corresponding to the observations of the Gaussian signal-dependent noise model is as follows:

$$\sigma^2(x_p)=\sigma_r^2+\sigma_s^2y_p \tag{12}$$

Arranging the above formula will result in $y_p=\frac{\sigma^2(x_p)-\sigma_r^2}{\sigma_s^2}$. Since the parameters σ_s and σ_r can be estimated by solving the above equations, just solve for $\sigma^2(x_p)$ to acquire the preliminary image. According to the assumption proposed by Lee [28], the mean $\mu(x_p)$ and variance $\sigma^2(x_p)$ of x_p are determined by the mean and variance of the observations in the neighborhood pixels point of the center pixel p. Thus, $\sigma^2(x_p)$ can be obtained as follows:

$$\sigma^2(x_p)=\frac{1}{M_t}\sum_{q\in W_p}\left(x_q-\mu(x_p)\right)^2 \tag{13}$$

where M_t is the number of pixels in the neighborhood. Substituting the expression of $\sigma^2(y_p)$ into the above formula will result in:

$$y_p=\frac{1}{\sigma_s^2M_t}\sum_{q\in W_p}\left(x_q-\mu(x_p)\right)^2-\frac{\sigma_r^2}{\sigma_s^2} \tag{14}$$

3.2. Convolutional Neural Network

The preliminary image obtained by the above image restoration algorithm based on the statistical characteristics of the local regions inevitably has two errors. The first error is caused when estimating the parameters σ_s and σ_r. The second error is brought about by the local statistical features of the selected pixels. Therefore, the denoising effects of the preliminary image are not ideal. The following two errors caused by the above algorithm are reduced by the convolutional neural network.

3.2.1. Network Design

The network in this paper employed a simple design, which can reduce the amount of calculation and speed up the training while completing the task. The function of the designed network is mainly to reduce the error of the preliminary image. The convolutional network structure is illustrated in Figure 2. The network takes the preliminary image and noisy image as input, and then extracts the shallow features of the image by a convolutional layer, and then extracts the deep features of the image through the residual network. The residual network is composed of a set of 6 residual blocks. Deeper networks are used

to extract deep features. However, after the network layer is deepened, the gradient disappearance and gradient explosion make the training of the network meaningless. Therefore, the residual network is used to avoid the problem, which not only ensures the effect but also controls speed. After the residual network extracts the deep-level features, the data are up_sampled using a up_convolutional layer and the image is returned to the original size. The last convolution layer fuses the features into a single-channel image. The size of the convolution kernel used in the first convolutional layer and the last convolutional layer are both 9×9, and the size of the convolution kernel used by all the remaining convolutional layers is 3×3. The convolution layers in the residuals block are all 128 convolution kernels with a stride of 1. Only in this way can each layer in the residual network output the same size as the feature map. It helps to fuse features. The LeakyReLU activation function with a slope of 0.2 is employed in the paper, which fixes the problem that the non-positive gradient of ReLU is 0, causing some parameters never to be updated. This activation function has a small positive slope in the negative area, thus it can backpropagate even with negative input values. Furthermore, after experiments, it is found that batch normalization is hardly helpful for the denoising of real images. Part of the reason is that the real noise distribution is fundamentally different from the Gaussian distribution. The architecture of the proposed network is shown in Figure 3a, and the residual connection is shown in Figure 3b. Table 1 shows the parameter settings of Convolutional Neural Networks.

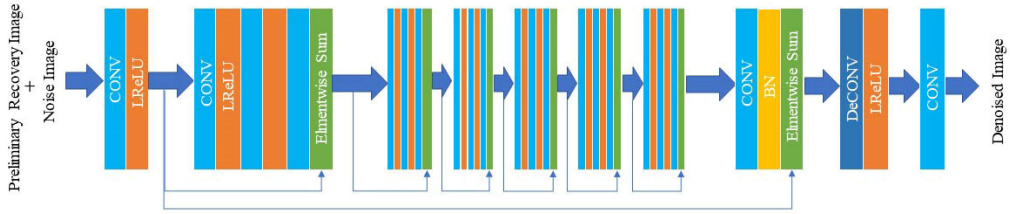


Figure 2. Convolutional network diagram.

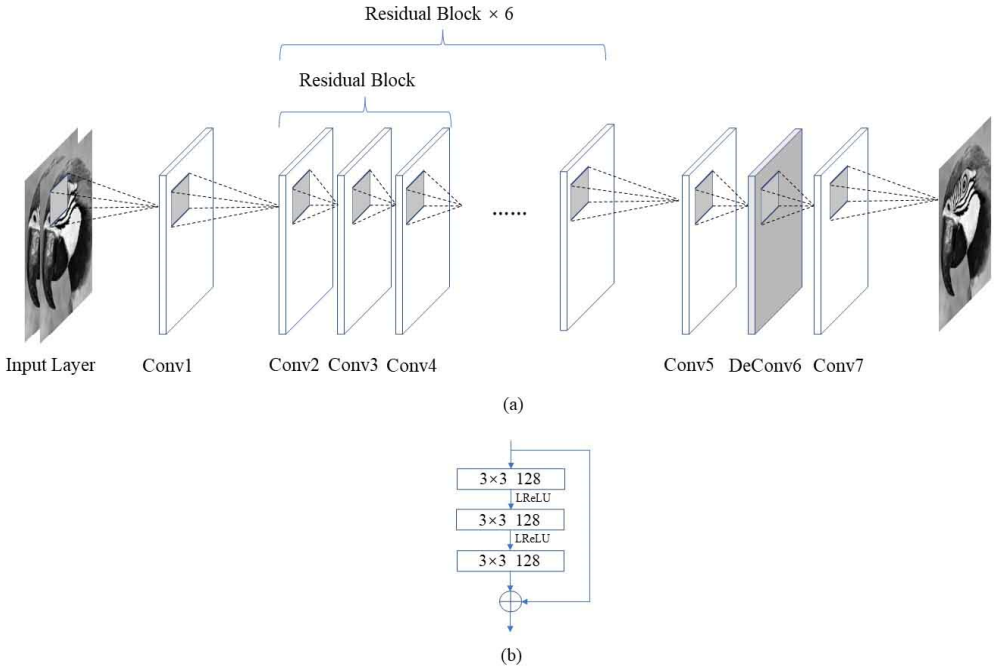


Figure 3. The designed network details of each convolutional layer. (**a**) Convolutional neural network structure. (**b**) Residual connection.

Table 1. The parameter settings of Convolutional Neural Network.

Layer	Kernel	Leaky-ReLU	Stride
Conv1	$9 \times 9 \times 64$	Yes	1
Conv2	$3 \times 3 \times 128$	Yes	1
Conv3	$3 \times 3 \times 128$	Yes	1
Conv4	$3 \times 3 \times 128$	No	1
Conv5	$3 \times 3 \times 128$	No	1
DeConv6	$3 \times 3 \times 256$	Yes	1
Conv7	$9 \times 9 \times 1$	No	1

3.2.2. Loss Function Design

The mean squared error (MSE) below is used as the objective function (the loss function) to measure the difference between the predicted image $R(Y_j)$ and the corresponding ground truth value $Y_j - X_j$. X_j represents the jth clean image. θ represents the parameters of the trained model in the network. $\{(Y_j, X_j)\}_{j=1}^{N}$ represents there are N image pairs of predicted images and real images. Potentially clean images are recovered by the Adam optimizer using a loss function.

$$L(\theta) = \frac{1}{2N}\sum_{j=1}^{N}\left|\left|R(Y_j, \theta) - (Y_j - X_j)\right|\right|_2^2 \tag{15}$$

4. Experiment

In this section, we assess the performance of the proposed method and compare it with other methods to show the improvement of the method relative to other methods.

The experimental environment comprised an Intel Core i7-9700k CPU, 128 GB RAM, and a NVIDIA GeForce RTX 2070 GPU. In addition, the deep learning framework used was Pytorch.

The paper uses the Berkeley Segmentation Dataset (Berkeley segmentation dataset, BSD) (BSD500 for short) as the network training dataset. The BSD500 dataset contains 500 grayscale images, and all images are of size 481×321. In the experiment, 200 sets of noise are added to each image to construct an image with real noise. Therefore, the training dataset consists of 10^5 training samples in total. The noise parameters σ_s and σ_r are randomly sampled from $[10^{-3}, 10^{-2}]$ to $[10^{-2.5}, 10^{-1}]$, respectively.

In the selection of test data sets, we use Kodak, McMaster, and Set12 as test data sets, and compare the more advanced methods proposed on these data sets. Although KPN [3] is designed for multi-frame image input, it can be adjusted to a single-frame image by changing the network input to compare with our method.

For comparison with other methods, we evaluated the noise levels corresponding to three fixed sets of σ_s and σ_r. Similar to [29], gamma correction was performed first, then we added signal-dependent Gaussian noise, took the generated noise image as the input: $N(0, \sigma_r^2 + \sigma_s y_p)$, where y_p was the intensity of image pixels and the noise parameters σ_s and σ_r were randomly sampled from $[10^{-3}, 10^{-2}]$ to $[10^{-2.5}, 10^{-1}]$. Furthermore, similar to [4], the noise level was estimated as: $\sqrt{\sigma_r^2 + \sigma_s y_p}$. Note that the noise Equation (6) assigned three single sets of parameter values, i.e., σ_1 ($\sigma_s = 2.0 \times 10^{-2}$, $\sigma_r = 10^{-1}$), σ_2 ($\sigma_s = 6.0 \times 10^{-2}$, $\sigma_r = 2 \times 10^{-1}$), and σ_3 ($\sigma_s = 9.0 \times 10^{-2}$, $\sigma_r = 3 \times 10^{-1}$), to generate clean-noisy image pairs for image denoising. We claim that harnessing noise Equation (6) with combinations of different parameter settings may lead to better results. In spite of that, it was found that the parameter setting as above is a more robust candidate than several other choices.

Similar to most CNNs used for image denoising, the optimizer chooses the adaptive moment estimation (Adam) method in setting network parameters, and the loss function is

MSE. The whole training process contains 180 epochs, and the batch size is set to 20, thus the training data set can be divided into 5000 batches. However, for each epoch, we only sample the first 3000 batches for training. Before the next epoch training, we shuffle the order of all batches and select the top 3000 batches for training. The training adopts a linear piecewise learning rate, the learning rate of the top 90 epochs is 10^{-3}, the next 60 epochs are 10^{-4}, and the last 30 epochs are 10^{-5}.

To test the effectiveness of each block for our network on image denoising, we used ablation experiments to analyze the performance. The images in the Set12 dataset were selected for the experiment, and the noise level σ_1 ($\sigma_s = 2.0 \times 10^{-2}$, $\sigma_r = 10^{-1}$) was added. Then, we denoised the images separately in groups. The denoising results are shown in Table 2, which shows the PSNR. The results show that residuals made the denoising effect better. Additionally, the addition of preliminary image made the denoising effect better.

Table 2. The average PSNR(DB) results of different methods on three datasets. (a) Remove three residual blocks. (b) Remove all residual blocks. (c) Remove the preliminary image from the input. (d) Original network.

Method	a	b	c	d
PSNR	28.67	29.61	31.24	33.50

The paper selects four methods: KPN [3], CBDNet [24], SRNet [14], and WCNN [13]. The link to the code is posted in references [30–33]. Five methods are compared on all the datasets used in the paper. KPN generates a stack of per-pixel filter kernels that jointly aligns, averages, and denoises a burst to produce a clean version of the reference frame. The establishment of the CBD model takes into account both the Poisson–Gaussian model and the in-camera processing pipeline. However, the Poisson–Gaussian model does not have a wider range of scenarios than the signal-dependent Gaussian model in the paper. Each subnetwork of WCNN has its input and output and is supervised by its loss function to capture image features with a specific direction and scale, allowing the WCNN to have sufficient convolutional layers to capture image features while avoiding vanishing/exploding gradients. SRNet embeds the CNN into the sparse representation framework. In each phase, two subnetworks are designed with MSR block to model the updating of the sparse coefficient and image, respectively. However, it suffers from performance bottlenecks and large time consumption.

Figure 4a–d is a diagram of each stage of the paper. To see the contrast, we enlarge the green part of the image into the red part. It can be seen that the images have more detail loss and less clarity even if the preliminary image can see the denoising effect. The denoising effect of the restored image is better. The picture detail loss is less, the recovery effect is better.

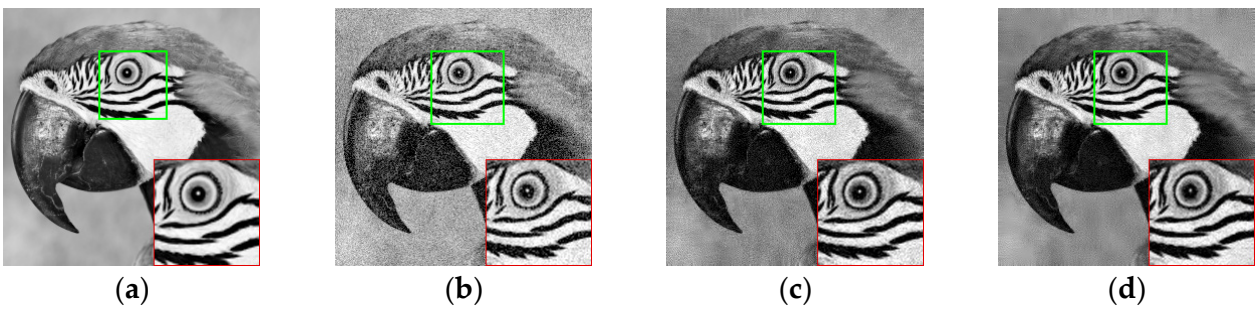


Figure 4. The denoising process of our algorithm. (**a**) Original image. (**b**) Noise image (σ_3 ($\sigma_s = 9.0 \times 10^{-2}$, $\sigma_r = 3 \times 10^{-1}$)). (**c**) The preliminary image. (**d**) The final image.

Figure 5a–e shows a visual comparison of different algorithms in the Set12 dataset. The KPN image effect is normal, the details are lost, and the denoising effect of the shadow area is not very ideal. CBDNet is better than KPN in detail recovery. It loses less detail

and restores the shadow part well, but the overall image is not clear. CBDNet is better at processing low-noise images. While SRNet and WCNN performed well in terms of image sharpness and detail, there is still room for improvement. The algorithm proposed in this paper can recover image details more clearly and obtain a better visual effect. It still does not effectively remove all real noise, because the noise model does not perfectly represent real noise.

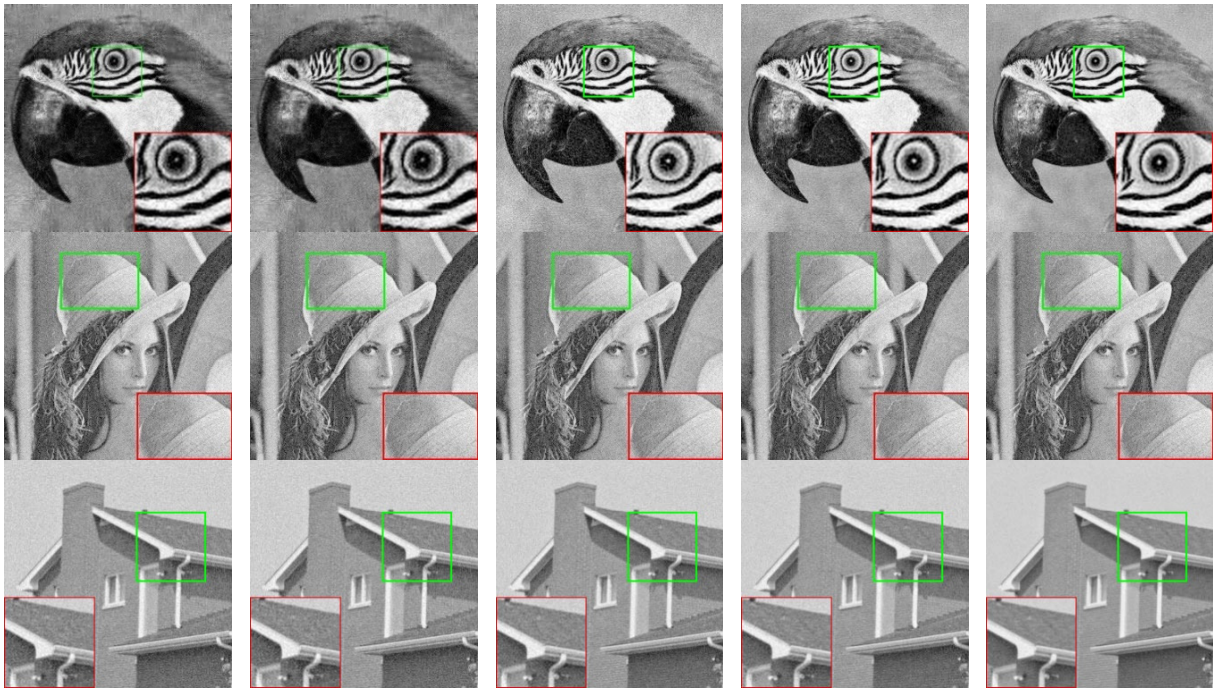


Figure 5. Denoising results of the images on the Set12 dataset by different methods. (**a**) KPN [3]. (**b**) CBDNet [24]. (**c**) SRNet [14]. (**d**) WCNN [13]. (**e**) Ours.

Table 3 shows the average PSNR in Set12 data sets. Table 4 shows the average SSIM of different algorithms in Set12 data sets. The average PSNR of the algorithm in the paper is higher than the other algorithms. Except for low-noise images, the denoising effect of CBDNet is slightly better than that of the method in this paper. The performance of SRNet and WCNN is similar to that of the proposed algorithm. However, in terms of SSIM, our algorithm is superior to other algorithms. It shows that the proposed algorithm is superior to the previous algorithms in denoising performance. The average SSIM index of the restored image is close to that of the original image, indicating that the quality of the restored image is close to that of the original noiseless image.

Table 3. The average PSNR(DB) results of different methods on three datasets.

Dataset	Noise Level	KPN [3]	CBDNet [24]	SRNet [14]	WCNN [13]	Ours
McMaster	σ1	32.69	34.34	33.76	33.65	34.31
	σ2	31.09	31.51	31.73	31.92	33.25
	σ3	29.76	29.84	30.42	30.78	31.03
Set12	σ1	30.96	33.53	33.11	32.91	33.50
	σ2	29.82	30.88	31.16	31.23	31.52
	σ3	28.59	28.79	29.28	29.57	29.95
Kodak	σ1	31.98	35.71	34.95	34.71	35.65
	σ2	30.57	31.86	32.44	32.93	33.63
	σ3	29.45	29.79	30.07	30.48	31.46

Table 4. The average SSIM results of different methods on three datasets.

Dataset	Noise Level	KPN [3]	CBDNet [24]	SRNet [14]	WCNN [13]	Ours
McMaster	σ1	0.684	0.785	0.727	0.723	0.783
	σ2	0.615	0.661	0.689	0.697	0.715
	σ3	0.564	0.613	0.621	0.635	0.651
Set12	σ1	0.668	0.729	0.710	0.701	0.724
	σ2	0.632	0.655	0.642	0.646	0.708
	σ3	0.551	0.577	0.598	0.585	0.604
Kodak	σ1	0.718	0.789	0.771	0.762	0.783
	σ2	0.684	0.724	0.735	0.738	0.761
	σ3	0.642	0.665	0.689	0.691	0.710

Figure 6 shows the average PSNR of our algorithm on the Set12 test set at the noise level of σ_3 ($\sigma_s = 9.0 \times 10^{-2}$, $\sigma_r = 3 \times 10^{-1}$). It shows the line graph of PSNR changing with the number of iterations. As the number of iterations increases, PSNR gradually flattens out.

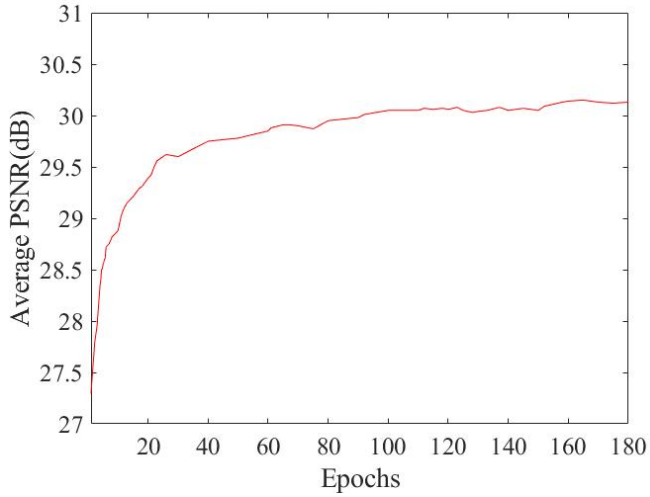


Figure 6. The relationship between the number of iterations Epoch and PSNR.

Figure 7 shows the loss of our algorithm on the Set12 test set at the noise level of σ_3 ($\sigma_s = 9.0 \times 10^{-2}$, $\sigma_r = 3 \times 10^{-1}$). It shows the line graph of loss changing with the number of iterations. As the number of iterations increases, loss decreases and gradually flattens out.

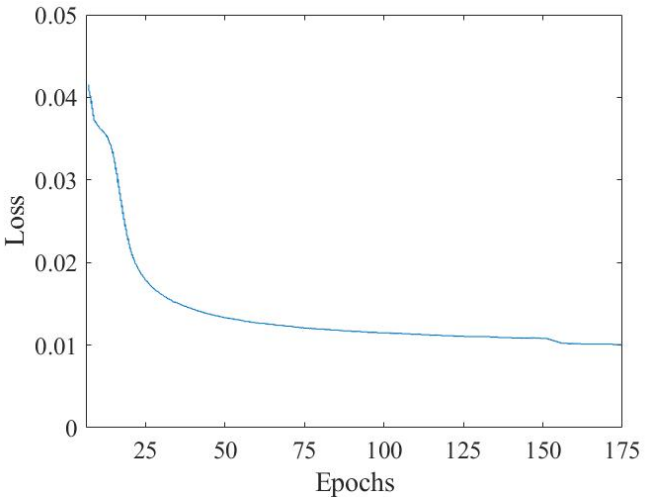
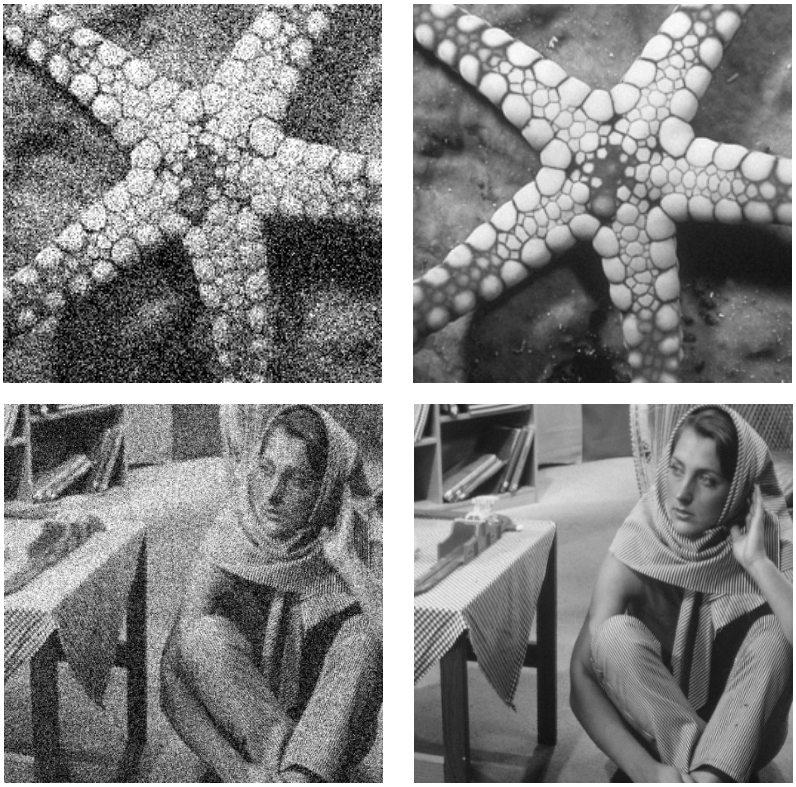


Figure 7. The relationship between the number of iterations of Epoch and Loss.

To verify the denoising effect of our algorithm under high noise, we added a set of denoising experiments σ ($\sigma_s = 9.0 \times 10^{-1}$, $\sigma_r = 3.0 \times 1$) for verification. Figure 8a,b shows a comparison of the denoising effect of the proposed method in the case of high noise.



(**a**) (**b**)

Figure 8. Image denoising comparison under high noise σ ($\sigma_s = 9.0 \times 10^{-1}$, $\sigma_r = 3.0 \times 1$). (**a**) High noise image. (**b**) Denoising image.

To verify the performance of the denoising algorithm in real scenes, we selected the α57 digital cameras launched by Sony to collect 30 images. To evaluate the performance of our method in terms of visual quality, we performed a subjective evaluation as a qualitative analysis experiment. The evaluation was carried out as a rating experiment where 40 users experienced in image processing viewed, in random order, a set of images. The users were asked to provide ratings for each image according to the following attributes: noise and detail. The final result of the ratings averaged over the observers is illustrated in Figures 9 and 10.

Figure 11a–f shows two images we randomly selected. From the first picture, the details of the houses in the shaded part of the image restored by KPN are missing, and the overall effect of the image is not ideal. The local detail of the CBDNet algorithm is stronger than KPN, but the overall visual effect is a bit fuzzy, and the image quality is not good. WCNN and SRNet are better than the first two algorithms in terms of sharpness and image detail, but there is still room for improvement. In the marked areas in the image, we can see that other algorithms are rough in detail recovery in the shadow area, and the denoising effect is not ideal. Our algorithm performs well in this respect. The algorithm in this paper has the strongest local detail retention ability, and the restored image has fine details and a clear texture. In the second picture, KPN is missing some details in the recovered image. The text in the book is obscured, and the overall picture is poor in definition. In the image recovered by CBDNet, the text of the book can be seen, but it is not clear, and the overall image is a little fuzzy. The images recovered by WCNN and SRNet are good, but lower than the algorithm in this paper. In the marked areas in the images, we can see that the labels in the images recovered by other algorithms are fuzzy, and the denoising effect around the

labels is not ideal, while the label recovered by our algorithm is clear and the denoising effect is ideal. The algorithm in this paper has a good performance in detail recovery and overall quality.

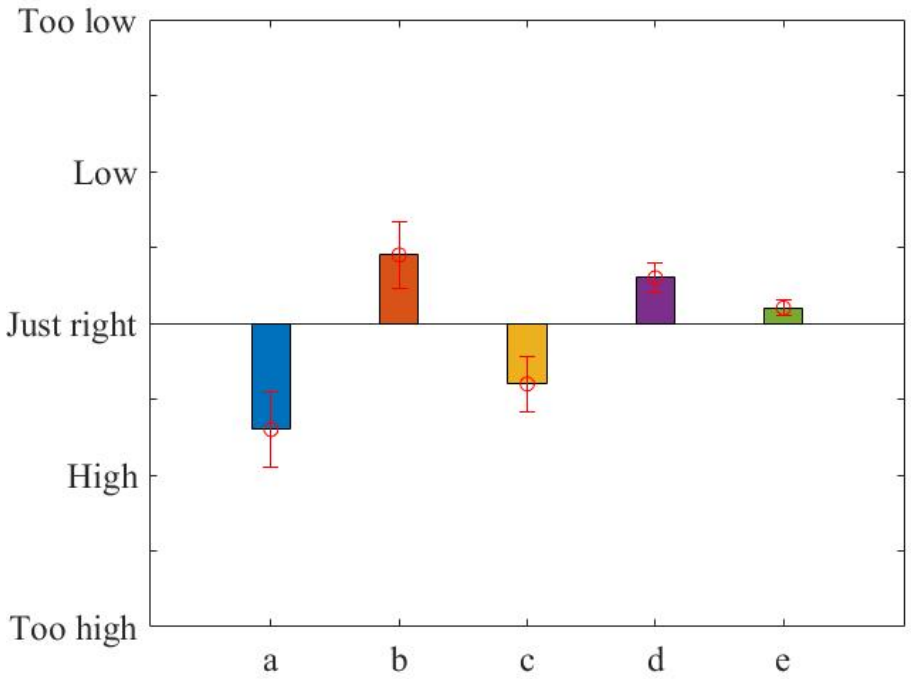


Figure 9. Qualitative analysis result of noise, showing the average ratings from the conducted experiment, with error bars for the standard errors. (**a**) KPN [3]. (**b**) CBDNet [24]. (**c**) SRNet [14]. (**d**) WCNN [13]. (**e**) Ours.

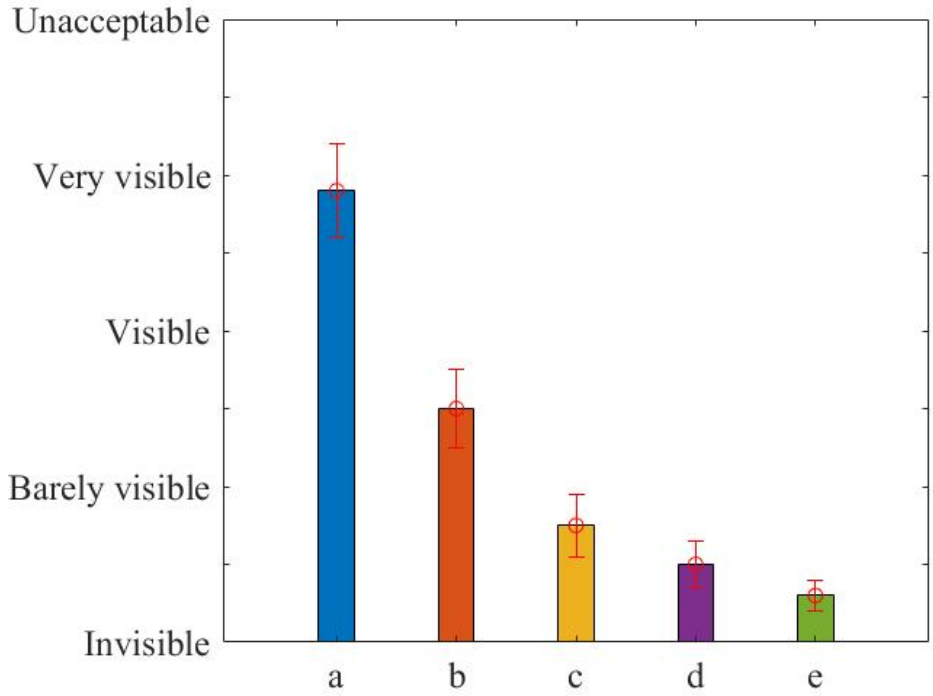


Figure 10. Qualitative analysis result of detail, showing the average ratings from the conducted experiment, with error bars for the standard errors. (**a**) KPN [3]. (**b**) CBDNet [24]. (**c**) SRNet [14]. (**d**) WCNN [13]. (**e**) Ours.

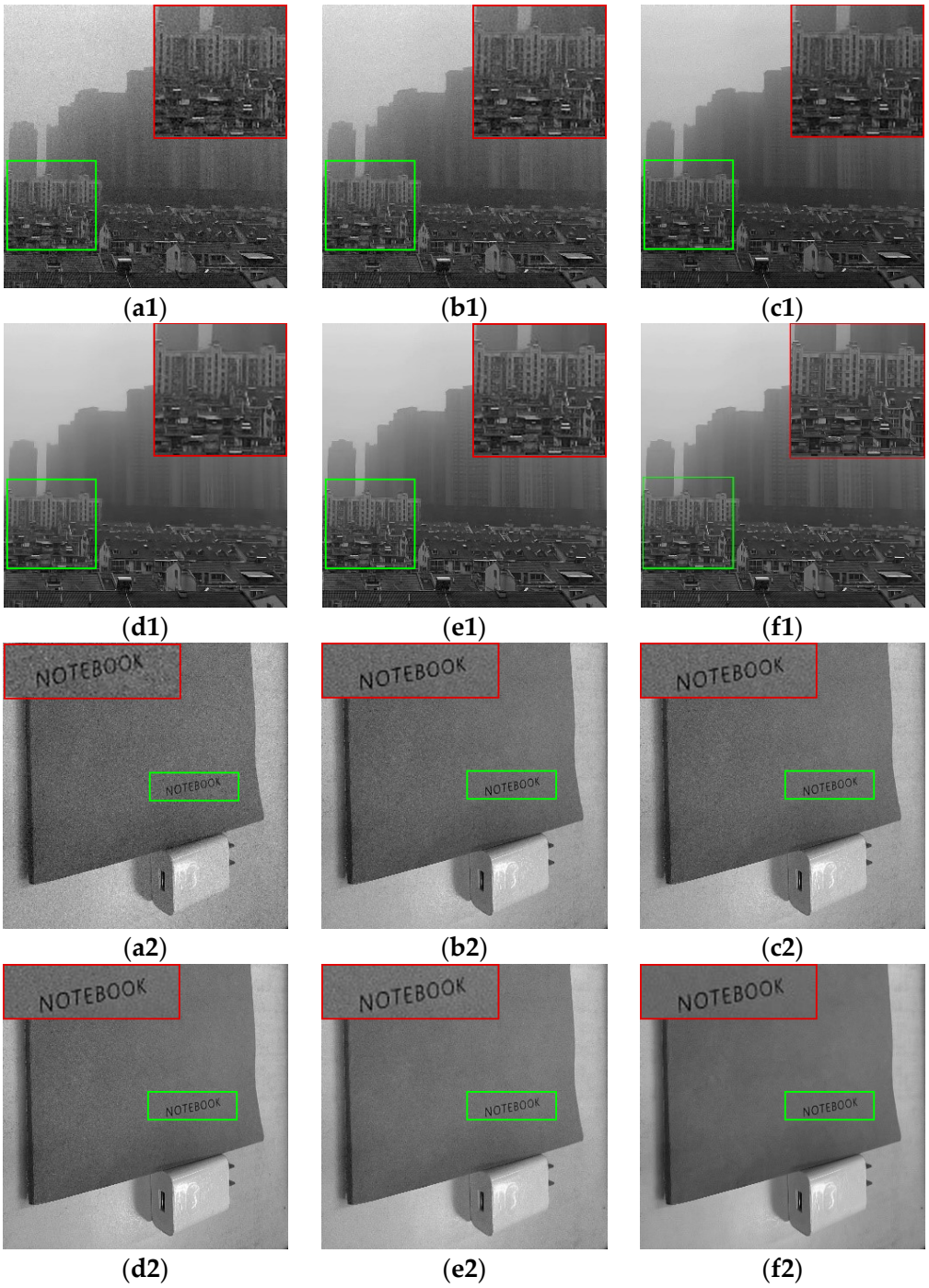


Figure 11. Image denoising comparison in real scenes. (**a1**,**a2**) The original image. (**b1**,**b2**) KPN [3]. (**c1**,**c2**) CBDNet [24]. (**d1**,**d2**) SRNet [14]. (**e1**,**e2**) WCNN [13]. (**f1**,**f2**) Ours.

Figure 12a,b shows the comparison before and after denoising of our algorithm. It can be intuitively seen from comparison figures that the noise in the images is greatly reduced after denoising by our algorithm. The image recovery effect is good, and the details are perfectly restored.

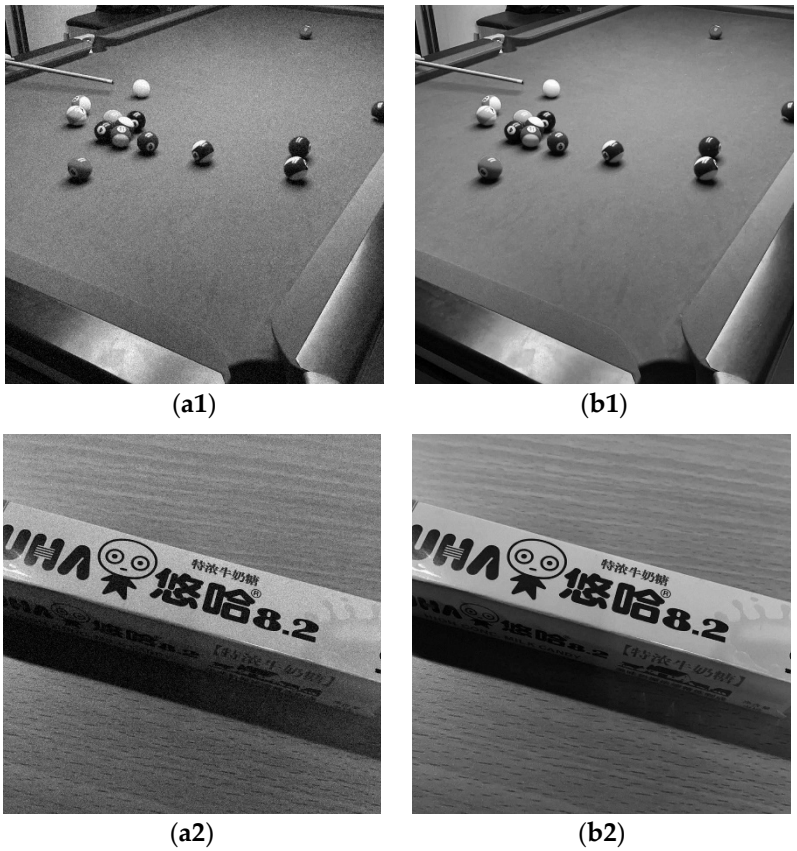


Figure 12. Image denoising comparison. (**a1**,**a2**) The original image. (**b1**,**b2**) Ours.

In general, our algorithm can reconstruct the detailed features more successfully. Although it does not achieve the best denoising effect in low noise images, the overall denoising effect is better and the restored images are clear, which is better than other algorithms.

5. Conclusions

The paper presents a deep signal-dependent denoising noise algorithm. There are two main stages to the work. Firstly, the noise parameters of the image are estimated by the noise parameters, and then the preliminary image is obtained according to the local statistical characteristics. Secondly, using the initial recovery map as input to the network along with the original map, take the initial restored image and the original image together as the input of the network, and the convolutional neural network can be used to reduce the error of the preliminary image and make the recovered image closer to the original image. We use the noise level of the noise image and the noise image together as the input of the convolutional neural network to obtain a wider range of noise levels than the single noise image as the input. It makes the denoising algorithm more widely used and has a better effect on detail recovery. We compared and analyzed denoising algorithms based on subjective vision. It was verified that the algorithm in this paper has a relatively evident inhibitory effect on signal-dependent noise, and the recovered images are of higher quality. This paper deals with grayscale images. For the future work, we hope to denoise color noise images. We believe that a combination of traditional image denoising and deep learning will become a great way to deal with noisy images.

Author Contributions: Conceptualization, L.Z.; methodology, L.Z. and J.W.; software, S.L.; validation, L.Z., S.L. and J.W.; formal analysis, S.L.; investigation, S.L.; resources, J.W.; data curation, L.Z.; writing—original draft preparation, S.L.; writing—review and editing, J.W.; visualization, J.W.; supervision, L.Z.; project administration, L.Z. and J.W.; funding acquisition, L.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Quzhou Science and Technology Plan Project, grant number 2022K108, and the Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014; and Basic Scientific Research Foundation Project of Provincial Colleges and Universities in Hei longjiang Province, grant number 2022KYYWF-FC05.

Data Availability Statement: As the algorithm in this paper relates to the project currently in progress, the project is private for some reasons, so the code is not convenient to be made public. If any reader wants to obtain the code of this article, you can send us an email. After receiving the email, we will determine whether to give the code to the reader after evaluation.

Acknowledgments: The authors acknowledge the Quzhou Science and Technology Plan Project (grant number 2022K108) and the Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Short Name	Full Name
CNNs	Convolutional Neural Networks
AWGN	Additive White Gaussian Noise
DnCNN	Feed-forward Denoising Convolutional Neural Networks
NLF	Noise Level Function
BDE	Bayesian Deep Ensemble
DRNE	Deep Residual Noise Estimator
KPN	Kernel Prediction Network
MKPN	Multi-Kernel Prediction Networks
RED	Regularization by Denoising
WCNN	Wide CNN
MSR-block	Multiscale Residual Block
CBDNet	Convolutional Blind Denoising Network
NERNet	Noise Estimation module and Removal Network
SDNet	Software Defined Network
SRNet	Sparse Representation-based Network
MUS-ADD	Multi Scaling Aided Double Decker
ADMM	Alternating Direction Method of Multipliers
MS-WGSC	Multi-Scale Weighted Group Sparse Coding
AWTD	Adaptive Weighted Tensor
CRCNN	Cascaded and Recursive Convolutional Neural Network
KCRD	Kronecker Component with Low-Rank Dictionary
GRDN	Grouped Residual Dense Network
MSE	Mean Squared Error
PSNR	Peak Signal-to-Noise Ratio
SSIM	Structural Similarity

References

1. Radlak, K.; Malinski, L.; Smolka, B. Deep learning based switching filter for impulsive noise removal in color images. *Sensors* **2020**, *20*, 2782. [CrossRef]
2. Zhang, W. Digital image processing method for estimating leaf length and width tested using kiwifruit leaves (*Actinidia chinensis Planch*). *PLoS ONE* **2020**, *15*, e0235499. [CrossRef]
3. Mildenhall, B.; Barron, J.T.; Chen, J.; Sharlet, D. Burst denoising with kernel prediction networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; Volume 13, pp. 2502–2510.
4. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image de-noising. *IEEE Trans. Image Process.* **2017**, *26*, 3142–3155. [CrossRef]
5. Foi, A.; Trimeche, M.; Katkovnik, V.; Egiazarian, K. Practical Poissonian-Gaussian Noise Modeling and Fitting for Single-Image Raw-Data. *IEEE Trans. Image Process.* **2008**, *17*, 1737–1754. [CrossRef]
6. Liu, X.; Tanaka, M.; Okutomi, M. Practical Signal-Dependent Noise Parameter Estimation From a Single Noisy Image. *IEEE Trans. Image Process.* **2014**, *23*, 4361–4371. [CrossRef]

7. Liu, X.; Tanaka, M.; Okutomi, M. Signal dependent noise removal from a single image. *IEEE Int. Conf. Image Process. IEEE* **2014**, *20*, 2679–2683.
8. Tan, H.; Xiao, H.; Lai, S.; Liu, Y.; Zhang, M. Pixelwise Estimation of Signal-Dependent Image Noise Using Deep Residual Learning. *Comput. Intell. Neurosci.* **2019**, *30*, 45–46. [CrossRef]
9. Marinc, T.; Srinivasan, V.; Gul, S.; Hellge, C.; Samek, W. Multi-Kernel Prediction Networks for Denoising of Burst Images. In Proceedings of the 2019 IEEE International Conference on Image Processing, Taipei, Taiwan, 22–25 September 2019; Volume 15, pp. 2404–2408. [CrossRef]
10. Guo, B.; Song, K.; Dong, H.; Yan, Y.; Tu, Z.; Zhu, L. NERNet: Noise estimation and removal network for image denoising. *J. Vis. Commun. Image Represent.* **2020**, *71*, 102851. [CrossRef]
11. Qi, G.; Hu, G.; Mazur, N. A novel multi-modality image simultaneous denoising and fusion method based on sparse representation. *Computers* **2021**, *10*, 129. [CrossRef]
12. Xie, Z.; Liu, L.; Wang, C. Model-guided boosting for image denoising. *Signal Process.* **2022**, *201*, 108721. [CrossRef]
13. Liu, G.; Dang, M.; Liu, J.; Xiang, R.; Tian, Y.; Luo, N. True wide convolutional neural network for image denoising. *Inf. Sci.* **2022**, *610*, 171–184. [CrossRef]
14. Sheng, J.; Lv, G.; Wang, Z.; Feng, Q. SRNet: Sparse representation-based network for image denoising. *Digit. Signal Process.* **2022**, *130*, 103702. [CrossRef]
15. Tojo, L.; Maik, V.; Devi, M. Image Denoising Using Multi Scaling Aided Double Decker Convolutional Neural Network. *Optik* **2022**, *13*, 170350. [CrossRef]
16. Xu, S.; Zhang, J.; Wang, J.; Sun, K.; Zhang, C.; Liu, J.; Hu, J. A model-driven network for guided image denoising. *Inf. Fusion* **2022**, *85*, 60–71. [CrossRef]
17. Ou, Y.; Swamy, M.; Luo, J.; Li, B. Single image denoising via multi-scale weighted group sparse coding. *Signal Process.* **2022**, *200*, 108650. [CrossRef]
18. Zhang, L.; Liu, C. Robust Low-Rank Analysis with Adaptive Weighted Tensor for Image Denoising. *Displays* **2022**, *73*, 102200. [CrossRef]
19. Li, X.; Meng, X.; Xiong, B. A fractional variational image denoising model with two-component regularization terms. *Appl. Math. Comput.* **2022**, *427*, 127178. [CrossRef]
20. Khowaja, S.A.; Yahya, B.N.; Lee, S.L. Cascaded and Recursive ConvNets (CRCNN): An effective and flexible approach for image denoising. *Signal Process. Image Commun.* **2021**, *99*, 116420. [CrossRef]
21. Zhang, L.; Liu, C. Kronecker component with robust low-rank dictionary for image denoising. *Displays* **2022**, *74*, 102194. [CrossRef]
22. Phan, T.D.K. A weighted total variation based image denoising model using mean curvature. *Optik* **2020**, *217*, 164940. [CrossRef]
23. Yao, H.; Zou, M.; Qin, C.; Zhang, X. Signal-Dependent Noise Estimation for a Real-Camera Model via Weight and Shape Constraints. *IEEE Trans. Multimed.* **2021**, *24*, 640–654. [CrossRef]
24. Guo, S.; Yan, Z.; Zhang, K.; Zuo, W.; Zhang, L. Toward Convolutional Blind Denoising of Real Photographs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; Volume 16, pp. 1712–1722. [CrossRef]
25. Zhao, H.; Shao, W.; Bao, B.; Li, H. A simple and robust deep convolutional approach to blind image denoising. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27 October–2 November 2019; Volume 29.
26. Kim, D.W.; Ryun Chung, J.; Jung, S.W. Grdn: Grouped residual dense network for real image denoising and gan-based real-world noise modeling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–20 June 2019; Volume 14.
27. Akyüz, A.O.; Reinhard, E. Noise reduction in high dynamic range imaging. *J. Vis. Commun. Image Represent.* **2007**, *18*, 366–376. [CrossRef]
28. Lee, J.-S. Digital Image Enhancement and Noise Filtering by Use of Local Statistics. *IEEE Trans. Pattern Anal. Mach. Intell.* **1980**, *2*, 165–168. [CrossRef]
29. Plotz, T.; Roth, S. Benchmarking denoising algorithms with real photographs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; Volume 17, pp. 1586–1595.
30. Available online: https://paperswithcode.com/paper/burst-denoising-with-kernel-prediction (accessed on 5 November 2022).
31. Available online: https://paperswithcode.com/paper/toward-convolutional-blind-denoising-of-real (accessed on 8 November 2022).
32. Available online: https://github.com/JiechaoSheng/SRNet (accessed on 9 November 2022).
33. Available online: https://paperswithcode.com/paper/dc-wcnn-a-deep-cascade-of-wavelet-based (accessed on 11 November 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Research on Spectrum Prediction Technology Based on B-LTF

Xue Wang, Qian Chen and Xiaoyang Yu *

School of Measurement and Communication Engineering, Harbin University of Science and Technology, Harbin 150080, China

* Correspondence: 2220600068@stu.hrbust.edu.cn

Abstract: With the rapid development of global communication technology, the problem of scarce spectrum resources has become increasingly prominent. In order to alleviate the problem of frequency use, rationally use limited spectrum resources and improve frequency utilization, spectrum prediction technology has emerged. Through the effective prediction of spectrum usage, the number of subsequent spectrum sensing processes can be slowed down, and the accuracy of spectrum decisions can be increased to improve the response speed of the whole cognitive radio technology. The rise of deep learning has brought changes to traditional spectrum predicting algorithms. This paper proposes a spectrum predicting method called Back Propagation-Long short-term memory Time Forecasting (B-LTF) by using Back Propagation-Long Short-term Memory (BP-LSTM) network model. According to the historical spectrum data, the future spectrum trend and the channel state of the future time node are predicted. The purpose of our research is to achieve dynamic spectrum access by improving the accuracy of spectrum prediction and better assisting cognitive radio technology. By comparing with BP, LSTM and Gate Recurrent Unit (GRU) network models, we clarify that the improved model of recurrent time network can deal with time series more effectively. The simulation results show that the proposed model has better prediction performance, and the change in time series length has a significant impact on the prediction accuracy of the deep learning model.

Keywords: cognitive radio (CR); spectrum prediction; deep learning; long short-term memory network (LSTM); long sequence time-series forecasting (LSTF)

1. Introduction

Information has changed human life, from sharing bicycles to sharing everything, from the commercialization of 5G to the research and development of 6G. In today's era, communication technology is vigorously developed, and the low-Earth orbit satellite Internet of things is used as the architecture to create a blue ocean of satellite Internet of things applications to realize the Internet of everything. With a large number of devices being connected to the wireless network, the scarcity of wireless spectrum resources is becoming increasingly serious, and the whole wireless communication environment has become extremely severe. The current spectrum resource allocation strategy is mainly a static spectrum allocation strategy, which licenses the spectrum to a limited number of users. The decision stipulates that even if the licensed band is idle, other users are still not able to access and use it, which results in a shortage of spectrum resources, but it is a lot of waste.

According to the research report of the Federal Communications Commission (FCC) of the United States, the spectrum occupancy of different licensed frequency bands varies due to the different number of services and requirements, ranging from 15% to 85% [1]. Additionally, according to the survey of global spectrum utilization in recent years, the spectrum utilization rate is very low, and there is a large amount of spectrum waste worldwide. The American University of Sharjah measured the local UHF band (300 MHz to 3000 MHz) in 2017, and the study showed that the average occupancy rate was between 10% and 35% [2]. Yu et al. used R&S EM100 digital compact receiver to measure the

spectrum occupancy of China's radio broadcast allocated frequency band from 87.5 MHz to 108 MHz, and the results showed that there were a large number of spectrum holes [3]. In 2018, spectrum measurements of 6 GHz band models at three different locations in San Luis Potosi, Mexico, showed that the spectrum utilization rate was only 4.73% [4]. In 2019, using RF Explorer 6G Combo to detect the spectrum occupancy of the GSM900 band in the Samsun area, and the calculation results showed that the occupancy rate was about 8.5% and 82% at -40 dBm and -75 dBm, respectively [5]. The 24 h detection was carried out on the frequency band from 30 MHz to 1030 MHz in Pakistan, and the results showed that the average spectrum utilization rate was only 35.41%, among which the minimum occupancy rate of the 850 MHz to 1030 MHz band was only 25% [6]. As can be seen in the above data, spectrums in large Numbers in the world cannot be effectively used, according to research, at any time and any place, the average spectrum utilization rate is not more than 5% [7].

As early as 1999, in order to solve the problem of spectrum resources, Dr. Joseph Mitola proposed the concept of Cognitive Radio (CR) [8]. After several years of radio technology development, the current CR technology mainly includes spectrum sensing, spectrum decision, spectrum shifting and spectrum sharing [9–11]. Through the above four modules, CR technology has largely improved the spectrum resources problems. According to the data of "Cognitive Radio Market Report", CR technology can effectively improve spectrum utilization, reaching 3–10% [12]. Spectrum sensing technology is an indispensable part of the four modules, which realizes the dynamic access of the spectrum by sensing the holes in the spectrum environment. However, in the process of spectrum sensing, in order to obtain more spectrum resources, the range of spectrum sensing is often increased, but the corresponding perception time will be extended, and a lot of calculations and more and more complex algorithm processes will be generated. In order to solve such problems, Haykin first proposed spectrum prediction technology based on spectrum sensing technology in 2005, and then around 2012, researchers strengthened the concept of spectrum prediction technology. First of all, spectrum prediction technology can find the relationship between the spectrum through the study of the historical spectrum segment, predict the future spectrum state, find a range of possible free frequency bands, and then carry out spectrum perception of this range, which can directly find the free spectrum resources in a short time, so as to solve the problem of high spectrum perception consumption. Secondly, when the received signal is incomplete, we can also predict the amplitude and trend of the complete signal according to the spectrum prediction technology and carry out information demodulation. Finally, it allows users to access the underutilized spectrum opportunistically in the time domain or spectrum domain without significantly affecting other users, which provides the possibility for subsequent spectrum decision making, spectrum migration and spectrum sharing, thus assisting cognitive radio technology.

In the current research, the prediction results of spectrum prediction technology are relatively single, which can only complete the prediction in the ordinary time dimension but cannot reflect the characteristics of the frequency band at that time. The existing spectrum prediction processes can be roughly divided into two categories [13], namely, by predicting the quality of the channel and predicting from the perspective of channel occupancy. The current common spectrum prediction technology methods are the regression analysis Model and Hidden Markov Model (HMM). The model based on HMM is simpler and widely used in CR. This kind of method has better accuracy in predicting spectrum occupancy even in complex industrial environments [14]. However, the traditional HMM spectrum prediction algorithm has the problems of time extension, and the prediction accuracy is easily affected by the uncertainty of the matrix to be tested. For this reason, [15] proposes a spectrum occupancy prediction method based on a segmented Markov model, which uses a density clustering density algorithm to predict channel states in clusters. This algorithm not only improves the prediction accuracy but also reduces the prediction uncertainty. The piecewise prediction method of the periodic channel can better describe the change characteristics of historical data in the periodic channel and improve the final prediction accuracy without

changing the algorithm complexity [16]. The statistical learning method is used to learn the historical data of spectrum perception. Under the condition of ensuring the prediction rate, based on HMM, the optimal frequency of shortwave cognition is found to accurately predict the channel state of shortwave users [17]. In order to solve the shortcomings of the existing channel access technology in the shortwave ALE system, the three-state HMM is proposed to be used in spectrum prediction technology in [18], which greatly improves the prediction accuracy and utilization rate in the shortwave ALE system. Due to the nature of these two algorithms in mathematical calculation, although it will be convenient to understand the prediction process, in general, in the spectrum occupancy prediction method based on HMM, the system consumption is relatively large. Furthermore, the regression method is highly complex, and is not suitable for continuous prediction. In addition, the idea of cooperative prediction can also be applied to prediction technology. For the energy-constrained cognitive radio network, the combination of the cooperative prediction model and the spectrum sensing framework can overcome the problems of the local prediction model, so as to improve the spectrum efficiency in the case of low energy consumption [19].

At present, these prediction techniques solve the spectrum problem by developing complex algorithms, which often consume a lot of time and energy in the prediction, which hinders the success of dynamic spectrum access. Therefore, we urgently need flexible and intelligent spectrum prediction technology. The emergence of machine learning brings new hope to many fields, among which deep learning has shown good ability in the classification and prediction of some complex systems, giving researchers the idea of applying deep learning to spectrum prediction technology.

In [20], a spectrum prediction technology based on Multilayer Perceptron (MLP) is proposed. The structural parameters of the artificial neural network are configured, and all parameters are randomly initialized and then iteratively trained. The gradient is constantly calculated, and the parameter values are updated until the iteration conditions are met. However, due to the structure of the MLP network itself, the learning speed is slow, and it is easy to fall into the situation of local extremum. The complex nonlinear processing ability of feedforward neural networks can greatly improve the learning speed and prediction accuracy in prediction, among which the BP network [21] and its variants are common. Later, researchers proposed Recurrent Neural Networks (RNN) on the basis of BP, which were constructed by sliding windows to analyze spectrum data in a specified time. However, RNN is prone to the problem of gradient disappearance in spectrum prediction, so a spectrum prediction method based on the Deep Recurrent Neural Network (DRRN) [22] and a series of improved network algorithms are proposed. In [23], spectrum prediction techniques based on the Long Short-term Memory network (LSTM) and BP network were discussed. In [24], a new spectrum prediction framework was developed for LSTM, which improved its processing of time series. In [25], prediction errors of LSTM were compared with those of integrated moving average autoregression and delay neural networks, which proved that LSTM has certain advantages and good prediction performance in time series problems. In order to solve the long-term predictions of spectrum data, ref. [26] proposes a convolutional LSTM model for spectrum prediction—ConvLSTM. In the 450–520 MHz frequency band, the long-term spatial-spectral-time joint prediction of the spectrum signal is carried out. The results show that the model exhibits a stable value of Root Mean Squared Error (RMSE) for different channels, but it rises with the increase in the time step. In [27], STS-PredNet was proposed by combining ConvLSTM and the predictive recurrent neural network (PredRNN). This model shows stable RMSE performance in multi-time step spatial-temporal spectrum prediction, and effectively improves the prediction performance in different prediction ranges. For the ISM band with high burst characteristics, ref. [28] proposed an algorithm named Classified-based Deep Reinforcement Learning (C-DRL) to quickly and efficiently complete the matching between the state and the prediction model, achieving the effect of fast convergence, simple operation and high prediction rate. In order to reduce the time cost of convolutional

network prediction in multi-channel, Gao et al. use LSTM, Seq-to-Seq modeling, exploit the intrinsic correlation and time correlation between channels, and propose a multi-channel multi-step joint spectrum prediction algorithm, which can effectively improve the prediction accuracy of multi-channel [29].

In addition, for different industrial frequency bands and different communication systems, researchers have also proposed other prediction algorithms based on LSTM networks, which contribute greatly to the research of spectrum prediction technology [30–34]. Despite the high performance of LSTM, it may bring challenges in terms of computational burden and handling missing data, and since spectral data is equivalent to long time series and LSTM has reduced prediction accuracy for long sequences, we propose the B-LTF algorithm to improve the problem of long sequence prediction. It slows down the impact of sequence length change on the prediction performance of the network so that the overall spectrum prediction rate is improved. Secondly, in this paper, we not only predict the channel state but also predict the spectrum signal. The prediction of the channel state is first processed by the gate threshold and then predicted, so as to know the channel state of the node in the future time, and the signal spectrum prediction goes deeper into the deeper level, that is, the trend and amplitude change of the future signal can be known through the sparse spectrum signal data, which is of great help to the subsequent cognitive radio technology.

Since deep learning algorithms are typically exploratory, they do not have any preconditions or assumptions about the data. Therefore, in many cases, they provide higher accuracy than traditional probabilistic and statistical algorithms. Although deep learning has been very popular in recent years, it is still in the initial stage in the field of communication, especially in CR technology. Therefore, how to use deep learning algorithms to make spectrum prediction more efficient and effective is an emerging research direction. Our main contributions in this paper are summarized as follows:

- We study the problem of spectrum availability prediction and discuss the time spectrum occupancy characteristics together.
- We proposed the B-LTF algorithm, combined the BP network with LSTM, built a new network structure and realized the spectrum prediction from the neural network. We studied the influence of the sequence length of spectrum data and the model prediction rate in detail and effectively improved the accuracy of spectrum prediction.
- We show that the analysis of simulation prediction values obtained by simulating the current channel state shows that a long short-term memory network and its improved model can effectively process the time series, the GRU model has a simpler structure and the training time of the GRU model will be significantly reduced compared with the LSTM network, and the improved B-LTF algorithm compared with the LSTM, BP and GRU has better predictive performance. In addition, when the sequence length of spectrum data increases, the model prediction rate tends to be saturated or reduced.

The rest of this paper is organized as follows. We first introduce the spectrum prediction models, deep learning models and related knowledge in Section 2. Section 3 briefly describes the designed B-LTF model for spectrum prediction, which is followed by simulation results and discussions in Section 4. Finally, we conclude the paper in Section 5.

2. System Model

2.1. Spectrum Prediction Model

In CRN, spectrum sensing techniques work by sensing the frequency band of interest in time slot units, where the time length of each time slot is T_s. The sensed signal strength $\mu(t)$ of each time slot is compared with a gate threshold λ, and whether this frequency band is occupied is determined by the following decision rule.

$$\gamma(t) = \begin{cases} 0 & \mu(t) < \lambda \\ 1 & \mu(t) \geq \lambda \end{cases} \tag{1}$$

The number "1" indicates that the frequency band is occupied by the PU, and "0" indicates that the frequency band is not occupied. As shown in Figure 1, red indicates that the frequency band is occupied, and white indicates that the frequency band is idle:

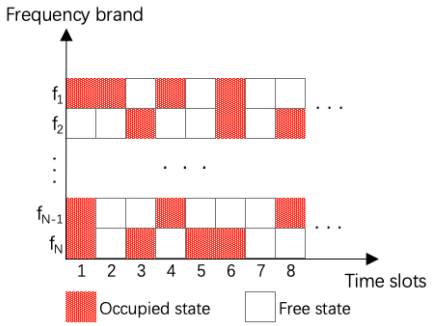


Figure 1. Analog frequency band usage.

After sensing the frequency band, the spectrum prediction technology uses the time-dependent spectrum point data for prediction and predicts the spectrum state of the next time slot by mining the internal relationship between the spectrum states of the historical time slot. In other words, assuming that there are T historical time slots of spectral data, through X_{t-T+1}, X_{t-T+2}, $\cdots\cdots$, X_t performs network training to find internal connections so as to obtain the predicted value of X_{t+1} for time slot $t + 1$. The detailed prediction model is shown in Figure 2.

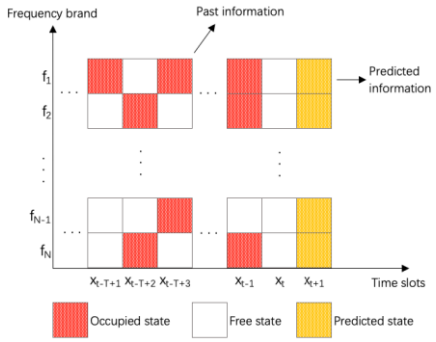


Figure 2. An example of a spectrum prediction model.

2.2. Deep Learning Model

In the current research of radio technology, the neural network is regarded as one of the effective methods to solve spectrum problems. It usually consists of an input layer, one or more hidden layers, and an output layer. By changing the number of hidden layers and the number of neurons in each layer, different complex nonlinear functions can be constructed. The current network types are mainly divided into feedforward neural networks and feedback neural network. The following mainly introduces the improved models of BP network and improvement of LSTM.

2.2.1. Conventional BP Network

In 1986, the research team led by Rumelhart and MeCelland first proposed the BP neural network, which trains the feedforward network according to the error backpropagation algorithm and is one of the common models in the current neural network model. In terms of network structure, the model structure of the BP neural network is relatively simple and is mainly composed of three parts: input layer, hidden layer and output layer. Figure 3 shows the structure of a single-layer BP neural network.

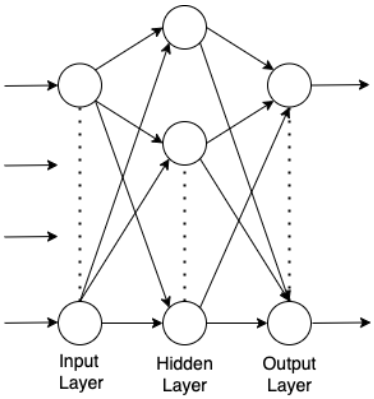


Figure 3. Single-layer BP neural network architecture.

The BP-neural network algorithm is as follows (Algorithm 1):

Algorithm 1. BP-neural network algorithm

1. Get the training data set XTrain and test data set XTest.
2. Set the structure parameters of BP neural network model.
3. Input XTrain, by passing forward: from input layer to hidden layer to output layer, get X_Train.
4. compare the XTrain with the YTrain to obtain the prediction error.
5. When the prediction error e > eth is satisfied and the number of iterations Nit < n, the error backward propagation process is performed to update the weights and then go back to 1, otherwise to 6.
6. When e < eth or Nit = n, the training is finished and obtain the trained network model.
7. Put the XTest into the trained BP neural network to obtain YTest.

where YTrain is the expected output, eth is the specified error size, and n is the set number of training sessions.

For the prediction model design of the BP network, we set three network layers: an input layer, an output layer and a hidden layer. Since a single hidden layer is enough to realize the nonlinear mapping relationship of any dimension, there is no need to increase the number of hidden layers but increase the training time. In addition, the number of nodes in the hidden layer is set to 288, and the number of neurons in the input layer and output layer is set to 1. It is specified that the maximum number of trainings is 250, the learning rate is 0.005, and the target error accuracy of training is 0.00001. The detailed BP network structure diagram is shown in Figure 4.

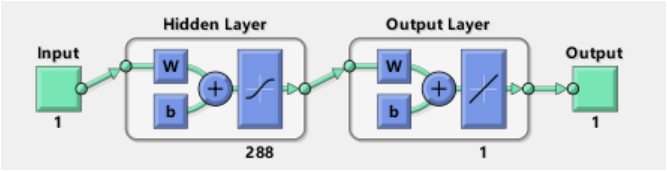


Figure 4. Designed BP network model.

2.2.2. Emerging LSTM Network

In order to solve the problems such as gradient disappearance in RNN, an LSTM network is designed by introducing a threshold mechanism, namely the memory unit [35]. Figure 5 shows the structure of the memory cell. It is mainly composed of memory cells and three gates: the input gate, forget gate and output gate.

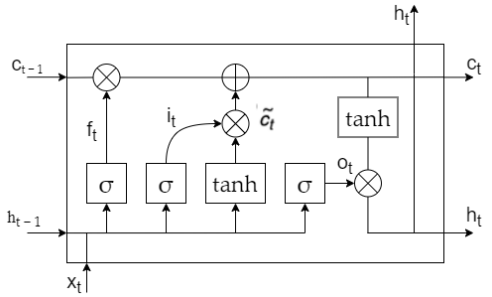


Figure 5. The memory cell structure of an LSTM network.

The LSTM network model designed in this paper will consist of five layers, first an input layer, then an LSTM layer, followed by a forgetting layer, immediately followed by a fully connected layer, and finally an output layer. The number of nodes in the LSTM layer is set to 1024, the activation function of the hidden layer uses tanh activation function, the initialization uses orthogonal initialization, the activation function of the full connection layer is set to the '*softmax*' activation function, and the forgetting rate is set to 0.2 in the forgetting layer. The optimizer of the whole LSTM network uses Adaptive momentum (Adams) algorithm optimizer, and the number of training sessions is set to 200–300. The detailed network structure diagram is shown in Figure 6. At present, the LSTM layer is set to one layer, but the number of LSTM layers may be increased in the subsequent simulation in order to explore the influencing factors of the prediction performance.

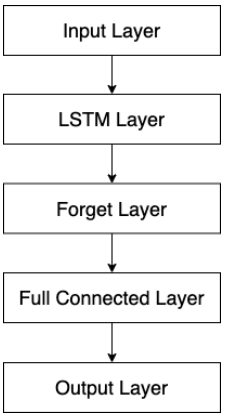


Figure 6. The designed LSTM network model.

2.2.3. GRU Network Model

GRU neural network is a popular neural network in recent years, which is a variant of the LSTM network. Compared with the three gate structures of the LSTM network, the network structure of GRU is simpler, with only two gate structures: the update gate and reset gate and the specific network structure is shown in Figure 7.

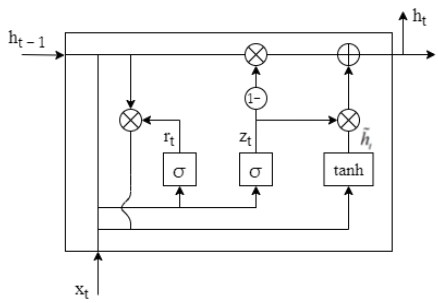


Figure 7. GRU network unit model.

The reset gate and update gate in the GRU network act as the function of updating and filtering the input signal [36]. The reset gate "r" controls the amount of the previous state signal flowing into the current state, and the update gate "z" controls the degree of the current state when the previous state signal flows into the current state. The specific mathematical formula is shown below.

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{2}$$

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{3}$$

$$\widetilde{h}_t = \tanh\left(W_{\widetilde{h}} \cdot [r_t \otimes h_{t-1}, x_t]\right) \tag{4}$$

$$h_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes \widetilde{h}_t \tag{5}$$

where σ is the *sigmoid* function, tanh is the hyperbolic tangent activation function, and W is the weight matrix of the corresponding module. Equation (2) shows that when time step is t, the input vector is the result of linear transformation obtained by multiplying x_t and the state h_{t-1} saved at the previous time step $t-1$ with the weight matrix, respectively. The reset gate uses an activation function to convert the sum of two results into [0,1]. Equation (3) is the relevant formula of the update gate, which is similar to the expression of the reset gate. Equation (4) is expressed as the memory content in the current cell, and the current information and historical information are added and transformed by the tanh function. Equation (5) represents the final output of the memory unit.

From the structural model and calculation formula of the GRU network, it can be seen that the GRU network, such as the LSTM network, memorizes the information of historical time steps and will not be eliminated with the increase in time steps, but combines the historical state information with the current state information, and then passes it to the subsequent cells. However, compared with the LSTM network, when solving consistent types of problems, because the structure of GRU is simpler, the time required for the whole training will be greatly reduced, and the final results are similar to the results of the LSTM network.

The GRU network is also similar to the LSTM network model in model design. Firstly, a GRU layer is set, the number of nodes in the network layer is set to 1024, the tanh function is also used as the activation function, the optimizer chooses Adams, the number of training sessions is set to 200–300, and a fully connected layer follows the GRU layer. There is also an input layer and an output layer, and the complete GRU network model is shown in Figure 8.

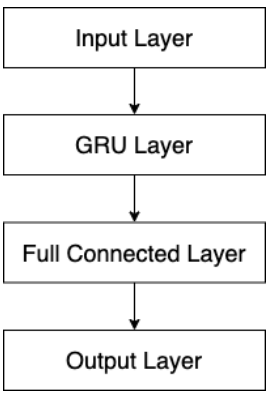


Figure 8. The designed GRU network model.

3. B-LTF Model

3.1. Related Theories and Formulas

Compared with the traditional HMM model, BP neural network can store a large number of specific connections between input and output, obtain the estimation error

of all levels through reverse transmission, and then use the steepest descent method to continuously update the weights and thresholds of the whole network model to achieve the purpose of minimizing the error. The process is mainly divided into two stages. The first stage is the forward propagation of the signal from the input layer through the hidden layer and finally to the output layer. The second stage is the back propagation of the error, from the output layer to the hidden layer, and finally to the input layer, adjusting the weights and biases from the hidden layer to the output layer in turn, and the weights and biases from the input layer to the hidden layer. The detailed mathematical formula is as follows:

Hidden layer output:

$$H_j = g\left(\sum_{i=1}^{n} \omega_{ij} x_i + a_j\right) \tag{6}$$

Output from the output layer:

$$O_k = \sum_{j=1}^{l} H_j \omega_{jk} + b_k \tag{7}$$

Error formula:

$$E = \frac{1}{2}\sum_{k=1}^{m} (Y_k - O_k)^2 \tag{8}$$

Weight update formula:

$$\begin{cases} \omega_{ij} = \omega_{ij} + \eta H_j (1 - H_j) x_i \sum_{k=1}^{m} \omega_{jk} e_k \\ \omega_{jk} = \omega_{jk} + \eta H_j e_k \end{cases} \tag{9}$$

Bias update formula:

$$\begin{cases} a_j = a_j + \eta H_j (1 - H_j) \sum_{k=1}^{m} \omega_{jk} e_k \\ b_k = b_k + \eta e_k \end{cases} \tag{10}$$

The number of nodes in the input layer is n, the number of nodes in the hidden layer is l, and the number of nodes in the output layer is m. The weights ω_{ij} from the input layer to the hidden layer, the weights ω_{jk} from the hidden layer to the output layer, the bias a_j from the input layer to the hidden layer, and the bias b_k from the hidden layer to the output layer. The learning rate is η and the activation function is $g(x)$. The activation function is taken as a *Sigmoid* function.

The LSTM network can perform selective memory of historical information due to the existence of three gates and can better deal with time series problems. The input gate i controls how much information goes into the memory cell, the forget gate f determines how much state information of previous cells is remembered, the output gate o controls the amount of information output by calculating the storage size of the memory cell, and the most critical memory cell gives the LSTM the ability to freely choose the content of memory in each time step. The mathematical model of the network is as follows:

$$i_t = \sigma\left(W^i x_t + U^i h_{t-1} + b_i\right) \tag{11}$$

$$f_t = \sigma\left(W^f x_t + U^f h_{t-1} + b_f\right) \tag{12}$$

$$o_t = \sigma(W^o x_t + U^o h_{t-1} + b_o) \tag{13}$$

$$\widetilde{c}_t = \tanh(W^c x_t + U^c h_{t-1} + b_c) \tag{14}$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \widetilde{c}_t \tag{15}$$

$$h_t = o_t \otimes \tanh(c_t) \tag{16}$$

where x_t is the input vector at time step t, the hidden layer state vector is h_t, c_t is the memory cell, σ is the sigmoid function, tanh is the hyperbolic tangent activation function, W and U are the weight matrix of the corresponding module, and b is the bias vector of the corresponding template. Equations (11) and (14) are related calculations for input gates, Equation (15) is the cell state equation at time 't', Equation (12) is related calculations for the forget gate, Equations (13) and (16) are related calculations for the output gate.

3.2. Model Introduction

This paper mainly focuses on the prediction and research of spectrum signals. Since spectrum signals are equivalent to a long time series, their prediction is a series-to-sequence prediction. Therefore, according to this situation, we proposed the B-LTF algorithm, which is mainly a prediction model composed of the BP neural network and LSTM network. Among them, the BP neural network is not only simple in structure but also has strong nonlinear mapping ability. However, it does not consider the time correlation between the data, which often has a big problem for the prediction of time series. When facing the same data set, it will still wait for the same result after transformation. In addition, some traditional neural networks are prone to problems such as the disappearance of the gradient in the prediction, but the LSTM network changes this situation. It can solve the gradient problem in the prediction, and because of the existence of gates, the LSTM neural network has better performance in dealing with time series problems. Therefore, combining the advantages of the BP neural network and the LSTM neural network, the construction of the BP-LSTM combined network prediction model can not only solve the deficiencies of the BP model but also improve the convergence speed of the overall model, so as to improve the prediction rate of the model.

Before training, the data set was classified into the training set and prediction set. In the training, the training data is first preprocessed and then input into the BP neural network model and LSTM neural network model, respectively. After the BP neural network and LSTM neural network training, two trained network models are obtained, as shown in Figure 9 In the process of prediction, the data is first input into the BP network model to obtain the output result y_1, and then y_1 is input into the LSTM network model to obtain the output data y_2, and finally, y_1 and y_2 are weighted by the weight ratio to obtain the output data y_3, that is, the final prediction result. The complete process of prediction algorithm (Algorithm 2) and prediction model is as follows:

Algorithm 2. B-LTF algorithm

Input: Historical spectrum data $\{X_1, X_2, \ldots, X_t\}$, Time dimension T;
Output: The spectrum prediction results;
//Construct the dataset;
1. The spectrum data is preprocessed according to the threshold value, so as to turn the data into a sequence composed of [0,1] "S_p" represents the frequency band state
2. $\boldsymbol{\mathcal{D}} \leftarrow \emptyset$
3.Put the sample S_p into $\boldsymbol{\mathcal{D}}$
4. A segment is randomly selected in $\boldsymbol{\mathcal{D}}$ and divided into $\boldsymbol{\mathcal{D}}_{\text{Train}}$ and $\boldsymbol{\mathcal{D}}_{\text{Test}}$ according to the ratio of training set to test set
//Train the model;
5. Initialize all learnable parameters W, b in BP-LSTM
6. **Repeat**
7. Randomly select a batch of instances from $\boldsymbol{\mathcal{D}}_{\text{Train}}$
8. Update W, b in the network
9. **Until** the training epochs are met
//Test the model;
10.Put $\boldsymbol{\mathcal{D}}_{\text{Test}}$ into the trained model
11. Output the prediction results

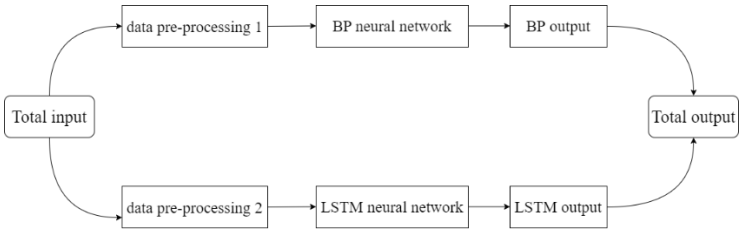


Figure 9. B-LTF training flow.

For the B-LTF network model, we set the training period epochs = 250, the optimizer adopts Adam. For B-LTF network parameters, it is divided into the BP layer and LSTM layer, the number of neurons is 288 and 1024, and the neuron loss rate is 0.2. In addition, different from the prediction using BP neural network alone, the output dimension of the BP layer needs to be consistent with the input dimension of the LSTM layer. The detailed model framework is shown in Figure 10.

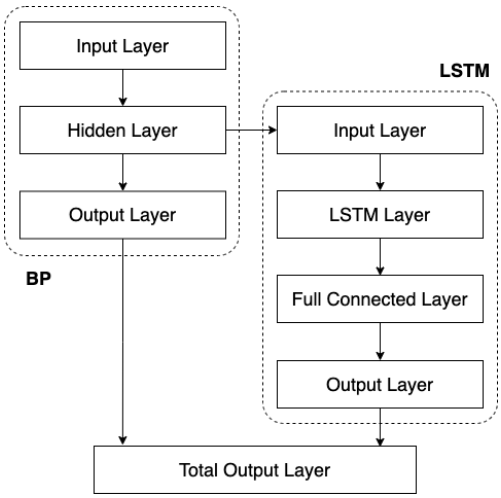


Figure 10. The designed B-LTF network model.

4. Experiment Evaluation

In this paper, we use the B-LTF network model and three common neural networks to make two kinds of simulation prediction. The first one is to compare the accuracy of each neural network for the future channel state prediction, change the respective network parameters, and explore the factors affecting the prediction degree. The second is sequence-to-sequence prediction, which predicts the trend and amplitude of the spectrum signal in the future period through the known spectrum signal.

4.1. Channel State Prediction

In this paper, the spectrum data are from the use of the high-performance spectrum analyzer AgilentE4440A by RWTH Aachen University in Germany on the wireless spectrum of a school in Maastricht, Netherlands. The measurement band range is mainly 20 MHz to 1.52 GHz and 1.5 GHz to 3 GHz. The whole measurement band is scanned each time, and the scanning time is 1.8 s on average. The measurement resolution bandwidth is 200 KHz. Because of the huge number of spectrum data measured, it is not conducive to the simulation of all the spectrum data, so we selected the frequency band data from 890 MHz to 915 MHz (GSM900 uplink band) and 935 MHz to 960 MHz (GSM900 downlink band) from the collected spectrum data for processing and data analysis. Then, nine channels from the downlink band of GSM900 are selected as the data of this experiment for the training and prediction of the model.

4.1.1. Data Preprocessing

Before analyzing the spectrum data, it is necessary to preprocess the existing original signal spectrum. There are only two states in the actual spectrum: idle state and occupied state. Usually, a threshold value is set first, and the spectrum states are distinguished by comparing the spectrum value with the threshold value. That is, if the spectrum value of the channel is lower than the threshold at a certain time, the channel is considered to be unused and idle and is represented by "0", while the opposite is represented by "1".

In the research, the determination of the threshold value plays a great role in the experimental results, because when the threshold value is set too large, some useful signals will be missed, but when the threshold value is set too small, noise signals will be mistaken for useful signals. How to select the threshold value is a key step in data processing and analysis. Because there is a large amount of original spectrum data, select some data to display, as shown in Table 1.

Table 1. Partial spectrum data.

Channel \ Slot	1	2	3	4	5	6	7	8	9
1	−97.799	−96.879	−98.058	−101.042	−100.307	−100.696	−100.751	−101.346	−100.867
2	−101.126	−100.244	−100.150	−101.270	−103.100	−102.444	−102.598	−102.805	−102.943
3	−99.644	−100.088	−99.616	−100.602	−100.106	−100.632	−101.950	−101.225	−100.784
4	−99.763	−97.423	−99.049	−98.909	−97.472	−98.264	−100.034	−97.922	−99.077
5	−99.010	−96.495	−99.228	−99.385	−99.082	−98.963	−100.305	−98.338	−98.964
6	−96.176	−97.853	−97.399	−99.4540	−99.284	−98.286	−97.975	−99.013	−100.780

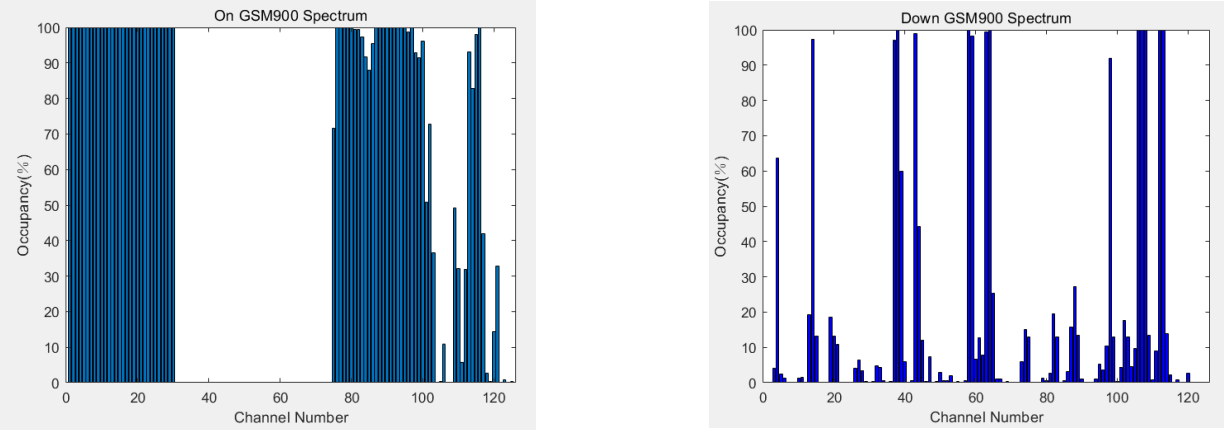
In the data analysis of this experiment, the threshold value is set as −100 dB by referring to the method of researchers' threshold value, and the channel state is distinguished by comparing the spectrum value of −100 dB with the original useful signal. When −100 dB is less than the spectrum value of the signal, it is represented by "1"; when it is greater than the spectrum value, it is represented by "0" to realize the conversion of the spectrum value into the binary representation of 0 and 1, where "0" means that the channel is idle at this moment, and "1" means that the channel is used by PU. So, Table 1 can be converted to Table 2 to indicate the occupied or idle state of the channel.

Table 2. Partially processed data.

Channel \ Slot	1	2	3	4	5	6	7	8	9
1	1	1	1	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	1	0	1	0	0	0	0	0	0
4	1	1	1	1	1	1	0	1	1
5	1	1	1	1	1	1	0	1	1
6	1	1	1	1	1	1	1	1	0

4.1.2. Calculate the Channel Occupancy

According to the processed experimental data, we can calculate the corresponding occupancy of each channel. Here, we define channel occupancy as the number of occupied channels divided by the total number of idle plus occupied channels in the entire channel. Figure 11 shows the channel occupancy in the upstream and downstream bands of the GSM900. The abscissa is the channel number, and the ordinate is the channel occupancy.



(**a**) (**b**)

Figure 11. Occupancy of the channel: (**a**) the channel occupancy in the uplink band of GSM900; (**b**) the channel occupancy in the downlink band of GSM900.

It can be seen that the occupancy of each channel is not high, and even the occupancy of some channels is 0. The results show that the corresponding spectrum utilization is not very ideal, and there is a lot of space to improve the utilization. In order to calculate the occupancy of each channel more intuitively, we conducted further statistical analysis on the GSM900 uplink band and GSM900 downlink band, respectively. The proportion of the number of channels with occupancy less than O in each band to the total number of channels in this band is shown in Table 3, where $O \in [10\%, 100\%]$.

Table 3. Occupancy of the GSM900 channel.

Occupancy / Frequency Band	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Uplink band of GSM900	0.44	0.456	0.456	0.488	0.504	0.512	0.512	0.528	0.544	1
Downlink band of GSM900	0.696	0.848	0.864	0.864	0.872	0.88	0.888	0.888	0.888	1

As can be seen from the data in Table 3, the channel occupancy in the downlink band of GSM900 is not ideal, with nearly 85% of the channels occupying less than 20%, and nearly half of the channels in the uplink band occupying less than 40%. There are a large number of unused or underused channels in the GSM900 band, which wastes spectrum resources to a great extent, but also has more space to use. Therefore, we can start from the low-occupied frequency band and improve the channel occupancy rate and spectrum utilization by spectrum prediction technology.

4.1.3. Evaluation Criterion

In order to make the network model designed in this paper able to evaluate the prediction more realistically, the following evaluation indicators are defined:

Root Mean Squared Error (RMSE) is the square root of the ratio between the sum of the squared differences between the predicted value (x_i) and the true value (y_i) and the number of predictions (n), which is calculated as follows.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2} \tag{17}$$

It is a common metric to evaluate the performance of response prediction. The lower the RMSE, the smaller the error between the predicted value and the true value of the model, and the better the model is.

Mean Absolute Error (MAE) is the ratio between the sum of the absolute errors of the predicted value and the actual value and the number of predictions. The specific formula is as follows. MAE can better reflect the actual situation of the predicted value error.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \tag{18}$$

The prediction accuracy is the ratio of the number of successful predictions to the total number of predictions, which gives a very intuitive indication of the accuracy and performance of the model prediction, and is denoted by "Acc".

$$\text{Acc} = \frac{\text{Predict the right amount}}{\text{Total number of predictions}} \tag{19}$$

4.1.4. Comparison of Simulation Results

We used the processed data in Section 4.1.1 for experiments to study the relationship between the total sequence length and the model prediction rate. In this experiment, by changing the length of the sequence, the prediction effects of the four models under different sequence lengths are obtained, and the performance of the model prediction is judged by comparing the prediction accuracy of the spectrum signal prediction of these models. A detailed simulation of the prediction rate is shown in Figure 12.

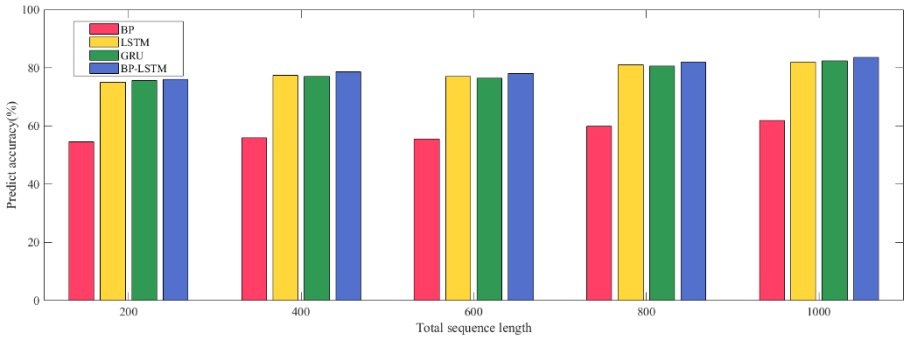


Figure 12. Total sequence length vs. prediction rate.

As can be seen from Figure 13, when the sequence length changes, the training results of different models are different, resulting in different prediction effects. In general, the B-LTF model has the best prediction effect. It can be seen that when the sequence length is 1000, the prediction accuracy is more than 80%, and the prediction effect of the LSTM model and GRU model is not much different. The BP model is lower, but the prediction rate is also about 60%. At the same time, for the same model, the prediction accuracy shows an upward trend with the increase in the total sequence length. This is due to the increase in historical information, the model can better learn and find the relationship between each point, thereby improving the prediction accuracy. However, it can be found that the growth trend will gradually become smaller, so whether the final prediction accuracy will saturate when the sequence is long enough needs to be tested with more data sets. In order to analyze and compare the prediction effect of each model more intuitively, the predicted values of the four models are compared with the true values in Figure 13. In the four simulation diagrams, the horizontal coordinate is the length of the sequence, the vertical coordinate is the channel state value, the blue line represents the current real channel state, and the red line represents the channel state predicted by the four models. The more the red overlaps with blue, the higher the prediction accuracy of the model.

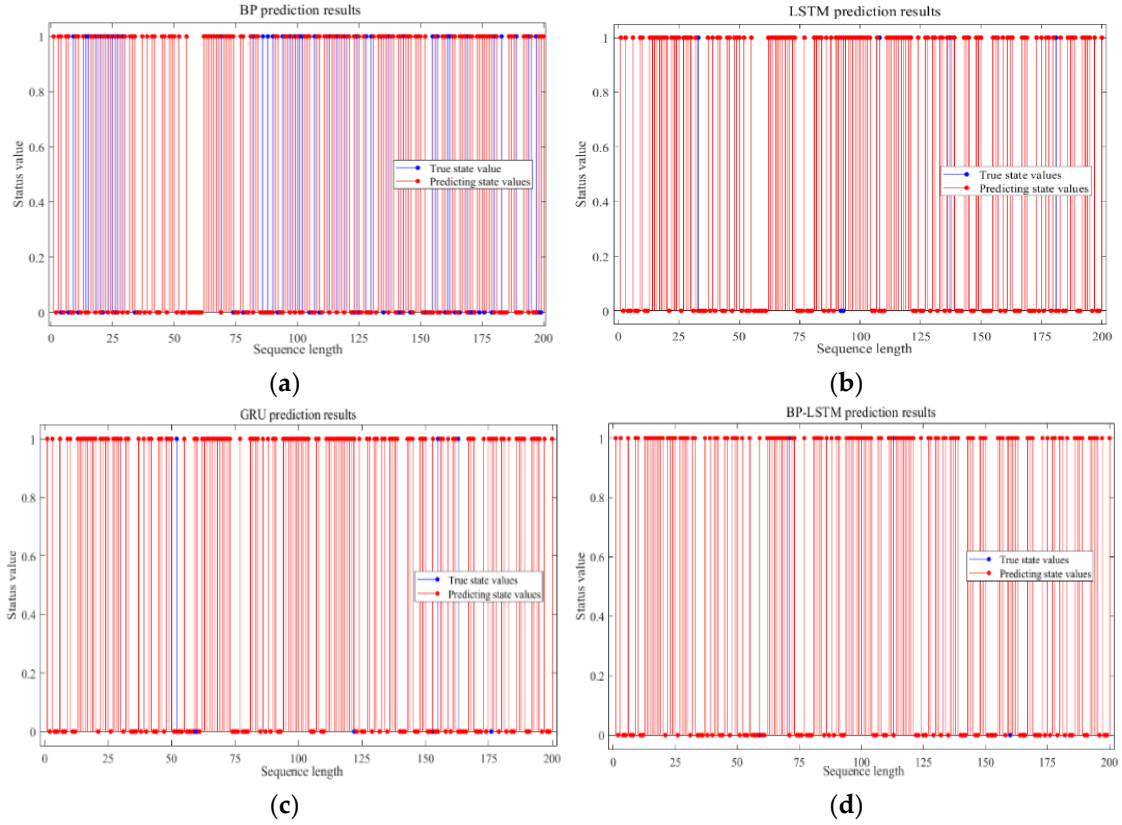


Figure 13. Plot of predicted vs. true values for the four models: (**a**) BP prediction results; (**b**) LSTM prediction results; (**c**) GRU prediction results; (**d**) B-LTF prediction results.

In addition, for the LSTM model, GRU model and B-LTF model, the size of the sliding window is changed to compare the prediction effects of the three models, as shown in Figure 14.

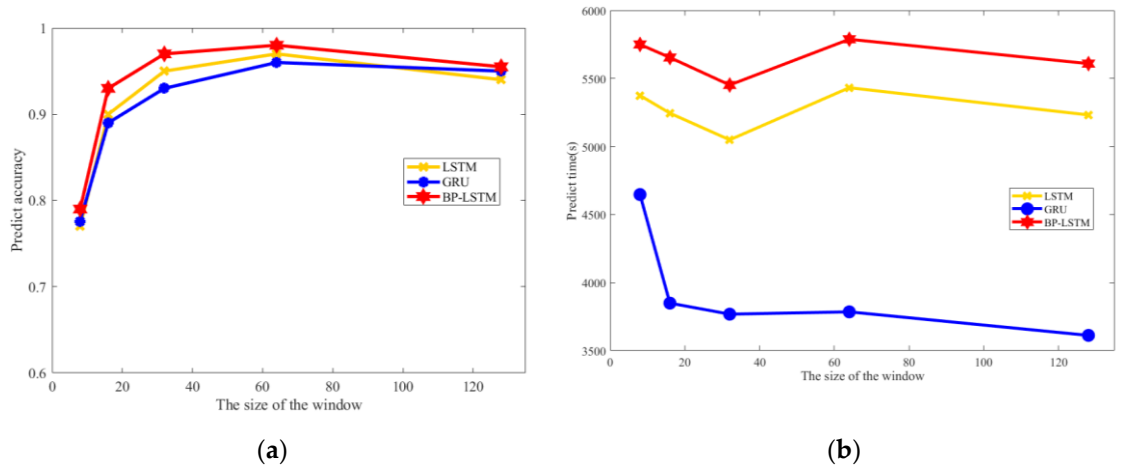


Figure 14. Window size variation: (**a**) the prediction accuracy of the three models at different window sizes; (**b**) the corresponding training times.

It can be seen from (a) that with the increase in window size, the prediction accuracy of the three network models shows an upward trend, which is more obvious from 8 to 16, and the upward trend is moderated after 16. The prediction accuracy of the three network models is excellent, and the performance of B-LTF is better. When the window size is 64, the prediction rate reaches about 98%, which is about 1% higher than that of LSTM and

GRU models, and the performance of LSTM and GRU is similar. In addition, it can also be seen that when the window size is changed again, the prediction accuracy is saturated, no longer changes, and may even decrease a little, which is caused by the structure of the LSTM network model itself. From (b), it can be seen that the overall training is often relatively long. GRU, as the "brother" of LSTM, in the case of the same complexity, because of its simpler structure, the operation speed is also faster. B-LTF, as a combination model, takes the highest operation time among the three models. When the window size is above 32 and the number of training rounds reaches about 120, the value of loss almost no longer changes and always maintains a value. Therefore, in actual prediction, the number of training rounds can be appropriately reduced to reduce the operation time of the model.

4.2. Prediction of the Trend of the Spectrum Signal

For the spectrum signal, after processing according to the threshold value, we can know the channel usage of the current time slot, so as to predict the future state. In addition, it can also directly predict the size and trend of the future spectrum signal through the internal relationship between the current spectrum value of the spectrum signal. The channel state only predicts whether the channel is occupied, while signal spectrum prediction goes deeper and can predict the signal amplitude. The purpose of the two is not the same. If it is accessible, that is, as long as the idle channel is found, and the signal spectrum prediction can be used to complete the signal spectrum, that is, to complete the signal spectrum with a small amount of sparse data and prediction. Since the spectrum signal can be regarded as a time series, the neural network can also be used for sequence-to-sequence prediction.

4.2.1. Data Processing and Parameter Setting

The data of this simulation experiment is also used in RWTH Aachen University in Germany mentioned in Section 4.1 because the power amplitude trend of the future spectrum is predicted, so there is no need for excessive data preprocessing.

As for the data set, it is first divided into the training set and test set. Here, the first 80% is still selected as the training set, and the last 20% is used as the prediction set. The time series is predicted by the network model designed in Sections 2.2 and 3.2. Therefore, instead of comparing the prediction accuracy, the RMSE and MAE of these models for spectrum signal prediction are compared.

4.2.2. Experimental Result

Figure 15 shows the comparison between the predicted value of the spectrum and the real value of the spectrum of the above four models. It can be seen intuitively that the predicted value of the B-LTF model is closer to the trend of the real value. In addition, under the same spectrum data and the same number of hidden layers, the RMSE and MAE values of the four models are shown in Table 4.

From the prediction results and corresponding values of the above four models, it can be seen that the prediction results of the BP model are too jumping and not very stable, resulting in a small MAE value but a large RMSE value. However, the prediction results of the other three models are relatively stable, although there are errors with the true value, but the difference changes little. It also proves that the prediction performance of the BP network model for spectrum trend still needs to be improved, compared with the trend prediction of the other three models are better. Among them, the B−LTF network model has the best prediction effect, and the error between the prediction result and the true value is the smallest, which also verifies that its evaluation index is better than the other three models.

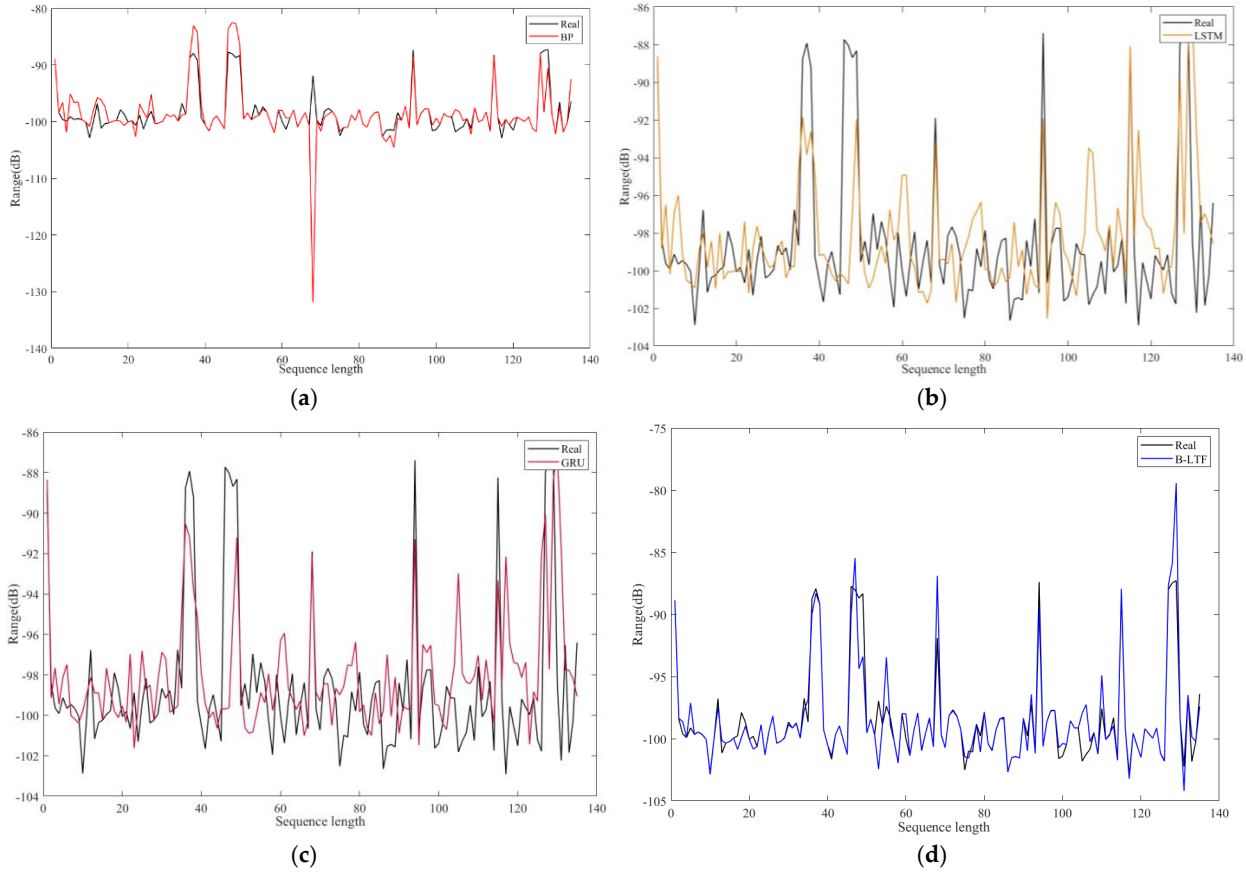


Figure 15. Plot of actual vs. predicted values: (**a**) true value vs. BP model prediction results; (**b**) true value vs. LSTM model prediction results; (**c**) true value vs. GRU model prediction results; (**d**) true value vs. B−LTF model prediction results.

Table 4. Comparison of RMSE and MAE values of the four models.

Model \ Evaluation Index	RMSE	MAE
BP	3.927	1.6417
LSTM	3.2344	2.1965
GRU	3.3741	2.3031
B-LTF	1.6746	0.6309

For the prediction of channel state, the influence of different sequence lengths and window sizes on the prediction performance is studied. The results show that with the increase in sequence length and window size, the prediction performance of the four models is on the rise. In addition, according to the analysis of the above four models, the LSTM model and its improved model have better prediction performance than the BP neural network. The B-LTF has better prediction performance but is replaced by a longer training time. For the trend prediction of spectrum prediction, the experimental results show that the improved recurrent neural network has certain advantages in processing time series, and the prediction effects of LSTM and GRU models are good. The overall prediction effect of the B-LTF model is closer to the true value. There will be some big jumps in the spectrum values, making the final prediction not very close to the true value. In general, although the prediction performance of these models is ideal, and the B-LTF model improves the prediction performance of the original LSTM model again, there is a long training process, and it can be found that the increase in sequence length will reduce the prediction performance of the above models.

5. Conclusions

In this paper, through the processing and analysis of an actual spectrum observation data set, we find that the spectrum utilization rate of the GSM900 band is not high, and there is a large waste of resources, which proves that the spectrum in today's world has a major problem. Based on this, we propose the B-LTF algorithm to predict the spectrum using neural networks. In it, we analyze the performance of four network models, namely the B-LTF model, GRU model, LSTM model and BP model, in predicting the channel occupancy status and predicting the future spectrum trend in detail. From the overall simulation results, the prediction effect of the long short-term memory network and its improved model is better than that of the BP network model, which also proves that they have a strong ability to deal with time series. When predicting the channel occupancy state, the B-LTF model improves the prediction accuracy by about 1% to 2% compared with the LSTM and GRU models. In the spectrum signal trend prediction, the B-LTF model can predict the overall trend and amplitude change of the future spectrum signal more effectively, but the sequence length has a great influence on the prediction results in the overall simulation and the operation time is too long. Therefore, how to reduce the operation time and reduce the impact of sequence length on the prediction results while improving the spectrum prediction performance are the main issues worth exploring in future research.

Author Contributions: Q.C. and X.W. put forward the idea of this paper. Q.C. finished the design of the study and the algorithms. Q.C. and X.W. contributed to the experimental work and the data analysis. Q.C. and X.Y. made figures and tables. Q.C. and X.W. drafted the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Heilongjiang Postdoctoral Foundation, grant number LBH-Z19168 and Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kolodzy, P. *Spectrum Policy Task Force Report*; FCC ET Docket No. 02-155; Federal Communications Commission: Washington, DC, USA, 2002; Volume 40, pp. 147–158.
2. Kiftaro, F.; El-Tarhuni, M.; Assaleh, K. UHF Spectrum Occupancy Measurements in Sharjah—UAE. In Proceedings of the 2017 9th IEEE-GCC Conference and Exhibition (GCCCE), Manama, Bahrain, 8–11 May 2017.
3. Yu, X.; Kong, M.; Li, X. Spectrum occupancy in the 87.5–108 MHz band: Measurement and analysis. In Proceedings of the 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 25–26 March 2017.
4. Ramirez, D.; Cardenas-Juarez, M.; Pineda-Rico, U. Spectrum Occupancy Measurements in the Sub-6 GHz Band for Smart Spectrum Applications. In Proceedings of the 2018 IEEE 10th Latin-American Conference on Communications (LATINCOM), Guadalajara, Mexico, 14–16 November 2018.
5. Engiz, B.K.; Rajab, Y.A. Investigation of Spectrum Occupancy in GSM Band in Samsun, Turkey. In Proceedings of the 2019 6th International Conference on Electrical and Electronics Engineering (ICEEE), Istanbul, Turkey, 16–17 April 2019.
6. Zulfiqar, M.D.; Ismail, K.; Hassan, N.U. Radio Spectrum Occupancy Measurement from 30MHz-1030MHz in Pakistan. In Proceedings of the 2019 UK/ China Emerging Technologies (UCET), Glasgow, UK, 21–22 August 2019.
7. Tandra, R.; Sahai, A. Fundamental limits on detection in low SNR under noise uncertainty. In Proceedings of the 2005 International Conference on Wireless Networks, Communications and Mobile Computing, Maui, HI, USA, 13–16 June 2005.
8. Mitola, J.; Maguire, G.Q. Cognitive radio: Making software radios more personal. *IEEE Pers. Commun.* **1999**, *6*, 13–18. [CrossRef]
9. Woods, T.; Furman, S. All-domain Spectrum Command and Control via Hierarchical Dynamic Spectrum Sharing with Implemented Dynamic Spectrum Access Toolchain. In Proceedings of the MILCOM 2021–2021 IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 29 November–2 December 2021.
10. Madhavan, A.; Govindarajan, Y. Wideband Spectrum Sensing in Dynamic Spectrum Access Systems Using Bayesian Learning. *J. Phys. Conf. Ser.* **2021**, *1964*, 62067–62075. [CrossRef]

11. Toma, O.H.; Lopez-Benitez, M. Cooperative Spectrum Sensing: A New Approach for Minimum Interference and Maximum Utilisation. In Proceedings of the 2021 IEEE International Conference on Communications Workshops (ICC Workshops), Montreal, QC, Canada, 14–23 June 2021.
12. Solanki, S.; Dehalwar, V. Spectrum Sensing in Cognitive Radio Using CNN-RNN and Transfer Learning. *IEEE Access* **2022**, *10*, 113482–113492. [CrossRef]
13. Huang, P.; Liu, C.J. Wireless Spectrum Occupancy Prediction Based on Partial Periodic Pattern Mining. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1925–1934. [CrossRef]
14. Saad, A.; Staehle, B.; Knorr, R. Spectrum prediction using hidden Markov models for industrial cognitive radio. In Proceedings of the 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), New York, NY, USA, 17–19 October 2016.
15. Wu, J.W.; Li, Y. HMM Cooperative Spectrum Prediction Algorithm Based on Density Clustering. *J. Comput. Appl.* **2018**, *45*, 129–134. [CrossRef]
16. Liu, X.N.; Shi, R. Spectrum Occupancy Prediction Method Based on Segmented Markov Model. *Radio Eng.* **2020**, *50*, 690–695.
17. Wang, D.; Cao, P. High frequency cognitive frequency selection mechanism based on hidden Markov model. *J. Comput. Appl.* **2016**, *36*, 1179–1182, 1187.
18. Lin, G.; Cheng, Y.P. Performance Analysis of Three-State HMM in HF Channel Estimation. *Commun. Technol.* **2016**, *49*, 286–292.
19. Chauhan, P.; Deka, S.K. Cooperative Spectrum Prediction-Driven Sensing for Energy Constrained Cognitive Radio Networks. *IEEE Access* **2021**, *10*, 26107–26118. [CrossRef]
20. Zhao, J.L.; Wang, M.W.; Yuan, J.S. Based on neural network spectrum prediction of cognitive radio. In Proceedings of the 2011 International Conference on Electronics, Communications and Control (ICECC), Ningbo, China, 9–11 September 2011.
21. He, Y.; Meng, Z. A dynamic model of evaluating differential automatic method for solving plane problems based on BP neural network algorithm. *Phys. A Stat. Mech. Its Appl.* **2020**, *556*, 1–11. [CrossRef]
22. Tang, Z.; Li, S. Deep recurrent neural network for multiple time slot frequency spectrum predictions of cognitive radio. *KSII Trans. Internet Inf. Syst.* **2017**, *11*, 3029–3045.
23. Yu, L.; Chen, J.; Ding, G. Spectrum prediction via long short-term memory. In Proceedings of the 2017 3rd IEEE International Conference on Computer and Communications (ICCC), Chengdu, China, 13–16 December 2017.
24. Yu, L.; Chen, J. Spectrum Prediction Based on Taguchi Method in Deep Learning With Long Short-Term Memory. *IEEE Access* **2018**, *6*, 45923–45933. [CrossRef]
25. Chen, A.M.; Zhang, C.Y. UAV cognitive radio spectrum prediction based on long short-term memory network. *Foreign Electron. Meas. Technol.* **2021**, *40*, 37–43.
26. Shawel, B.S.; Woldegebreal, D.H.; Pollin, S. Convolutional LSTM-based Long-Term Spectrum Prediction for Dynamic Spectrum Access. In Proceedings of the 2019 27th European Signal Processing Conference (EUSIPCO), A Coruna, Spain, 2–6 September 2019.
27. Li, X.; Liu, Z.C. Deep Learning for Spectrum Prediction From Spatial-Temporal-Spectral Data. *IEEE Commun. Lett.* **2021**, *25*, 1216–1220. [CrossRef]
28. Yang, C.; Peng, T.; Zuo, P. A Spectrum Prediction Method for Bursty Frequency Bands. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), Nanjing, China, 29 March 2021.
29. Gao, Y.L.; Zhao, C.Y.; Fu, N. Joint multi-channel multi-step spectrum prediction algorithm. In Proceedings of the 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 27–30 September 2021.
30. Huang, Q.Y. Research On Cognitive Radio Spectrum Prediction Based On LSTM Neural Network. Master's Thesis, Harbin Institute of Technology, Harbin, China, June 2020.
31. Radhakrishnan, N.; Kandeepan, S. Performance Analysis of Long Short-Term Memory-Based Markovian Spectrum Prediction. *IEEE Access* **2021**, *9*, 149582–149595. [CrossRef]
32. Aygul, M.A.; Nazzal, M. Efficient Spectrum Occupancy Prediction Exploiting Multidimensional Correlations through Composite 2D-LSTM Models. *Sensors* **2021**, *21*, 135. [CrossRef] [PubMed]
33. Radhakrishnan, N.; Kandeepan, S. An Improved Initialization Method for Fast Learning in Long Short-Term Memory-Based Markovian Spectrum Prediction. *IEEE Trans. Cogn. Commun. Netw.* **2021**, *7*, 729–738. [CrossRef]
34. Radhakrishnan, N.; Kandeepan, S.; Yu, X.H. Soft Fusion based Cooperative Spectrum Prediction using LSTM. In Proceedings of the 15th International Conference on Signal Processing and Communication Systems (ICSPCS), Sydney, Australia, 13–15 December 2021.
35. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
36. Pezeshki, M. Sequence Modeling using Gated Recurrent Neural Networks. *Comput. Sci.* **2015**, *9*, 1–7.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Research on Analog Circuit Soft Fault Diagnosis Method Based on Mathematical Morphology Fractal Dimension

Xinmiao Lu [1,*], Cunfang Yang [1], Qiong Wu [1,2], Jiaxu Wang [1], Zihan Lu [1], Shuai Sun [1], Kaiyi Liu [3] and Dan Shao [4]

[1] School of Measurement-Control Technology and Communications Engineering, Harbin University of Science and Technology, Harbin 150080, China
[2] Heilongjiang Network Space Research Center, Harbin 150090, China
[3] Harbin Meteorological Bureau, Harbin 150028, China
[4] Harbin Vocational College of Science and Technology, Harbin 150399, China
* Correspondence: lvxinmiao0611@hrbust.edu.cn; Tel.: +86-187-0462-8841

Abstract: It is difficult for traditional circuit-fault feature-extraction methods to accurately distinguish between nonlinear analog-circuit faults and analog-circuit faults with high fault rates and high diagnostic costs. To solve this problem, this paper proposes a method of mathematical morphology fractal dimension (VMD-MMFD) based on variational mode decomposition for soft-fault feature extraction in analog circuits. First, the signal is decomposed into variational modes to suppress the influence of environmental noise, and multiple high-dimensional eigenmode functions with different center frequencies are obtained. The fractal dimension of the signal feature information component IMF is calculated, and then, KPCA (Kernel Principal Component Analysis) is used to remove the overlapping and redundant parts of the data. The fault set obtained is used as the basis for judging the working state and the fault type of the circuit. The experimental results of the simulation circuits show that this method can be effectively used for circuit-fault diagnosis.

Keywords: mathematical morphology fractal dimension; kernel principal component analysis; variational modal decomposition; feature extraction

1. Introduction

In daily life and production, circuits can be divided into two types: analog and digital. Although analog circuits generally account for less than 20% of circuit components, 80% of faults are caused by analog circuits [1]. The fault types of analog circuits can be divided into soft faults and hard faults. A hard fault refers to catastrophic faults such as open circuits and short circuits in electronic circuits, which are easy to identify. A soft fault indicates that the value of the components has changed, and the deviation in value exceeds the allowable range [2]. When a soft fault problem occurs, the circuit can still work normally, but if the component is not replaced in time, the soft fault will be upgraded to a hard fault, which will cause significant damage to the entire circuit [3] and even endanger people's lives and safety.

Many scholars have proposed many feature extraction methods for analog-circuit recognition. Common fault feature extraction methods of analog circuits include principal component analysis [4], factor analysis [5,6], and other linear extraction methods. These methods are relatively effective for linear circuits, but most circuits in daily life have nonlinear features. The above methods cannot reflect the non-stationary characteristics of the signal, resulting in the low separability of the extracted fault features, so there is a large classification error in fault-pattern recognition. Therefore, early soft fault research mainly introduced fuzzy algorithms, wavelet theory, and other means to determine the actual working conditions [7]. Although this algorithm improves the effect of fault diagnosis, some algorithms are seriously affected by the circuit state when analyzing fault characteristics, which makes the performance unstable. In order to solve this problem, in

recent years, some scholars have combined mathematical morphology with wavelets and proposed a new nonlinear wavelet morphological wavelet. Reference [8] applied morphological wavelets to detect power disturbance and successfully identified bad conditions in the power transmission process. For example, Zhuang Ning [9] and others combined fractal dimension with EMD (empirical mode decomposition) to extract features of ECG signals to identify different emotional performances. Zheng Zhi [10] and others combined the LMD (local mean decomposition) decomposition method with generalized fractal dimensions to identify gear faults. However, actual measured signals are often accompanied by abnormal events, such as the noise impact and intermittent signals, which lead to modal aliasing and greatly affect the subsequent fault-diagnosis rate. At the same time, a common method to calculate the fractal dimension is the covering method (the box-counting method). This method has inevitable shortcomings. The method itself uses the regular grid division method, so the estimation of the fractal dimension is very unstable in some cases [11].

In view of the above problems, this paper proposes a method to calculate the fractal dimensions (VMD-MMFD-KPCA) of mathematical morphology based on variational mode decomposition (VMD). Using the variational mode decomposition (VMD) of the signal, the interference information in the signal is eliminated to the maximum extent possible to solve the problem of mode aliasing of traditional EMD decomposition. When using morphological fractal dimensions to calculate the dimensions of signals, the instability of traditional box-dimension estimation is avoided, including effectively distinguishing different fault types. On this basis, the KPCA dimension-reduction method is introduced to reduce the dimensions of high-dimensional data calculated by dimensions, eliminate the redundancy and duplication in the sample, and provide a data basis for subsequent fault diagnosis. Compared with traditional fault-diagnosis methods, it shows better performance of feature extraction and diagnosis. Finally, the effectiveness of this method is demonstrated by a simulation circuit.

2. Variational Modal Decomposition

VMD decomposition is a nonrecursive signal-decomposition algorithm. The intrinsic mode function (IMF) in different frequency bands is obtained by adaptively decomposing signals. In the process of solving the mode function, image continuation is used to avoid the endpoint effect in EMD (empirical mode decomposition) and other decomposition methods. VMD's processing of nonlinear fault signals is helpful for extracting the characteristics of subsequent fault signals. The decomposed IMF has an independent center frequency and sparsity and can effectively avoid mode aliasing when the parameters are appropriate.

(1) In order to obtain a unidirectional spectrum, Hilbert transform is used to analyze and calculate each modal signal, and then the frequency shifting method is used to move the modal spectrum to the baseband:

$$\left[\left(\delta(t)+\frac{j}{\pi t}\right)*v_k(t)\right]e^{j\omega_k t} \tag{1}$$

$\delta(t)$ is the Dirac function; v_k and ω_k are the first IMF component and its center frequency decomposed, respectively.

(2) The bandwidth is estimated by the square norm of the gradient, and the constraint expression is:

$$\begin{cases} \min\limits_{\{v_k\},\{\omega_k\}} \left\{ \sum\limits_{k=1}^{K} \left\| \partial_t \left[\left(\delta(t)+\frac{j}{\pi t}\right)*v_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \\ s.t. \qquad \sum\limits_{k=1}^{K} v_k(t) = f(t) \end{cases} \tag{2}$$

∂_t is the gradient calculation, and * is the convolution calculation symbol.

(3) In order to obtain the optimal solution more efficiently, the constrained problem is transformed into an unconstrained problem by using a Lagrangian operator $\tau(t)$ and penalty factor α. The expanded Lagrangian function expression is:

$$\begin{aligned} L(\{v_k\},\{\omega_k\},\tau) = \alpha \sum_{k=1}^{K} \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * v_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \\ + \left\| f(t) - \sum_{k=1}^{K} v_k(t) \right\|_2^2 + \left[\tau(t), f(t) - \sum_{k=1}^{K} v_k(t) \right] \end{aligned} \tag{3}$$

(4) The alternating direction multiplier method is used to iteratively update each modal component and the center frequency, and the saddle point of the unconstrained function is obtained, which is the best solution to the problem. The iterative update expression of v_k, ω_k, τ is as follows:

$$\hat{v}_k^{n+1}(\omega) = \frac{\hat{f}(\omega) - \sum\limits_{i \neq k} \hat{v}_i(\omega) + \frac{\hat{\tau}(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2} \tag{4}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega |\hat{v}_k(\omega)|^2 d\omega}{\int_0^\infty |\hat{v}_k(\omega)|^2 d\omega} \tag{5}$$

$$\hat{\tau}^{n+1}(\omega) = \hat{\tau}^n(\omega) + \gamma \left[\hat{f}(\omega) - \sum_{k=1}^{K} \hat{u}_k^{n+1}(\omega) \right] \tag{6}$$

In Equation (6) n is the number of iterations and γ is the noise tolerance parameter.

(5) Judgment of the iteration termination condition.

$$\frac{\sum\limits_{k=1}^{K} \left\| \hat{v}_k^{n+1} - \hat{v}_k^n \right\|_2^2}{\sum\limits_{k=1}^{K} \left\| \hat{v}_k^n \right\|_2^2} < \varepsilon \tag{7}$$

where ε is the judgment accuracy. When $\varepsilon > 0$, the iteration stops and the cycle ends. The step flow chart is shown in Figure 1:

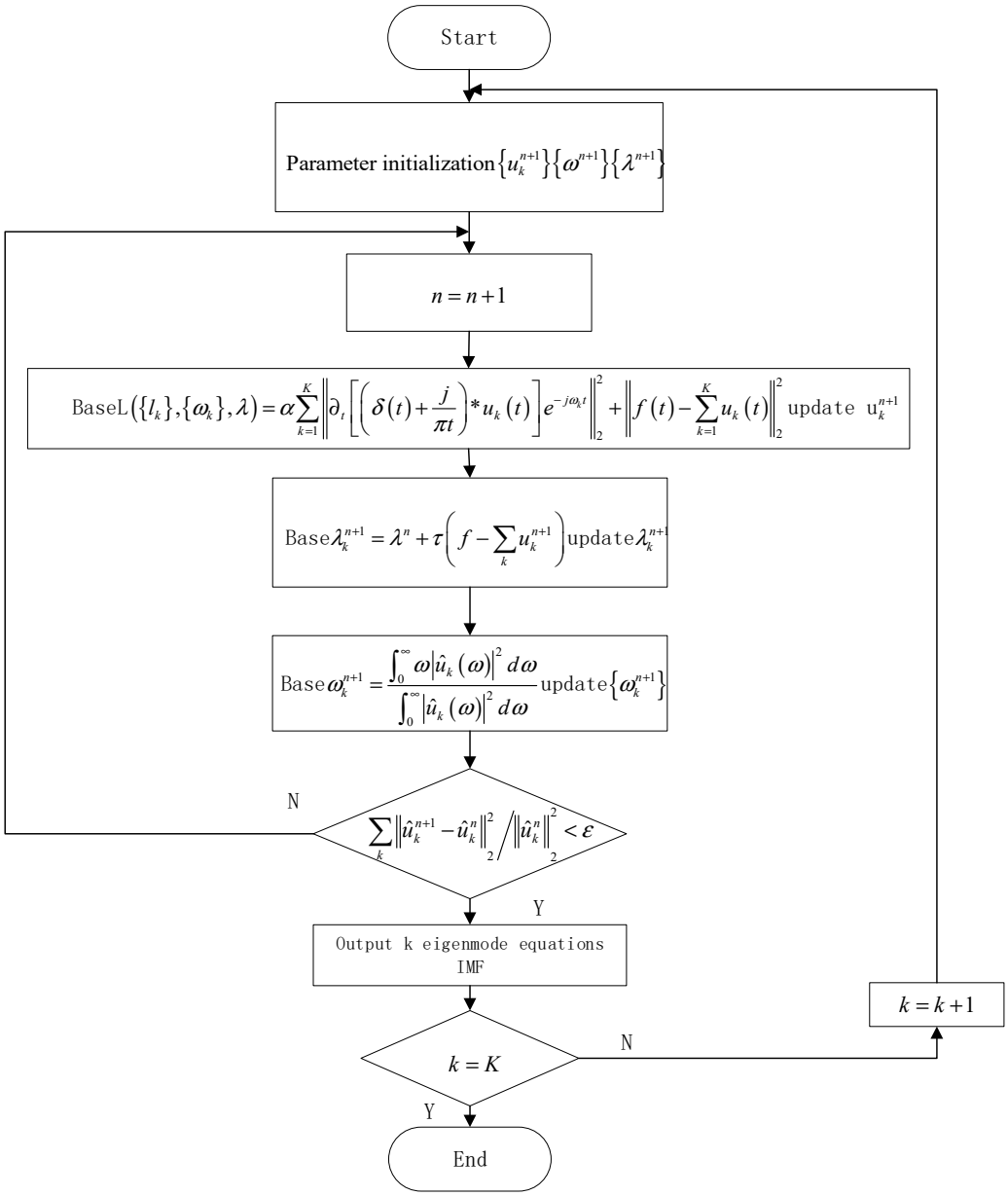


Figure 1. Variational Modal Decomposition Process.

3. Fractal Dimension Calculation Method Based on Mathematical Morphology

The key to estimating the fractal dimension is to measure the signal at different scales, which can be realized using multi-scale morphology [12]. In the process of gathering and covering the signal, the supremum function, i.e., the structural element g (n), is used for equivalent transformation, and the scale range $\varepsilon = 1, 2, \cdots, \varepsilon_{\max}$ is analyzed. The algorithm includes two basic operators: expansion and corrosion. For a one-dimensional discrete time signal $f(n)(n = 0, 1, 2, \cdots)$, the expansion and corrosion results of each scale are, respectively,

$$f\Theta g^{\oplus\varepsilon}(n) = \underbrace{((f\Theta g)\Theta g\cdots)\Theta g}_{\varepsilon_{Times}} \tag{8}$$

$$f\oplus g^{\oplus\varepsilon}(n) = \underbrace{((f\oplus g)\oplus g\cdots)\oplus g}_{\varepsilon_{Times}} \tag{9}$$

where Θ represents corrosion operation; $\oplus$ represents expansion operation; $g^{\oplus\varepsilon}(n)$ represents the structural elements used in the scale ε; f represents signal; g represents a struc-

tural element; and ε_{Times} is the number of expansion and corrosion. The coverage area of discrete signals at different scales is defined as

$$A_g(\varepsilon) = \sum_{n=1}^{N} (f \oplus \varepsilon g(n) - f \Theta \varepsilon g(n)) \tag{10}$$

When ε approaches zero, $A_g(\varepsilon)$ satisfies:

$$\lg \frac{A_g(\varepsilon)}{\varepsilon^2} \approx D_M \cdot \lg\left(\frac{1}{\varepsilon}\right) + c \tag{11}$$

where D_M is the Minkowski–Bouligand dimension of the signal, and c is a constant. The slope of the straight line obtained by a least-square fitting of the above equations $\lg \frac{A_g(\varepsilon)}{\varepsilon^2}$ and $\lg\left(\frac{1}{\varepsilon}\right)$ is D_M, which is the final required $f(n)$ fractal dimension.

Generally, the unit structure element $g(n)$ is chosen to be $\{0, 0, 0\}$, because this structure not only ensures that the dimension estimation is not affected by the signal amplitude range but also reduces the computational complexity of the algorithm. In principle, the maximum analysis scale $\varepsilon_{\max}$ is a positive integer less than $N/2$ (N is the number of discrete signal sampling points). When the data length is relatively large, appropriately reducing $\varepsilon_{\max}$ can reduce the amount of calculation [13]. In this paper, the length is 256.

4. Kernel Principal Component Analysis

The KPCA method is used to represent the nonlinear relationship between modeling data. It can effectively project linearly indivisible input data into a high-dimensional space that can be linearly separated, and then execute linear *PCA* in feature space H. Assume that the sample set is $x_1, x_2, \cdots, x_N \in R^m$, where: N is the number of samples and m is the number of variables. These samples are projected into the feature space H through a nonlinear mapping ϕ, which can be expressed as:

$$\phi : x_i \in R^m \to \phi(x_i) \in R^h \tag{12}$$

where h is the dimension of the feature space.

The dot product of vector $\phi(x_i)$ and $\phi(x_j)$ (i , $j = 1$, 2 , $\cdots$, N) in feature space H is:

$$\phi(x_i)^T \phi(x_j) = \langle \phi(x_i), \phi(x_j) \rangle = k(x_i, x_j) \tag{13}$$

where: k is the kernel function.

The covariance matrix of samples in the high-dimensional feature space H is:

$$Q = \frac{1}{N-1} \sum_{j=1}^{N} \phi(x_j) \phi(x_j)^T \tag{14}$$

Similar to linear PCA, KPCA in the feature space is equivalent to solving the eigenvalue problem. Let the eigenvector be V_k, the eigenvalue be μ_k, and its characteristic equation be:

$$\begin{gathered} \mu_k V_k = Q V_k = \frac{1}{N-1} \sum_{j=1}^{N} \phi(x_j) \phi(x_j)^T V_k = \\ \frac{1}{N-1} \sum_{j=1}^{N} \langle \phi(x_j), V_k \rangle \phi(x_j) \end{gathered} \tag{15}$$

In Equation (15), each eigenvector V_k in the covariance matrix Q can be regarded as a linear combination of $\phi(x_1), \phi(x_2), \cdots, \phi(x_N)$ [14], that is:

$$V_k = \sum_{i=1}^{N} v_{k,i} \phi(x_i), \qquad k = 1, 2, \cdots, N \tag{16}$$

where v_k and i are linear correlation coefficients.

Combining Equations (13) and (15), we can obtain

$$\mu_k \sum_{i=1}^{N} v_{k,i} K_{k,i} = \frac{1}{N-1} \sum_{i=1}^{N} v_{k,i} \sum_{j=1}^{N} K_{k,i} K_{ji} \tag{17}$$

Further simplification results in:

$$\lambda_k v_k = K v_k \tag{18}$$

where $\lambda_k = (N-1)\mu_k$; $K \in R^{N \times N}$ is a kernel matrix, whose eigenvalues are $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$, and the corresponding eigenvectors are $v_1, v_2, \cdots, v_N$. In the feature space H, the eigenvector V_k of the covariance matrix Q in Equation (15) forms a matrix $V_f = [V_1 \quad V_2 \quad \cdots \quad V_l \quad V_{l+1} \quad \cdots \quad V_N]$. One principal element (PC) is selected, and we obtain $\hat{V}_f = [V_1 \quad V_2 \quad \cdots \quad V_l] \in R^{N \times l}$ in the principal element space, since the eigenvector V_k should meet the normal constraint in the feature space H, that is, $\langle V_k, V_k \rangle = 1$. Thus, the eigenvectors $v_1, v_2, \cdots, v_N$ of the kernel matrix K can be expressed as:

$$\langle v_k, v_k \rangle = \frac{\langle V_k, V_k \rangle}{\lambda_k} \tag{19}$$

The norm of $\frac{1}{\sqrt{\lambda_k}}$ is taken as the eigenvector, so we have

$$v_k = \frac{1}{\sqrt{\lambda_k}} v^k \tag{20}$$

$\hat{V}_f$ in feature space H is expressed as

$$\hat{V}_f = \left[\frac{1}{\sqrt{\lambda_1}} X^T v^l \right] = X^T \hat{V} \Lambda^{-1/2} \tag{21}$$

where $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_l)$;$\hat{V} = [v^1 \quad v^2 \quad \cdots \quad v^l]$;$X = [\phi(x_1) \quad \phi(x_2) \quad \phi(x_N)]^T$.

5. Basic Theory of SVM

SVM can provide good generalization performance, and it has been widely used in small-sample classification [15]. As a classical nonlinear problem in the area of fault-location problems, SVM cannot find the classification hyperplane in the original sample space, so it is necessary to introduce a kernel function $K(x_i, x_j)$ to map to a linearly separable high-dimensional space. In this paper, a Gaussian kernel function mapping transformation is used, and the expression is as follows:

$$K(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) \tag{22}$$

where, $\left\| x_i - x_j \right\|^2$ is the square of the distance between x_i and x_j, and r is the super parameter of the kernel function itself:

$$\begin{aligned} &\min_{\omega, b, \zeta} \tfrac{1}{2} \omega^T \omega + c \sum_{i=1}^{N} \zeta_i \\ &s.t. \quad y_i(\omega^T \phi(x_i)) \geq 1 - \zeta_i, \quad i = 1, 2, \cdots N \\ &\qquad \zeta_i \geq 0, \quad i = 1, 2, \cdots, N \end{aligned} \tag{23}$$

where ζ_i is the relaxation variable, c is the penalty factor, x_i is the input i sample, and y_i is the sample type.

The Lagrangian penalty factor α is introduced, and finally the dual type of soft interval Gaussian kernel SVM is obtained as follows:

$$\begin{aligned} &\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) - \sum_{i=1}^{N} \alpha_i \\ &s.t. \quad 0 \le \alpha_i \le c, \qquad i = 1, 2, \cdots N, \\ &\qquad \sum_{i=1}^{N} \alpha_i y_i = 0 \end{aligned} \tag{24}$$

6. Experiment and Simulation

The method based on VMD-MMFD is used for feature extraction and the fault diagnosis of fault circuits. The specific process is shown in Figure 2.

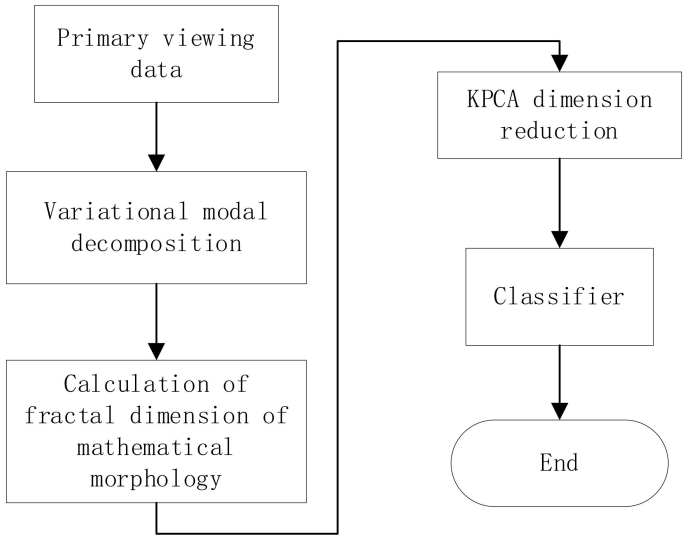


Figure 2. VMD-MMFD Method Flow Chart.

To verify the effectiveness of this method, simulation experiments and calculations were carried out through the software Multisim, Matlab, and Python. The Sallen–Key band-pass filter circuit was taken as an example to verify the analysis method of soft-fault diagnosis based on variational-mode decomposition and mathematical morphology fractal dimensions. The circuit diagram is shown in Figure 3.

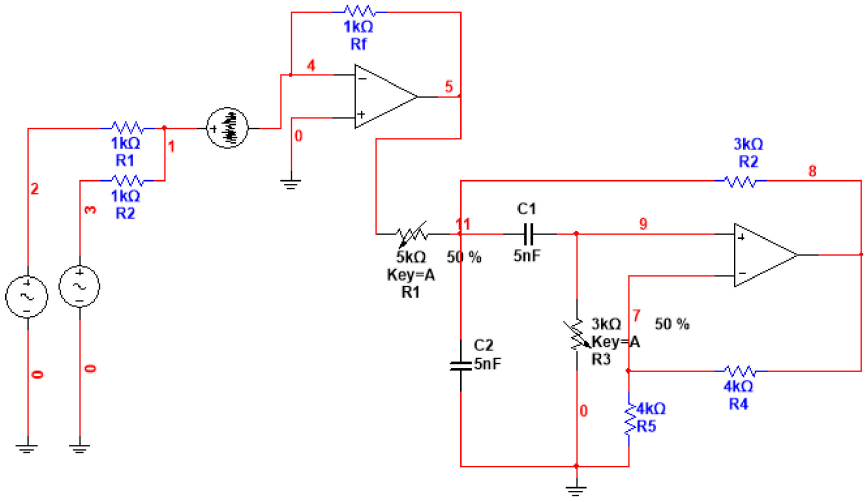


Figure 3. Sallen−Key band-pass filter circuit.

The excitation signal f = 3sin (30) + 3sin (20) + noise is used as the input source for the fault diagnosis, the tolerance of resistance and capacitance of the circuit is 10%, and the value of components within this range is considered normal. In this experiment, the soft fault state is set when the nominal value fluctuates by 30%. In this experiment, 15-parameter fault combinations are configured. The fault of the amplifier is affected by many factors, so it is not discussed here. The fault mode setting is shown in Table 1, the original signal is shown in Figure 4, the noise signal is shown in Figure 5, and the mixed noise input signal is shown in Figure 6.

Table 1. Fault Type Description.

Type Number	Type	Nominal Value	Fault Zone
1	Normal	-	-
2	$R_1 \uparrow 30\%$	4 kΩ	[4.4 kΩ, 5.2 kΩ]
3	$R_1 \downarrow 30\%$	4 kΩ	[2.8 kΩ, 3.6 kΩ]
4	$R_2 \uparrow 30\%$	3 kΩ	[3.3 kΩ, 3.9 kΩ]
5	$R_2 \downarrow 30\%$	3 kΩ	[2.1 kΩ, 2.7 kΩ]
6	$R_3 \uparrow 30\%$	2 kΩ	[2.2 kΩ, 2.6 kΩ]
7	$R_3 \downarrow 30\%$	2 kΩ	[1.4 kΩ, 1.8 kΩ]
8	R4 ↑ 30%	4 kΩ	[4.4 kΩ, 5.2 kΩ]
9	R4 ↓ 30%	4 kΩ	[2.8 kΩ, 3.6 kΩ]
10	R5 ↑ 30%	4 kΩ	[4.4 kΩ, 5.2 kΩ]
11	R5 ↓ 30%	4 kΩ	[2.8 kΩ, 3.6 kΩ]
12	C1 ↑ 30%	5 nF	[5.5 nF, 6.5 nF]
13	C1 ↓ 30%	5 nF	[3.5 nF, 4.5 nF]
14	C2 ↑ 30%	5 nF	[5.5 nF, 6.5 nF]
15	C2 ↓ 30%	5 nF	[3.5 nF, 4.5 nF]

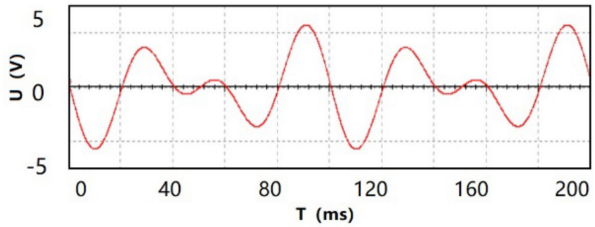


Figure 4. Original Signal.

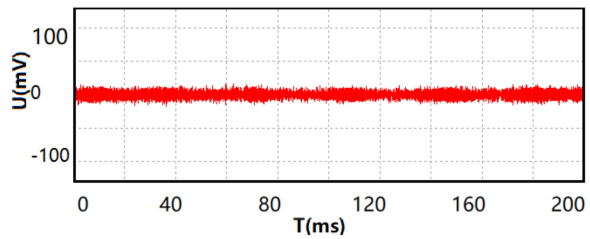


Figure 5. Noise Signal.

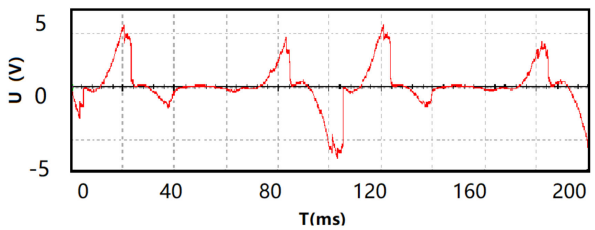


Figure 6. Input signal of mixed noise.

When VMD is used to decompose signals, the preset scale parameters and the second penalty factor are the main parameters that affect the decomposition accuracy. Therefore, for the VMD decomposition of measured signals, the reasonable selection of its parameters is the difficulty and key of this method. The center frequency of each order of IMF component obtained from the VMD decomposition of the signal is distributed from a low frequency to a high frequency. If the optimal preset scale parameter K is obtained, the center frequency of the last-order IMF component should be the maximum value, and the maximum center frequency value should remain stable. Based on the test and analysis of the VMD decomposition results of a large number of measured signals and reference [16], the second penalty factor $\alpha = 2000$ is selected in this paper. Take fault type 3 and fault type 5 signals as examples of VMD decomposition, as shown in Figures 7 and 8, the center frequency of the IMF component is obtained by decomposition, as shown in Figures 9 and 10. It can be seen from the figures that when K = 4, the central frequency of the IMF component reaches the maximum and tends to be stable, the frequencies between modes do not overlap, and the impact of noise and mode aliasing is effectively suppressed. With the preset scale parameter K > 4, the center frequencies of IMF 5 and IMF 6 components become unstable and mode aliasing occurs. Therefore, the VMD decomposition of the signal is the best when k = 4.

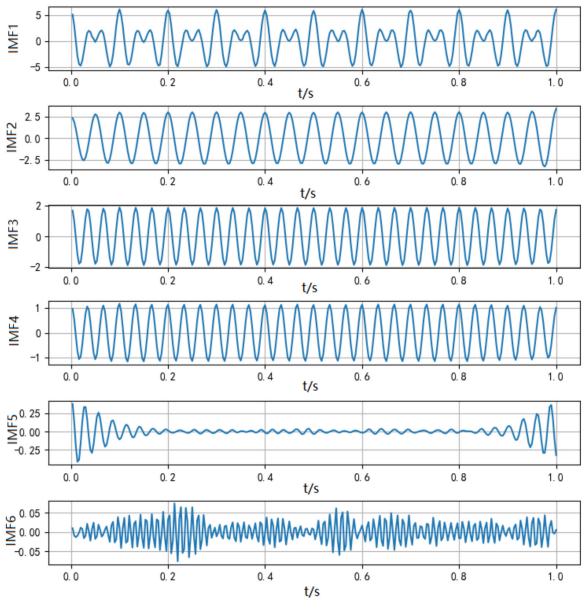


Figure 7. VMD Decomposition Sequence Diagram of Fault 3 Signal.

In recent years, in the field of circuit fault diagnosis, the method of combining empirical mode decomposition with box-dimension estimation has been commonly used to extract features [17]. The experiment compares this traditional method with the model of feature extraction based on variational mode decomposition and mathematical morphology. Take four states of types 2–5 as examples, where q is the scale and the value range is [0, 10]. Each state corresponds to 10 fractal dimensions. With the increase in the scale q, fault states are depicted from different dimensions, Dq is the fractal dimension value under different scales q, and the spectrum analysis is carried out.

In Figure 11, the IMF1 component of Fault03 does not overlap with other components, and the IMF1 components of Fault02, Fault04, and Fault05 have aliasing and different degrees of crossing. With the increase in q, the aliasing and crossing become more and more serious, making them difficult to distinguish. In Figure 12, the IMF1 components of Fault02 and Fault04 have serious crossing phenomena, and it is difficult to distinguish the fault types. When the q value is small, the IMF1 of Fault03 and Fault05 have no aliasing phenomena. With the increase in q, the IMF1 of Fault04 has serious cross phenomena with the IMF1 of Fault03 and Fault05. In Figure 13, the IMF1 of Fault04 and Fault05 has seriously

crossed, and it is impossible to distinguish the fault type. With the increase in the q value, the IMF1 of Fault03 and Fault04 has also mixed or even crossed phenomena. In Figure 14, when the q value is low, each fault type can be distinguished. With the increase in the q value, the IMF1 of Fault04 and Fault05 has aliasing and crossing phenomena, which affect the discrimination of the fault types. Later, with the increase in the scale of q, the IMF1 of Fault04 and Fault03-IMF1 has a certain degree of aliasing. The characteristic sample set of the IMF1 component dimension of the fault signal is shown in Table 2.

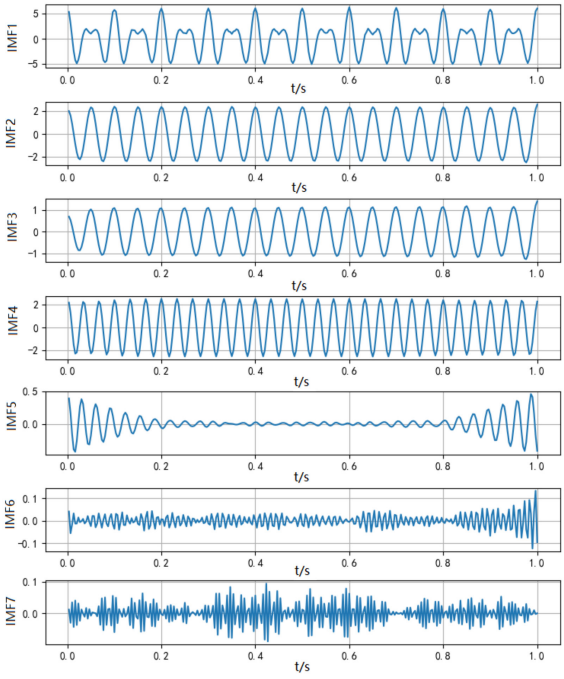


Figure 8. VMD Decomposition Sequence Diagram of Fault 5 Signal.

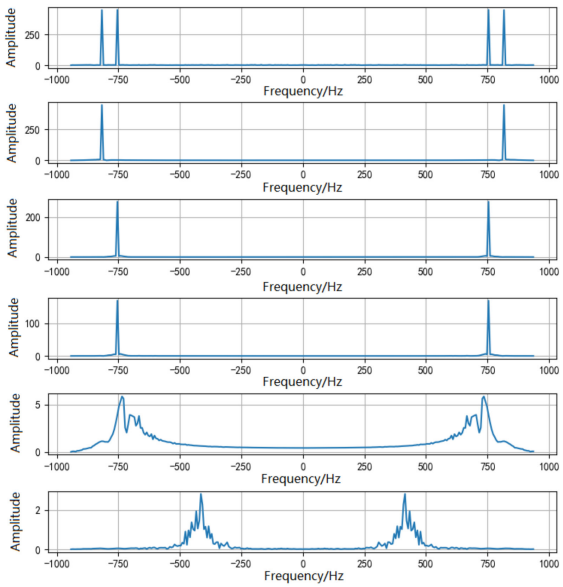


Figure 9. IMF Component Corresponding Spectrum of Fault 3.

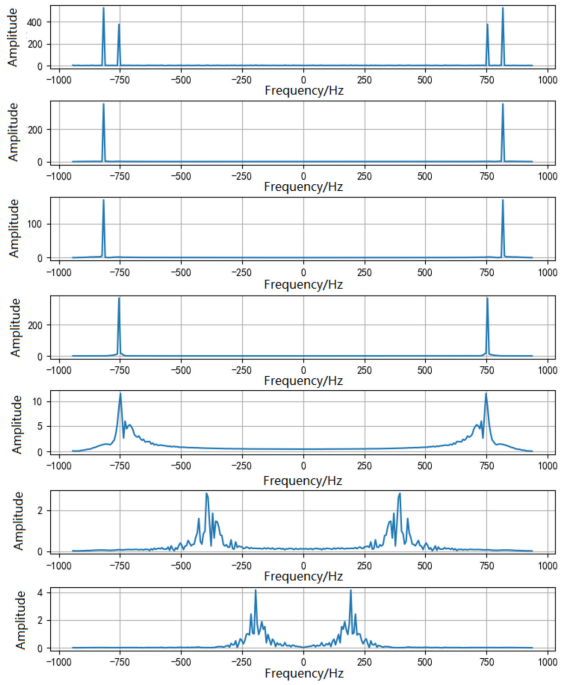


Figure 10. IMF Component Corresponding Spectrum of Fault 5.

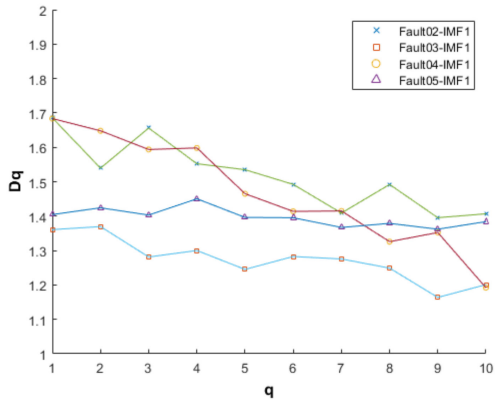


Figure 11. IMF1-EMD box fractal dimension map.

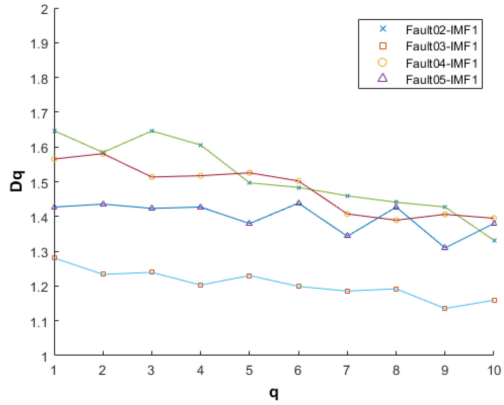


Figure 12. IMF1-EMD mathematical morphology fractal dimension map.

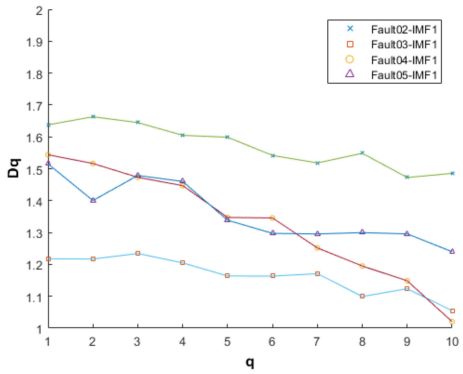


Figure 13. IMF1-VMD box fractal dimension map.

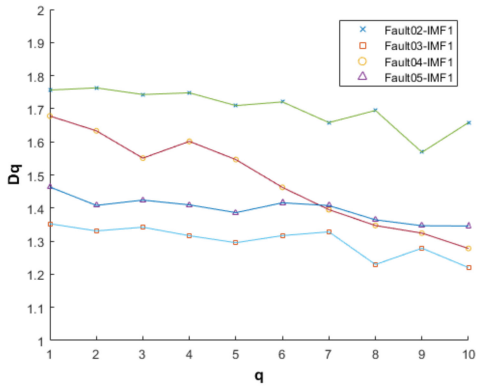


Figure 14. IMF1-VMD mathematical morphology fractal dimension map.

Table 2. Characteristic sample of the fault signal analysis dimension.

Fault Type		D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}
IMF1-EMD box fractal dimension	Fault 2	1.6870	1.6743	1.6061	1.6156	1.5567	1.4838	1.4095	1.4910	1.3974	1.4057
	Fault 3	1.3602	1.3735	1.2853	1.2908	1.2303	1.2689	1.2595	1.2419	1.1655	1.2092
	Fault 4	1.6854	1.6508	1.5835	1.6074	1.4556	1.4123	1.4073	1.3394	1.3462	1.1949
	Fault 5	1.4071	1.4256	1.4133	1.4572	1.4097	1.4085	1.3832	1.4120	1.3796	1.4140
IMF1-EMD mathematical morphology fractal dimension	Fault 2	1.6557	1.5995	1.6661	1.6124	1.5050	1.4923	1.4890	1.4820	1.4785	1.3868
	Fault 3	1.2908	1.2497	1.2513	1.2051	1.2482	1.2010	1.1755	1.1980	1.1438	1.1610
	Fault 4	1.5835	1.5981	1.5337	1.5402	1.5452	1.5372	1.4151	1.3950	1.4024	1.3920
	Fault 5	1.4245	1.4288	1.4231	1.4305	1.3863	1.4354	1.3571	1.4290	1.3221	1.3840
IMF1-VMD box fractal dimension	Fault 2	1.6377	1.6630	1.6447	1.6047	1.5988	1.5418	1.5182	1.5485	1.4724	1.4856
	Fault 3	1.2176	1.2186	1.2342	1.2047	1.1636	1.1646	1.1709	1.0993	1.1236	1.0548
	Fault 4	1.5439	1.5164	1.4729	1.4471	1.3472	1.3457	1.2516	1.1949	1.1485	1.0201
	Fault 5	1.5152	1.4002	1.4789	1.4600	1.3385	1.2972	1.2955	1.2996	1.2957	1.2398
IMF1-VMD mathematical morphology fractal dimension	Fault 2	1.7564	1.7629	1.7429	1.7480	1.7094	1.7206	1.6582	1.6947	1.5685	1.6582
	Fault 3	1.3526	1.3309	1.3422	1.3164	1.2954	1.3168	1.3281	1.2294	1.2786	1.2202
	Fault 4	1.6780	1.6333	1.5508	1.6014	1.5463	1.4624	1.3951	1.3472	1.3241	1.2779
	Fault 5	1.4636	1.4081	1.4240	1.4096	1.3862	1.4158	1.4075	1.3644	1.3465	1.3457

In Figure 15, the IMF2 of Fault04 and Fault05 has undergone serious aliasing and crossing, and the fault type of the IMF2 of Fault02 and Fault03 can be distinguished, in which aliasing and crossing have occurred. In Figure 16, the IMF2 of Fault02 and Fault03 has been separated, and the fault type is well distinguished, but the IMF2 of Fault04 and Fault05 shows aliasing with the increase in the q value, which is difficult to distinguish. In Fault02 and Fault04 in Figure 17, there is no aliasing, the effect of the fault type discrimination is good, and the IMF2 of Fault02 and Fault04 has serious aliasing and cross

phenomena, so it is impossible to distinguish fault types. In Figure 18, when the scale of q is low, the state-differentiation effect of various fault types is good. With the increase in the q value, the IMF2 of Fault02, Fault03, Fault04, and Fault05 shows aliasing and crossing phenomena to different degrees. The characteristic sample set of the IMF2 component dimension of the fault signal is shown in Table 3.

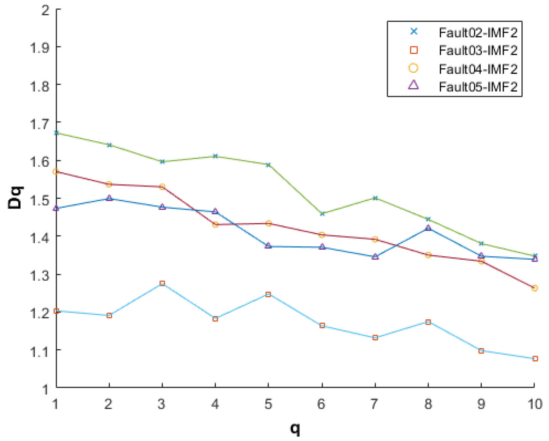


Figure 15. IMF2-EMD box fractal dimension map.

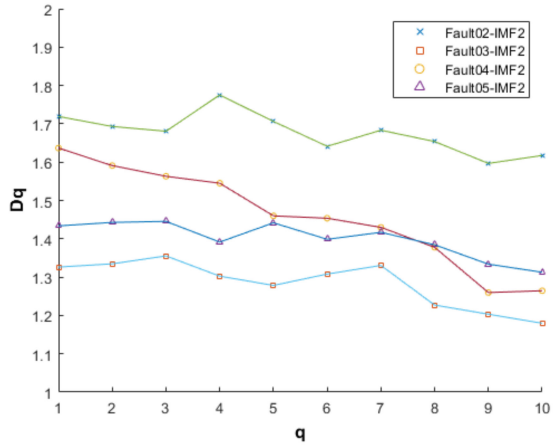


Figure 16. IMF2-EMD mathematical morphology fractal dimension map.

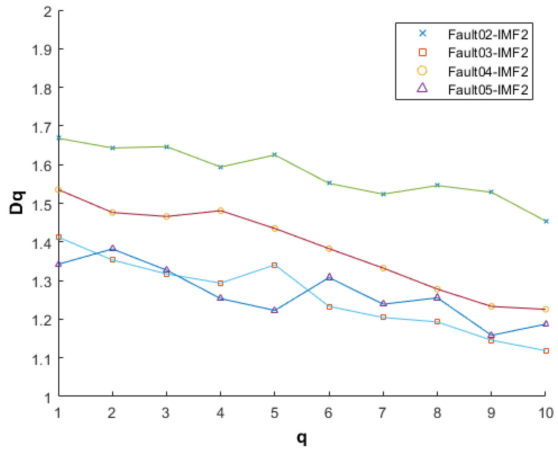


Figure 17. IMF2-VMD box fractal dimension map.

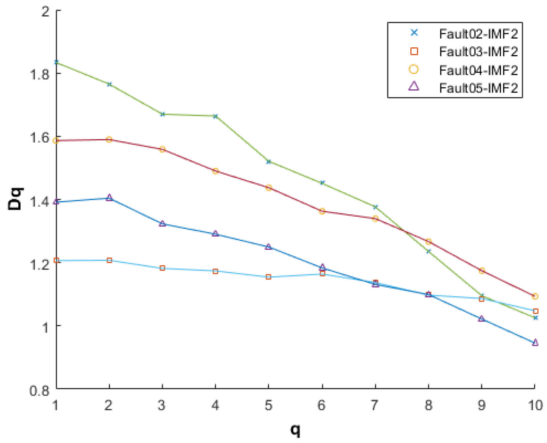


Figure 18. IMF2-VMD mathematical morphology fractal dimension map.

Table 3. Characteristic sample of the fault signal analysis dimension.

Fault Type		D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}
IMF2-EMD box fractal dimension	Fault 2	1.6724	1.6409	1.5960	1.6104	1.5887	1.4592	1.5010	1.4442	1.3803	1.3475
	Fault 3	1.2033	1.1910	1.2749	1.1832	1.2477	1.1637	1.1319	1.1749	1.0982	1.0771
	Fault 4	1.5704	1.5366	1.5300	1.4305	1.4335	1.4065	1.3918	1.3501	1.3340	1.2630
	Fault 5	1.4732	1.4991	1.4765	1.4639	1.3728	1.3707	1.3455	1.4209	1.3471	1.3389
IMF2-EMD mathematical morphology fractal dimension	Fault 2	1.7191	1.6930	1.6806	1.7753	1.7065	1.6413	1.6832	1.6537	1.5967	1.6174
	Fault 3	1.3262	1.3347	1.3554	1.3026	1.2784	1.3080	1.3305	1.2270	1.2034	1.1797
	Fault 4	1.6367	1.5909	1.5630	1.5450	1.4599	1.4536	1.4299	1.3777	1.2598	1.2645
	Fault 5	1.4337	1.4428	1.4455	1.3915	1.4417	1.3987	1.4169	1.3843	1.3339	1.3127
IMF2-VMD box fractal dimension	Fault 2	1.6682	1.6430	1.6465	1.5938	1.6249	1.5514	1.5236	1.5458	1.5285	1.4540
	Fault 3	1.4124	1.3532	1.3172	1.2933	1.3407	1.2331	1.2043	1.1930	1.1455	1.1187
	Fault 4	1.5354	1.4761	1.4657	1.4808	1.4348	1.3826	1.3322	1.2779	1.2330	1.2255
	Fault 5	1.3421	1.3818	1.3269	1.2532	1.2227	1.3078	1.2388	1.2553	1.1584	1.1871
IMF2-VMD mathematical morphology fractal dimension	Fault 2	1.8336	1.7650	1.6697	1.6639	1.5206	1.4509	1.3764	1.2351	1.0954	1.0255
	Fault 3	1.2066	1.2076	1.1819	1.1740	1.1543	1.1640	1.1372	1.0974	1.0867	1.0477
	Fault 4	1.5862	1.5894	1.5584	1.4899	1.4371	1.3627	1.3394	1.2667	1.1744	1.0933
	Fault 5	1.3914	1.4041	1.3228	1.2901	1.2495	1.1834	1.1311	1.0992	1.0216	0.9455

In Figure 19, with the increase in the number of IMF components, the dimension values of several components of the IMF are no longer stable, because with the increase in the number of modal decomposition components, each component contains less and less original feature information. After the dimension calculation process, the fault types become difficult to distinguish. When the scale of q is small, Fault04 and Fault05 have been aliased and crossed to some extent. With the increase in the scale of the q value, the IMF3 of Fault02, Fault04, and Fault05 has been aliased and crossed to some extent. In Figure 20, the IMF3 aliasing of Fault02, Fault03, Fault04, and Fault05 is obviously crossed, which makes it difficult to distinguish the fault type*. In Figure 21, the IMF3 fault type * of Fault02 is not aliased with other components, and the IMF3 of Fault03, Fault04, and Fault05 is completely aliased, so the fault types cannot be distinguished. In Figure 22, the IMF3 of Fault03 and Fault05 is seriously aliased and crossed, which makes it impossible to classify fault types. The IMF3 of Fault04 and Fault02 is aliased and crossed with the increase in the scale of q, which makes it difficult to classify faults. The sample set of dimension characteristics of the IMF3 components of fault signals is shown in Table 4.

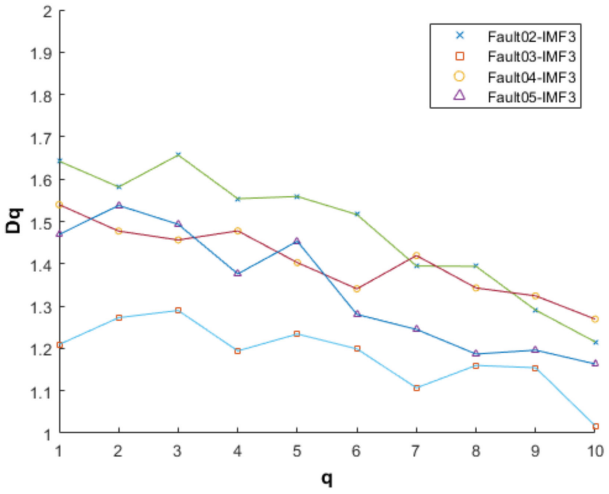


Figure 19. IMF3-EMD box fractal dimension map.

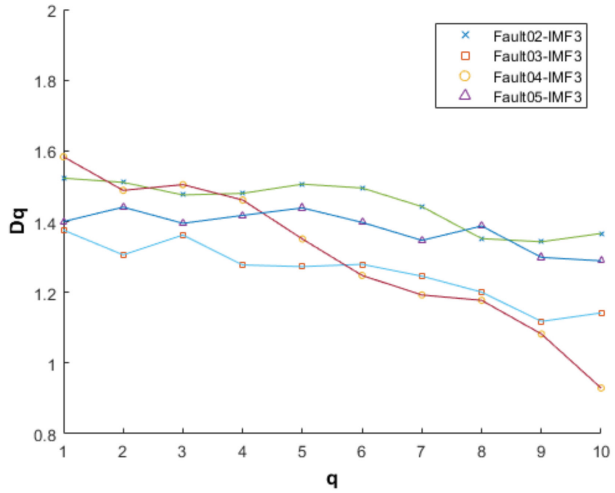


Figure 20. IMF3-EMD mathematical morphology fractal dimension map.

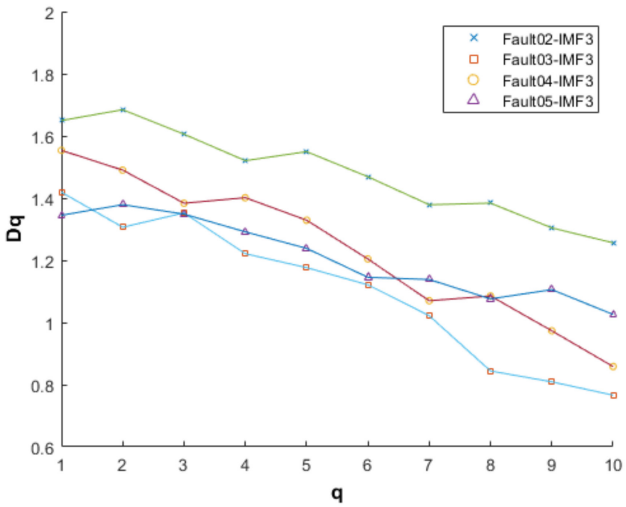


Figure 21. IMF3-VMD box fractal dimension map.

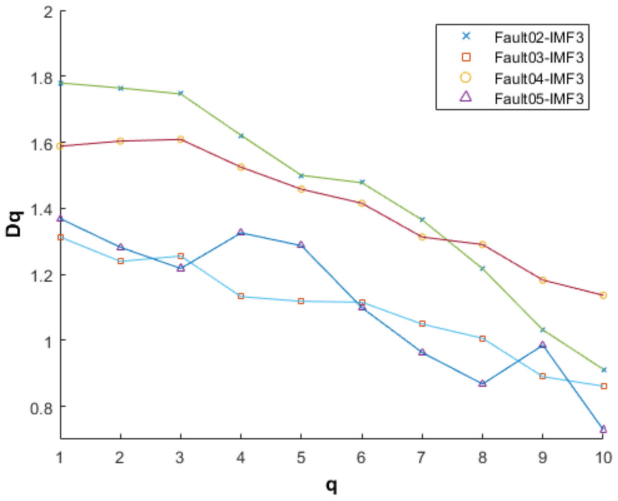


Figure 22. IMF3-VMD mathematical morphology fractal dimension map.

Table 4. Characteristic sample of the fault signal analysis dimension.

Fault Type		D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}
IMF3-EMD box fractal dimension	Fault 2	1.6425	1.5815	1.6568	1.5538	1.5591	1.5163	1.3947	1.3949	1.2907	1.2155
	Fault 3	1.2093	1.2727	1.2897	1.1941	1.2341	1.1991	1.1068	1.1599	1.1544	1.0175
	Fault 4	1.5398	1.4772	1.4563	1.4778	1.4028	1.3408	1.4195	1.3429	1.3243	1.2693
	Fault 5	1.4692	1.5379	1.4932	1.3761	1.4532	1.2804	1.2449	1.1868	1.1960	1.1634
IMF3-EMD mathematical morphology fractal dimension	Fault 2	1.5228	1.5109	1.4754	1.4798	1.5059	1.4950	1.4424	1.3519	1.3430	1.3664
	Fault 3	1.3759	1.3059	1.3622	1.2773	1.2727	1.2793	1.2458	1.2006	1.1177	1.1418
	Fault 4	1.5832	1.4883	1.5046	1.4610	1.3508	1.2476	1.1925	1.1773	1.0822	0.9293
	Fault 5	1.4000	1.4409	1.3951	1.4175	1.4387	1.3977	1.3469	1.3875	1.2988	1.2890
IMF3-VMD box fractal dimension	Fault 2	1.6495	1.6844	1.6046	1.5205	1.5495	1.4687	1.3787	1.3839	1.3042	1.2562
	Fault 3	1.4191	1.3063	1.3525	1.2212	1.1764	1.1212	1.0217	0.8440	0.8091	0.7661
	Fault 4	1.5537	1.4901	1.3837	1.4014	1.3290	1.2042	1.0699	1.0852	0.9733	0.8581
	Fault 5	1.3449	1.3789	1.3486	1.2915	1.2383	1.1452	1.1382	1.0759	1.1057	1.0257
IMF3-VMD mathematical morphology fractal dimension	Fault 2	1.7796	1.7642	1.7462	1.6209	1.4996	1.4773	1.3646	1.2168	1.0315	0.9121
	Fault 3	1.3130	1.2390	1.2553	1.1322	1.1181	1.1154	1.0490	1.0063	0.8897	0.8612
	Fault 4	1.5879	1.6034	1.6084	1.5244	1.4576	1.4153	1.3128	1.2901	1.1824	1.1364
	Fault 5	1.3690	1.2818	1.2176	1.3254	1.2867	1.0991	0.9629	0.8677	0.9853	0.7289

In Figure 23, with the increase in the number of IMF components, the dimension values of several IMF components fluctuate seriously, and the IMFs of Fault02, Fault03, Fault04, and Fault05 have all experienced serious aliasing and crossing. In Figure 24, the IMF4 in Fault02, Fault04, and Fault05 is aliased, so it is impossible to distinguish the fault type, and with the increase in the q value, the IMF3 in Fault03 is also aliased. In Figure 25, the IMF4 of Fault02, Fault04, and Fault05 shows serious aliasing and cross phenomena and even overlaps with the increase in the scale q, making it difficult to determine the fault type. Only the fault type of the IMF4 of Fault03 can be distinguished. In Figure 26, the IMF dimension maps of the four fault states fluctuate unstably, and the IMF4 of Fault02, Fault03, Fault04, and Fault05 is seriously aliased and crossed, making it impossible to judge the fault type. The characteristic sample set of the IMF4 component dimension of the fault signal is shown in Table 5.

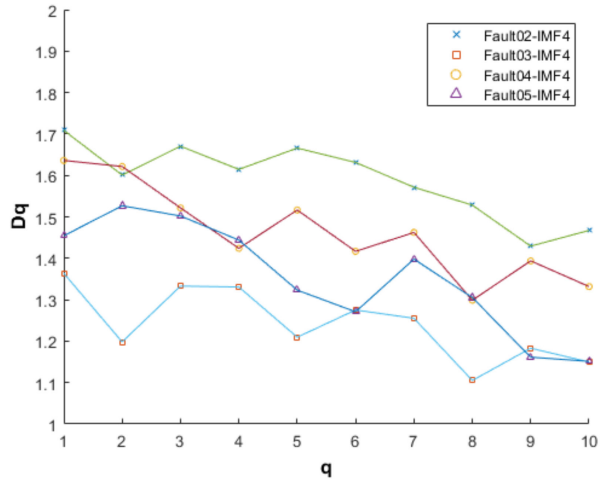


Figure 23. IMF4-EMD box fractal dimension map.

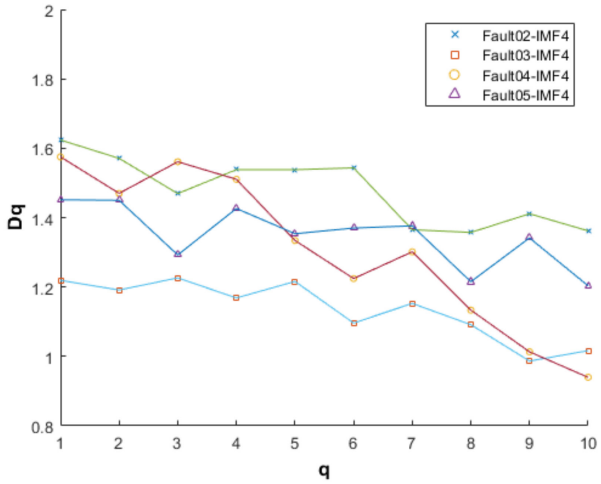


Figure 24. IMF4-EMD mathematical morphology fractal dimension map.

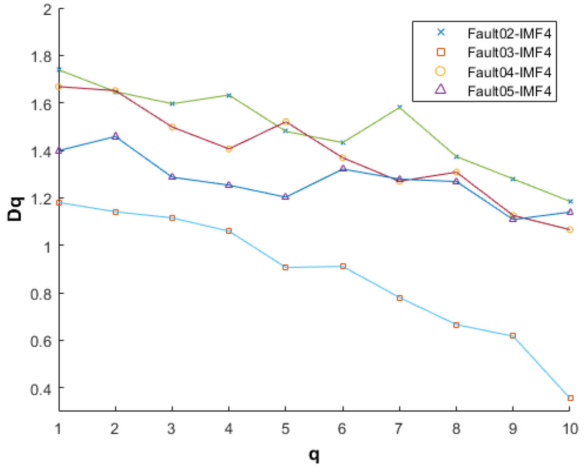


Figure 25. IMF4-VMD box fractal dimension map.

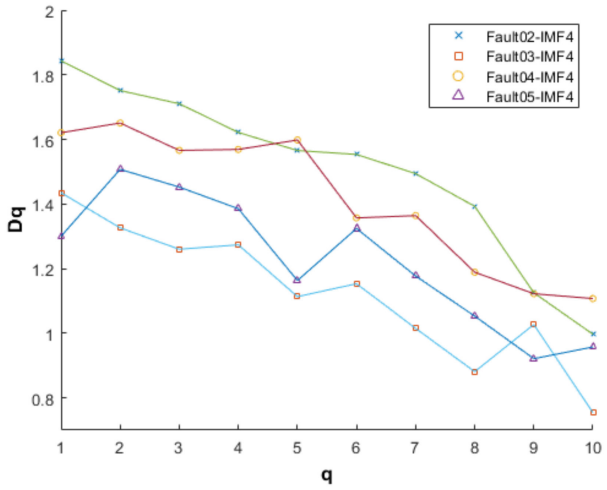


Figure 26. IMF4-VMD mathematical morphology fractal dimension map.

Table 5. Characteristic sample of the fault signal analysis dimension.

Fault Type		D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}
IMF4-EMD box fractal dimension	Fault 2	1.7086	1.6009	1.6704	1.6145	1.6659	1.6313	1.5714	1.5283	1.4293	1.4673
	Fault 3	1.3636	1.1981	1.3329	1.3307	1.2095	1.2752	1.2552	1.1051	1.1830	1.1496
	Fault 4	1.6361	1.6215	1.5215	1.4240	1.5164	1.4166	1.4624	1.2987	1.3937	1.3317
	Fault 5	1.4543	1.5268	1.5021	1.4439	1.3231	1.2708	1.3980	1.3052	1.1614	1.1513
IMF4-EMD mathematical morphology fractal dimension	Fault 2	1.6237	1.5708	1.4693	1.5382	1.5381	1.5434	1.3651	1.3573	1.4111	1.3618
	Fault 3	1.2192	1.1913	1.2263	1.1684	1.2160	1.0961	1.1527	1.0913	0.9866	1.0166
	Fault 4	1.5751	1.4699	1.5610	1.5105	1.3341	1.2244	1.3016	1.1332	1.0131	0.9396
	Fault 5	1.4514	1.4499	1.2924	1.4263	1.3537	1.3699	1.3763	1.2150	1.3415	1.2034
IMF4-VMD box fractal dimension	Fault 2	1.7392	1.6474	1.5963	1.6330	1.4800	1.4325	1.5802	1.3741	1.2797	1.1849
	Fault 3	1.1807	1.1413	1.1157	1.0596	0.9071	0.9112	0.7803	0.6663	0.6173	0.3559
	Fault 4	1.6682	1.6414	1.4984	1.4061	1.5206	1.3689	1.2697	1.3084	1.1258	1.0662
	Fault 5	1.3989	1.4584	1.2869	1.2534	1.2027	1.3211	1.2789	1.2685	1.1085	1.1401
IMF4-VMD mathematical morphology fractal dimension	Fault 2	1.8431	1.7512	1.7105	1.6216	1.5655	1.5535	1.4940	1.3929	1.1252	0.9969
	Fault 3	1.4350	1.3262	1.2600	1.2737	1.1133	1.1532	1.0151	0.8810	1.0270	0.7543
	Fault 4	1.6208	1.6509	1.5660	1.5690	1.5982	1.3570	1.3641	1.1889	1.1223	1.1073
	Fault 5	1.3001	1.5073	1.4525	1.3854	1.1631	1.3242	1.1771	1.0530	0.9213	0.9574

The feature extraction of different states of signals is carried out through the fractal dimension. Although the number of data samples increases and the fault set becomes larger, the high-dimensional data contain a lot of miscellaneous and repetitive data, which seriously affect the accuracy of fault diagnosis. The feature set needs to be dimensionally reduced. In the literature [18], different kernel functions have different effects on data dimensionality reduction. Generally, the Gaussian radial basis function (Formula (25)) is selected to perform the kernel function PCA on data sets to achieve dimensionality reduction and fault classification in high-dimensional feature space.

$$k(x,y) = \exp\left(-\frac{x-y^2}{2\sigma^2}\right) \tag{25}$$

The cumulative contribution rate of the principal component calculated by the KPCA algorithm is shown in Table 6. It can be seen from Table 6 that the first three principal components contain almost all the information of the system, so the first three principal components are retained as the eigenvalues of the system; that is, the feature space is reduced from high-dimensional to three-dimensional. KPCA can be used to calculate the princi-

pal component contribution rate, effectively eliminating the samples containing too much interference information, which greatly improves the accuracy of subsequent diagnoses.

Table 6. Cumulative contribution rate.

Cumulative Contribution Rate	KPCA1	KPCA2	KPCA3
Fault 2 (%)	93.26	95.59	98.88
Fault 3 (%)	95.33	96.21	99.11
Fault 4 (%)	96.34	98.88	100
Fault 5 (%)	97.77	99.18	100

Therefore, considering the feature extraction method of the VMD-MMFD-KPCA, the results are shown in Figure 27, and various fault types can be distinguished in the following figure.

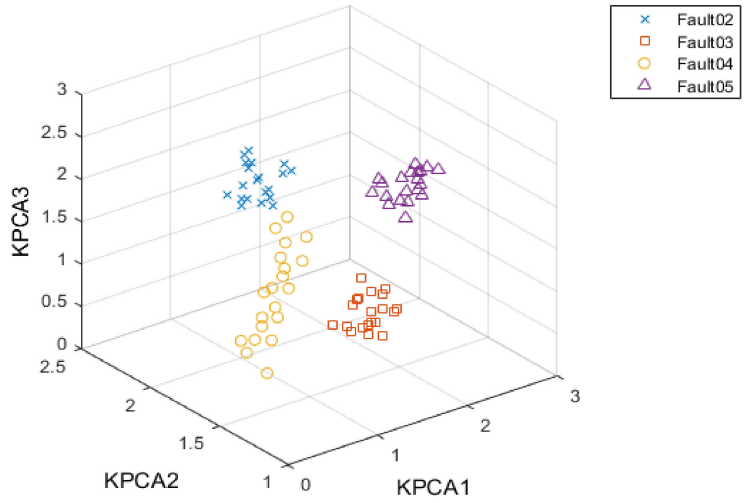


Figure 27. VMD-MMFD-KPCA feature extraction.

To further verify the effectiveness of the feature-extraction methods explained above, a support vector machine was used to complete fault diagnosis. Among them, the fault datasets that were used were all from the fault settings of various components of the simulation circuit. To achieve better experimental results, more than 50 signal samples were collected for each fault type, and more than 750 signal samples were collected for 15 fault types. Eighty percent of the samples from each fault type were randomly selected as the training sample, and the rest were selected as the test sample. The kernel function was introduced and determined in the previous theory, using Gaussian kernel function. After that, the data needed to be normalized. First, the data need to be filtered and sorted. For obviously abnormal data, we considered using the sample mean to replace it, so as to reduce the extent to which the abnormal sample data interferes with the results. To prevent the individual abnormal sample data volume from influencing the training results, the data were normalized according to Formula (26).

$$X_i^* = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} \tag{26}$$

After 100 iterations, diagnosis is carried out in combination with different diagnosis models, as shown in Figure 28.

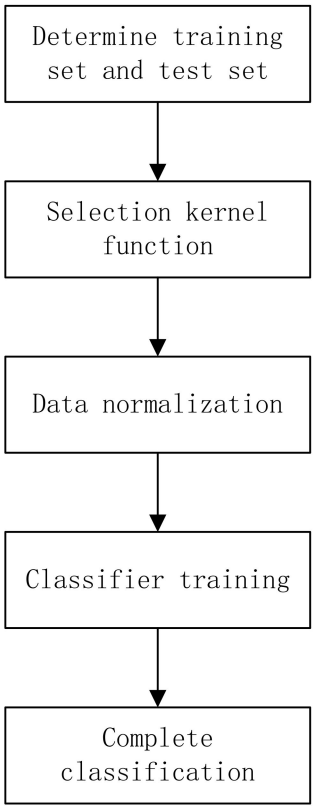


Figure 28. Flow chart of SVM Classification.

In the literature [19], there is a wavelet neural network (WNN) analog circuit soft fault diagnosis method based on kernel partial least square (KPLS) feature extraction, which uses support vector machine (KPLS-WNN-SVM) for fault classification. In the literature [20], a multifractal extraction method based on empirical mode decomposition is combined with support vector machine (EMD-MFD-SVM). The diagnostic rate of the diagnostic model combined with support vector machine (VMD-MMFD-KPCA-SVM) in this paper is shown in Table 7, which shows that the VMD-MMFD-KPCA-SVM method has the highest diagnostic accuracy.

Table 7. Fault Diagnosis Rate.

Diagnostic Model	Average Diagnostic Rate (%)
KPLS-WNN-SVM	87.3
EMD-MFD-SVM	91.9
VMD-MMFD-KPCA-SVM	96.3

7. Conclusions

This paper proposes a feature-extraction method based on VMD-MMFD-KPCA. Firstly, the fault signal is decomposed by VMD to obtain multiple modal IMF components, which greatly reduces the impact of modal aliasing. The appropriate IMF components were selected, and the fractal dimension of the mathematical morphology was calculated to obtain a large number of high-dimensional feature data sets. The data sets were reduced by KPCA to obtain low-dimensional data, which eliminated a large number of redundant and jumbled information in the high-dimensional data. The experimental results show that the feature-extraction method based on VMD-MMFD-KPCA has a good effect.

Author Contributions: Conceptualization, X.L.; methodology, X.L.; software, S.S.; validation, X.L., K.L., Q.W. and D.S.; writing—original draft preparation, C.Y., Z.L. and J.W.; writing—review and editing, C.Y.; project administration, X.L.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014.

Acknowledgments: The authors acknowledge Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Peng, W.; Yang, S. A new diagnosis approach for handling tolerance in analog and mixed-signal circuits by using fuzzy math. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2005**, *52*, 2118–2127. [CrossRef]
2. Zhu, Y.J.; Zhang, W.M.; Zhang, S.T.; Chen, J.J. Judgment and handling of "soft fault" of discrete component circuit. *Chin. Sci. Technol. J. Database Eng. Technol.* **2016**, *9*, 00300.
3. Lu, X.M.; Zhao, H.; Wu, Q. Research on fractal method for soft fault diagnosis of nonlinear analog circuits. *Model. Meas. Control. A* **2017**, *90*, 58–73. [CrossRef]
4. Han, H.T.; Ma, H.G.; Cao, J.F.; Zhang, J.L. Analog circuit fault diagnosis method based on nonlinear spectrum characteristics and kernel principal component analysis. *J. Electr. Technol.* **2012**, *27*, 248–2544.
5. Wang, Y.H.; Lu, J.; Pan, G.Q.; Feng, J.C. Research on Feature Extraction for Diagnostics of Analog Circuit Based on Factor Analysis. *Comput. Meas. Control* **2014**, *11*, 25–27.
6. Xiao, Y.; Feng, L. A novel neural-network approach of analog fault diagnosis based on kernel discriminant analysis and particle swarm optimization. *Appl. Soft Comput.* **2012**, *12*, 904–920. [CrossRef]
7. Guo, Q.; Zhang, W.B.; Su, H.T. Analog circuit fault diagnosis based on WPT and FOAGRNN. *Comput. Simul.* **2020**, *37*, 355–359.
8. Ji, T.Y.; Lu, Z.; Wu, Q.H. Detection of power disturbances using morphological gradient wavelet. *Signal Process.* **2008**, *88*, 255–267. [CrossRef]
9. Zhuang, N.; Zeng, Y.; Tong, L.; Zhang, C.; Zhang, H.; Yan, B. Emotion Recognition from EEG Signals Using Multidimensional Information in EMD Domain. *Biomed Res. Int.* **2017**, *2017*, 317–357. [CrossRef] [PubMed]
10. Zheng, Z.; Jiang, W.L.; Wang, Z.W.; Zhu, Y.; Yang, K. Gear fault diagnosis method based on local mean decomposition and generalized morphological fractal dimensions. *Mech. Mach. Theory* **2015**, *91*, 151–167. [CrossRef]
11. Nayak, S.R.; Mishra, J. Fractal Dimension Based Generalized Box-Counting Technique with Application to Grayscale Images. *Fractals* **2020**, *3*, 8. [CrossRef]
12. Zhang, K.; Cheng, J.S.; Yang, Y. A rolling bearing fault diagnosis method based on local mean decomposition and morphological fractal dimension. *J. Vib. Shock.* **2013**, *32*, 90–94.
13. Wang, F.J.; Duan, S.L.; Yu, H.L.; Li, H.K. Diesel Engine Fault Diagnosis Based on EEMD and Morphological Fractal Dimension. *J. Intern. Combust. Engine* **2012**, *30*, 557–562.
14. Yang, G.; Gu, X. Fault Diagnosis of Complex Chemical Processes Based on Enhanced Naive Bayesian Method. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 4649–4658. [CrossRef]
15. Gashteroodkhani, O.A.; Majidi, M.; Etezadi-Amoli, M.; Nematollahi, A.F.; Vahidi, B. A hybrid SVM-TT transform-based method for fault location in hybrid transmission lines with underground cables. *Electr. Power Syst. Res.* **2019**, *170*, 205–214. [CrossRef]
16. Tang, G.J.; Wang, X.L. Variational modal decomposition method and its application in early fault diagnosis of rolling bearings. *J. Vib. Eng.* **2016**, *29*, 11.
17. Han, D.Y.; Li, G.; Shi, P.M. Research on coupling fault diagnosis method of rotating machinery based on EMD and fractal box dimension. *Vib. Shock.* **2013**, *32*, 209–214.
18. Li, C.Y.; Liu, G.D.; Tan, B. Comparative study of hyperspectral remote sensing data dimensionality reduction based on PCA and KPCA. *Geospat. Inf.* **2022**, *20*, 89–93+103.
19. Cong, W.; Jing, B.; Yu, H.K. WNN soft fault diagnosis of analog circuit based on KPLS feature extraction. *J. Cent. South Univ. (Sci. Technol.)* **2014**, *45*, 1841–1846.
20. Lu, X.M.; Zhao, H.; Wu, Q. Generalized Multi-fractal Method for Soft Fault Feature Extraction of Non-linear Analog Circuits Based on EMD. *Rev. De La Fac. De Ing.* **2016**, *31*, 192–203.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Soft Fault Diagnosis of Analog Circuit Based on EEMD and Improved MF-DFA

Xinmiao Lu [1,*], Zihan Lu [1], Qiong Wu [1,2], Jiaxu Wang [1], Cunfang Yang [1], Shuai Sun [1], Dan Shao [3] and Kaiyi Liu [4]

[1] School of Measurement-Control Technology and Communications Engineering, Harbin University of Science and Technology, Harbin 150080, China
[2] Heilongjiang Network Space Research Center, Harbin 150090, China
[3] Harbin Vocational College of Science and Technology, Harbin 150399, China
[4] Harbin Meteorological Bureau, Harbin 150028, China
* Correspondence: lvxinmiao0611@hrbust.edu.cn; Tel.: +86-187-0462-8841

Abstract: Aiming at the problems of nonlinearity and serious confusion of fault characteristics in analog circuits, this paper proposed a fault diagnosis method for an analog circuit based on ensemble empirical pattern decomposition (EEMD) and improved multifractal detrended fluctuations analysis (MF-DFA). This method consists of three steps: preprocessing, feature extraction, and fault classification identification. First, the EEMD decomposition preprocesses (denoises) the original signal; then, the appropriate IMF components are selected by correlation analysis; then, the IMF components are processed by the improved MF-DFA, and the fault feature values are extracted by calculating the multifractal spectrum parameters, and then the feature values are input to a support vector machine (SVM) for classification, which enables the diagnosis of soft faults in analog circuits. The experimental results show that the proposed EEMD-improved MF-DFA method effectively extracts the features of soft faults in nonlinear analog circuits and obtains a high diagnosis rate.

Keywords: ensemble empirical pattern decomposition (EEMD); multifractal; detrended fluctuations analysis (DFA); support vector machines (SVM); circuit fault diagnosis

1. Introduction

At present, although the theory and application of digital signals is closely related to the rapid development of electronic technology, analog signals still occupy an important position. The proportion of analog and digital circuits in hybrid circuits is about 60% and will continue to grow, the data show. Although analog components account for only 20% of hybrid circuits, the cost of testing and diagnosing analog component has reached 80% of the total cost [1]. There are two types of analog circuit fault modes: hard fault and soft fault. Hard faults are catastrophic faults such as an open circuit and short circuit, which are easy to identify, and soft faults are when the parameters of circuit components constantly change and deviate from their nominal values. In the event of a soft fault, the topology of the circuit will not change, but it will affect the performance of the circuit. If the soft fault is not fixed in time, it becomes a hard fault [2]. If so, it can cause a lot of damage to the circuit and even threaten personal safety.

In order to identify soft faults in analog circuits, various diagnostic methods such as the fault dictionary method and the parameter identification method have been proposed by scholars. However, these methods are applicable to only single faults and are much more computationally intensive for diagnosing circuits dealing with multiple faults. With the development of fault diagnosis technology, linear extraction methods such as principal element analysis and factor analysis are proposed successively. These methods are more effective for linear circuits, but most circuits in daily life are nonlinear and the above methods cannot reflect the nonstationary characteristics of signals, which leads to low separability of the extracted fault features, which leads to large classification error

in fault pattern recognition. For this reason, scholars have proposed time–frequency domain analysis methods, such as SVM, principal component analysis (PCA), and singular value decomposition (SVD), but each of the above time–frequency domain methods has limitations and shortcomings; for example, the typical fault characteristic parameters of SVM are difficult to choose; PCA is a linear algorithm that cannot accommodate real processes with varying degrees of nonlinearity; the performance of SVD mainly depends on the selection of embedding dimensions; however, the current embedding dimensions are mostly selected empirically and poorly adapted [3–5]. In recent years, along with the development of artificial intelligence, more and more artificial intelligence-based methods are applied to the field of circuit fault diagnosis; for example, the literature [6] used artificial neural networks to train and model circuit fault samples, thus improving the circuit fault diagnosis accuracy and shortening the diagnosis time, but artificial neural networks require a large amount of experimental data to form an accurate classification of fault phenomena, while the algorithm converges slowly and is prone to local minima; the literature [7] applied a deep belief network method in deep learning for circuit fault feature extraction. Although it improved the analog circuit fault diagnosis rate, the deep learning model lacks theoretical support in parameter selection and the fault sample library is not rich enough, so there is still much room for improvement in deep learning in nonlinear analog circuit diagnosis. Due to the increasing complexity of electronic systems and the strong coupling of operating states, which leads to the complexity and diversity of their fault patterns, and sometimes even chaos, the traditional theories and methods to portray this complexity increasingly feel the limitations of the theory itself. The fractal theory developed in recent years has provided a good idea for solving this problem. Sheng et al. [8] combined local characteristic-scale decomposition with fractal theory to effectively extract fault features of analog circuits and achieve fault diagnosis of analog circuits. Yao et al. [9] used fractal theory and probabilistic neural networks to implement fault diagnosis of miniature circuit breakers. Mao et al. [10] proposed the method of combining support vector machines with fractal theory and obtained good results. However, the fractal in the above methods is a single fractal. A single fractal can only describe the fractal characteristics of the signal from one perspective, which is not comprehensive enough to describe the fractal characteristics of the signal and cannot portray the local fractal characteristics of the signal. In fault diagnosis, it is difficult to achieve accurate classification and identification once there are intervals where the dimensions of different state signals overlap. The improved MF-DFA method proposed in this paper analyzes from multiple scales to comprehensively respond to the distribution of nonuniformity and probability measures of the structure of fractal objects and provides a more refined description of local fractal features. At the same time, because the actual measured signals are often accompanied by abnormal events, such as noise effects, intermittent signals, etc., leading to the occurrence of the phenomenon of modal aliasing, which has a great impact on the subsequent fault diagnosis rate, this paper thus proposes the EEMD method to preprocess the signal, which can effectively avoid the phenomenon of modal aliasing.

In summary, an analog circuit fault signal state identification method based on EEMD and improved MF-DFA was proposed in this paper. Firstly, EEMD of the circuit fault signal was carried out to obtain several IMF components with no mode overlap; then, the IMF components with circuit fault signal characteristics were selected to reconstruct the fault signal according to the circuit fault characteristics. Later, the fault signal was analyzed by the improved MF-DFA, the Hurst exponent and multifractal spectrum of the fault signal were obtained, and the multifractal spectrum width, extreme value, dimension difference and generalized Hurst exponent fluctuation mean of fault signal were calculated to the characteristic parameters of circuit fault state. Finally, the feature parameters extracted from the improved MF-DFA were classified by SVM to realize the diagnosis of the analog circuits' soft fault.

The rest of this paper is organized as follows: Section 2 presents the signal preprocessing method (EEMD). The signal feature extraction method (improved MF-DFA) is proposed

in Section 3. Section 4 contains the experimental simulations and results. Section 5 contains the conclusion.

2. EEMD

EMD is an effective method for processing circuit fault signals. EMD has the adaptive property of delocalizing time-varying features, which can decompose the signal into the sum of several IMF and reduce the coupling of feature information between signals. By decomposing the signal by EMD method, we can grasp the characteristic information of the original data accurately and efficiently, which is helpful for the mining of information characteristics [11]. However, EMD leads to modal confusion in circuit signal decomposition. To this end, Wu et al. [12] proposed a new method, namely EEMD, in which the unstable signal is decomposed by EEMD to obtain a stationary IMF, effectively avoiding modal confusion.

The essence of EEMD is to superimpose circuit fault signal with Gaussian white noise, and then to EMD decompose it several times, using Gaussian white noise with statistical properties of uniform frequency distribution to make the circuit fault signal have continuity at different scales, thus reducing the degree of mode confounding of each IMF component [13]. In EEMD decomposition, the decomposed circuit fault signal components are averaged through several iterations, and the more the averaging process, the less noise interferes with the decomposition results and the more useful signals can be highlighted. Set the average number of times the original fault signal is processed to N, initially $1, 2, \ldots, N$. The decomposition of EEMD is as follows:

(1) Gaussian white noise $\rho_i(t)$ is added to the original circuit fault signal $x(t)$ to obtain the signal $x_i(t)$:

$$x_i(t) = x(t) + \rho_i(t) \tag{1}$$

(2) The fault signal components IMF at different scales are obtained by EMD decomposition:

$$x_i(t) = \sum_{i=1}^{n} IMF_{i,n}(t) + r_{i,n}(t) \tag{2}$$

where n is the number of IMFs after EMD decomposition; $r_{i,n}(t)$ is the residual component.

(3) $x(t)$ adds different Gaussian white noise and repeats steps (1) and (2), N is the number of clusters, so when the number of noise added is equal to the number of clusters, then the iteration of the algorithm is stopped. The results of EEMD decomposition are as follows:

$$x(t) = \sum_{k=1}^{n} IMF_{k,n}(t) + r_{k,n}(t) \tag{3}$$

(4) Averaging of the IMF components after EEMD decomposition to eliminate noise interference with the signal:

$$IMF_n(t) = \frac{1}{N} \sum_{K=1}^{N} IMF_{kn}(t) \tag{4}$$

The error between the reconstructed signal obtained using the sum of $IMF_n(t)$ and the original signal satisfies the following equation:

$$\omega_n = \frac{\omega}{\sqrt{N}} \tag{5}$$

N is the average number of times EEMD; ω is the standard deviation of the added white noise; and as can be seen from the equation above, ω_n decreases with the increase of the total N. It is known from several tests that the decomposition effect is best when ω is 0.2 and N is 100.

The flow chart of EEMD signal decomposition is shown in Figure 1.

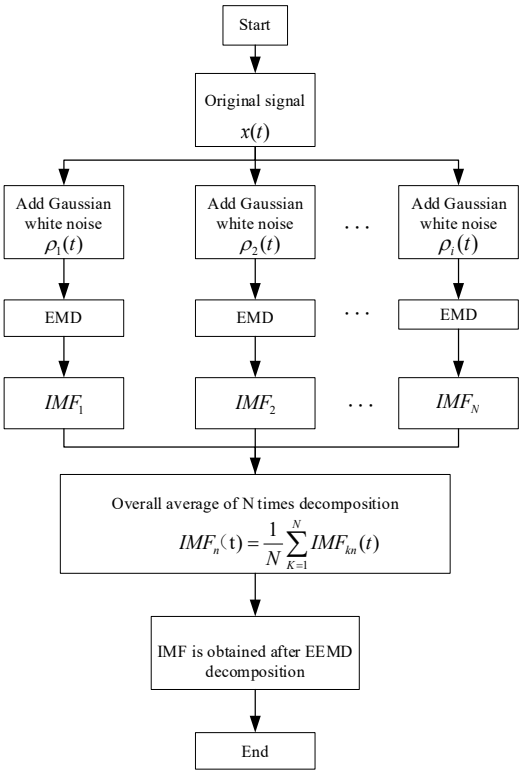


Figure 1. EEMD signal decomposition flow chart.

As can be seen from Figure 1, after the input excitation signal, we first add Gaussian white noise to the original signal several times, then we separately perform EMD decomposition on the signal after adding Gaussian white noise, and finally we statistically average the IMF components obtained after EMD decomposition to obtain the final IMF components. In order to eliminate the effect of noise confusion, we performed the above process for each circuit signal.

3. Improved MF-DFA

The MF-DFA method is a nonstationary time series multiple fractal characterization method based on the detrended fluctuation analysis (DFA) method proposed by Kantelhard J. w. et al. This method firstly eliminates the influence of the nonstationary trend of fault signal sequence by de-volatility process and then describes the fractal structure of time series in different dimensions by calculating the volatility functions of different order. Compared to the DFA method, the MF-DFA method avoids human-induced effects and can more accurately characterize the multiple fractal features hidden in non-stationary time series [14]. For a circuit fault signal sequence $x(n)$ of length N, the steps for MF-DFA are as follows:

(1) Calculate the cumulative deviation $y(n)$ of the fault signal sequence $x(n)$:

$$y(n) = \sum_{i=1}^{n}(x(i) - \overline{x}), n = 1, 2, \ldots, N \tag{6}$$

where $x(n)$ is the original fault signal sequence and $\overline{x}$ is the average value of the signal, $\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x(i)$.

(2) The traditional MF-DFA method uses the equal sequence partitioning method, which divides the cumulative deviation sequence $y(n)$ into m equal-length sequences without overlapping each other on average, where the data length of each subsequence is s. When N cannot divide s integerly, to avoid data remainder, the sequence is partitioned

again from the end of the sequence, and finally 2*m* equal-length subsequences are obtained. This leads to data discontinuity at the segmentation point of the fault signal subsequence, generating new pseudo-volatility errors, and there is a risk of losing the data at the end of the circuit fault sequence or disrupting the order of the original fault signal sequence in the process of constructing the subsequence, resulting in distortion of the estimated generalized Hurst index. In order to overcome the shortcoming of the traditional MF-DFA method, the sliding window technique is introduced to improve the MF-DFA sequence segmentation method. The principle of the sliding window segmentation method shown in Figure 2.

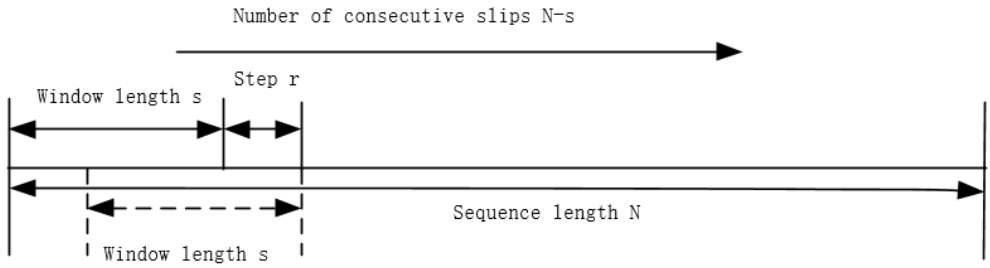


Figure 2. Sliding window principle.

In order to minimize the data discontinuity at the segmentation point of the fault signal subsequence and the risk of losing the data at the end of the circuit fault sequence or disrupting the order of the original fault signal sequence in the process of constructing the subsequence, this paper uses an overlapping window for segmenting the data sequence.

The sliding window overlap segmentation method results in an increase in the segmented subsequences from *m* or 2*m* to N-s + 1 (to avoid remaining data at the end, step $r = 1$).

(3) The local trend function of each subsequence is fitted with least-squares:

$$y_v(n) = a_0 + a_1 n + a_2 n^2 + \ldots + a_k n^k, n = 1, 2, \ldots, 2m; k = 1, 2, \ldots \tag{7}$$

where a_k is the coefficient of the polynomial fit and k is the order of the polynomial fit.

The polynomial fitting process is prone to underfitting and overfitting problems. Underfitting occurs generally because the complexity of the model used is too low and the amount of features used is too small. In this paper, because the circuit model is fixed, the number of features in the sample is increased by polynomial regression in this paper so as to solve the error problem caused by overfitting. Overfitting occurs due to a single sample of training data set, insufficient samples, and too much noise interference in the training process, so this paper selects data samples with positive and negative deviations to ensure the diversity of samples, while using the EEMD method to denoise the samples, which greatly avoids the test error caused by overfitting.

(4) Calculate the mean square error function when $v = 1, 2, 3, \ldots, m$, with

$$F^2(s, v) = \frac{1}{s} \sum_{n=1}^{s} \{y[(v-1)s + n] - y_v(n)\}^2 \tag{8}$$

When $v = m + 1, m + 2, \ldots, 2m$,

$$F^2(s, v) = \frac{1}{s} \sum_{n=1}^{s} \{y[N - (v-1)s + n] - y_v(n)\}^2 \tag{9}$$

(5) Traditional MF-DFA calculates the q-rung fluctuation function:

$$F(q, s) = \left\{ \frac{1}{2m} \sum_{v=1}^{2m} \left[F^2(s, v) \right]^{q/2} \right\}^{1/q}, q \neq 0 \tag{10}$$

$$F(q,s) = \exp\left\{\frac{1}{4m}\sum_{v=1}^{4m}\ln\left[F^2(s,v)\right]\right\}, q = 0 \tag{11}$$

Improved MF-DFA to calculate the q-rung fluctuation function:

$$F(q,s) = \left\{\frac{1}{N-s+1}\sum_{v=1}^{N-s+1}\left[F^2(s,v)\right]^{q/2}\right\}^{1/q}, q \neq 0 \tag{12}$$

$$F(q,s) = \exp\left\{\frac{1}{2(N-s+1)}\sum_{v=1}^{N-s+1}\ln\left[F^2(s,v)\right]\right\}, q = 0 \tag{13}$$

To ensure high stability of the fluctuation function $F(q,s)$, the fixed window length is $2k + 2 < s < int\ (N)$ (in which k is the polynomial fitting order).

(6) the order q takes a certain value in turn, the scale s takes different values, and repeat steps (2)–(5) to calculate the double logarithmic value of $F(q,s)$ for q.

(7) The fluctuation function is a function on the data length q and the subsequence data length s, which is related to s as shown below:

$$F(q,s) \propto s^{h(q)} \tag{14}$$

(8) The relationship between generalized Hurst index $h(q)$ and scalar index $\tau(q)$ obtained by MF-DFA method is as follows:

$$\tau(q) = qh(q) - 1 \tag{15}$$

Both sides of Equation (15) are simultaneously derived for q:

$$\frac{d\tau(q)}{dq} = h(q) + qh'(q) \tag{16}$$

The relationship between the multiple fractal spectrum $f(a)$, the singularity index a and the scalar index $\tau(q)$ is obtained by the Lejeune transform:

$$\begin{cases} a = \frac{d\tau(q)}{dq} \\ f(a) = aq - \tau(q) \end{cases} \tag{17}$$

Substituting Equation (15) into Equation (16) yields the following relationship:

$$\begin{cases} a = h(q) + qh'(q) \\ f(a) = q[a - h(q)] + 1 \end{cases} \tag{18}$$

The generalized Hurst index reflects the relationship between the different scales of circuit signal and is a characteristic measure of the multifractal characteristics of circuit signal. In a coordinate system with q as horizontal and $h(q)$ as vertical, if q and $h(q)$ are a nearly smooth straight line, the circuit signal has no multifractal characteristics; if q and $h(q)$ are a non-linear decreasing function, the circuit signal has multifractal characteristics, and the more $h(q)$ fluctuates, the stronger the multifractal characteristics of the circuit signal. Therefore, the generalized Hurst exponent fluctuation mean is taken as one of the fault's characteristics.

The width of the multifractal spectrum Δa is the difference between the strongest and weakest singularities of the signal, reflecting the strength of the multifractal characteristics of the circuit signal; the stronger the multifractal characteristics, the larger the value of Δa; for the singularity index $a_0(f_{\max} = f(a_0))$ corresponding to the extreme value point, which reflects the randomness of the circuit signal, the greater the randomness, the greater the a_0; the difference between the fractal dimension of the maximum and minimum probability subsets $\Delta f = f(a_{\max}) - f(a_{\min})$, reflecting the change in the maximum and

minimum peak frequencies of circuit signal, which is less than 0, indicating that the number of maximum probability subsets is greater than the number of minimum probability subsets, and vice versa. Through the above analysis, it can be seen that the multiple fractal spectrum parameters can reflect the intrinsic characteristics of a fault signal in different states quantitatively; therefore, four characteristic parameters are proposed as circuit fault state identification parameters.

As can be seen from Figure 3, after the original signal is processed by EEMD, a number of IMF components are obtained. The IMF components without mode overlap are selected by correlation calculation. Then, we calculate the cumulative deviation of these IMF components, and then cut the cumulative deviation sequence by sliding window to obtain N-S + 1 subseries, and then calculate the mean square error function and q-rung fluctuation function by polynomial fitting of these subseries, and finally obtain the multifractal spectrum of the fault signal by the q-rung fluctuation function, so as to calculate the multifractal spectrum parameters as eigenvalues into the support vector machine.

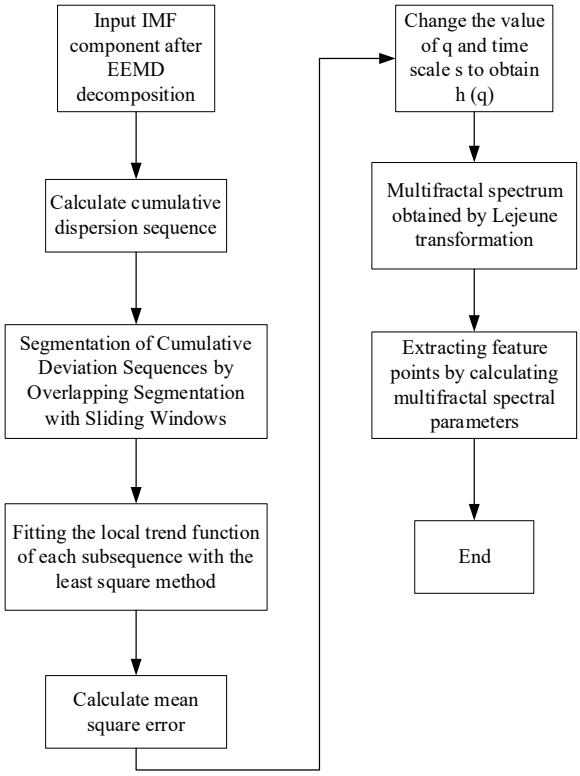


Figure 3. Flow chart of improved MF-DFA algorithm.

4. Simulation Experiment

4.1. Experimental Process

Figure 4 provides the flow chart of analog circuit fault diagnosis based on the EEMD-improved MFDFA.

The steps of the EEMD-improved MFDFA-based analog circuit fault diagnosis are as follows:

(1) Apply an excitation signal to the circuit to be tested, select the fault component by sensitivity analysis, and collect the output response under different fault modes for signal preprocessing.

(2) EEMD decomposition is performed on the acquired signal data to obtain each IMF component.

(3) Each IMF component is processed by improved MF-DFA to obtain the width, extreme value, dimensional difference and generalized Hurst index fluctuation mean of the multiple fractal spectrum of the fault signal, which constitutes the fault feature vector.

(4) The SVM is extended with a "one-to-many" approach for multi-classification.

(5) The training samples are input to the classifier for training, and the trained classifier is used to classify and diagnose the fault data.

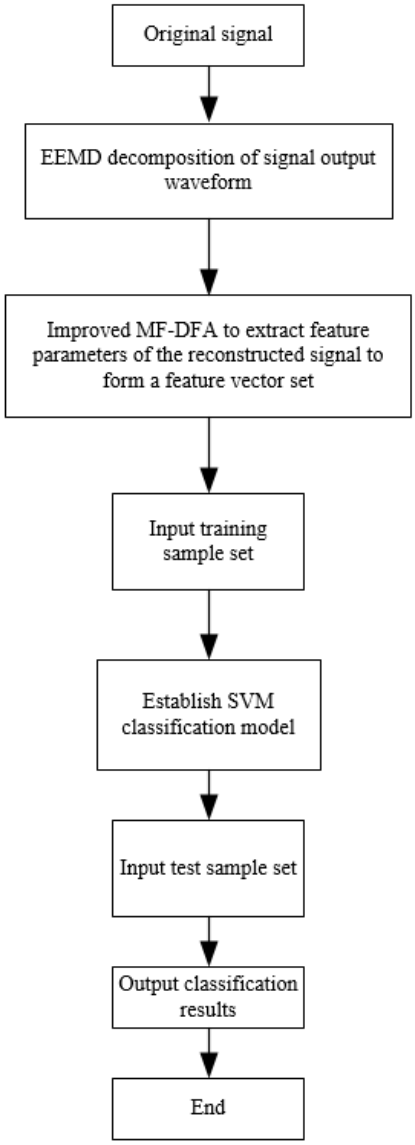


Figure 4. Basic flow chart for soft fault diagnosis of analog circuits.

4.2. Experimental Environment

In order to verify the validity of the experimental method, a circuit model was established in NI Multisim 14.0 using the bandpass filter circuit in Figure 5 as a diagnostic example, and data were processed with MATLAB R2021b. The parameters of the circuit components are shown in Figure 5. Resistance tolerance was set at 5% and capacitor tolerance at 10%, and the parameters of each element were considered to be in normal condition in this range, with more than 30% considered to be fault state.

The excitation signal for this experiment was $f(t) = \sin(800\pi t) + \sin(\sin(1000\pi t)) + \sin(\sin(2800\pi t)) + \text{noise}$. A total of ten kinds of parameter-type fault combinations were configured. The failure of amplifier is affected by many factors, so it is not discussed here. The fault mode settings are shown in Table 1.

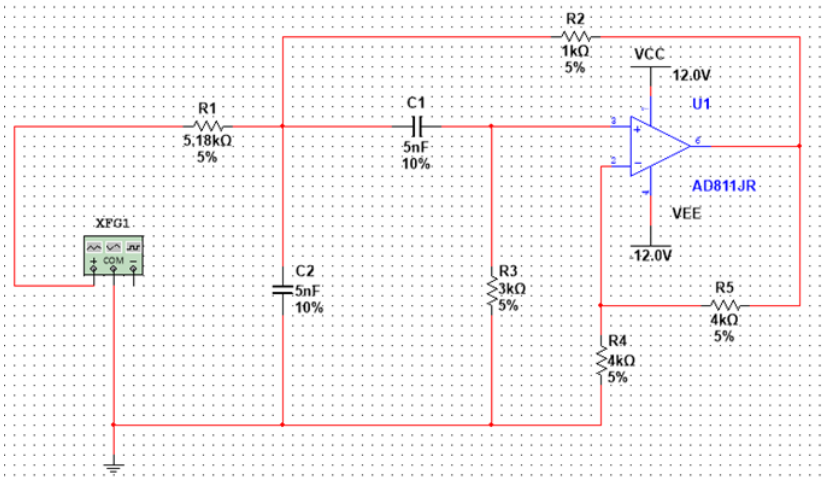


Figure 5. Bandpass filter circuit.

Table 1. Circuit failure modes of bandpass filters.

Fault Code	Fault State	Nominal Value	Fault Value
1	Normal	-	-
2	R_2↑30%	1 kΩ	1.3 kΩ
3	R_2↓30%	1 kΩ	0.7 kΩ
4	R_3↑30%	3 kΩ	3.9 kΩ
5	R_3↓30%	3 kΩ	2.1 kΩ
6	C1↑30%	5 nF	6.5 nF
7	C1↓30%	5 nF	3.5 nF
8	C2↑30%	5 nF	6.5 nF
9	C2↓30%	5 nF	3.5 nF

4.3. Experimental Result

Multisim simulation software was used to analyze the circuit instantaneously and sample the output response to obtain the original fault information. Fifty Monte Carlo analyses were performed for each type of the fault in Table 1, and the samples under each failure mode were randomly divided into two parts, with 30 sets of data participating in training and the other 20 sets of data for testing.

All sample data were decomposed by EEMD, and the results are provided in Figures 6 and 7 for the normal mode and the fault 7 mode. By comparing Figures 6 and 7, it can be seen that the fault information is mainly concentrated in the first three IMF components, so the first three components are selected for feature parameter extraction.

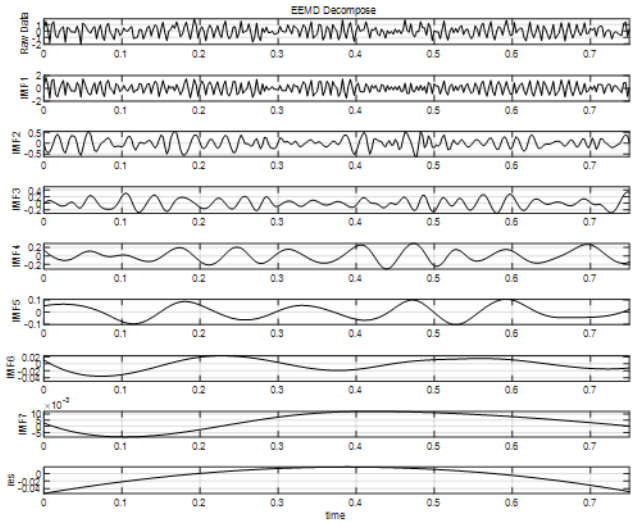


Figure 6. EEMD decomposition of the signal in normal condition.

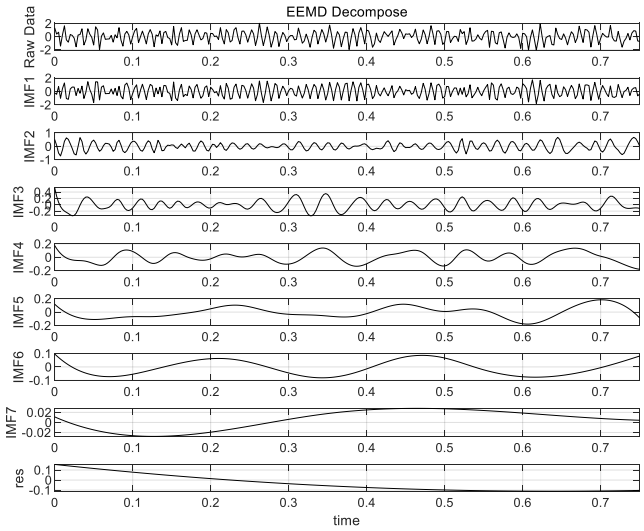


Figure 7. EEMD decomposition of the signal corresponding to fault 7.

The following is an example of fault 1, fault 2, fault 4, fault 6, and fault 8, and the number of samples for each state was selected as 50 for a total of 150 groups of samples. The improved MF−DFA was applied to the IMF components, and the order q range was taken as $[-5, 5]$, and a total of 11 q values were taken. The Hurst index curve and the multifractal spectrum were obtained, as shown in Figures 8 and 9. As can be seen from Figure 8, the generalized Hurst exponents of five signals is a curve of about q, indicating that the circuit signals of these five circuit states have multiple fractals characteristics. It can be seen from Figure 9 that the singular spectra of the five signals have different shapes, positions, and spectral widths. Based on the characteristic of singularity spectrum, the multiple fractal features of signal can be extracted further, and the fault diagnosis can be carried out effectively.

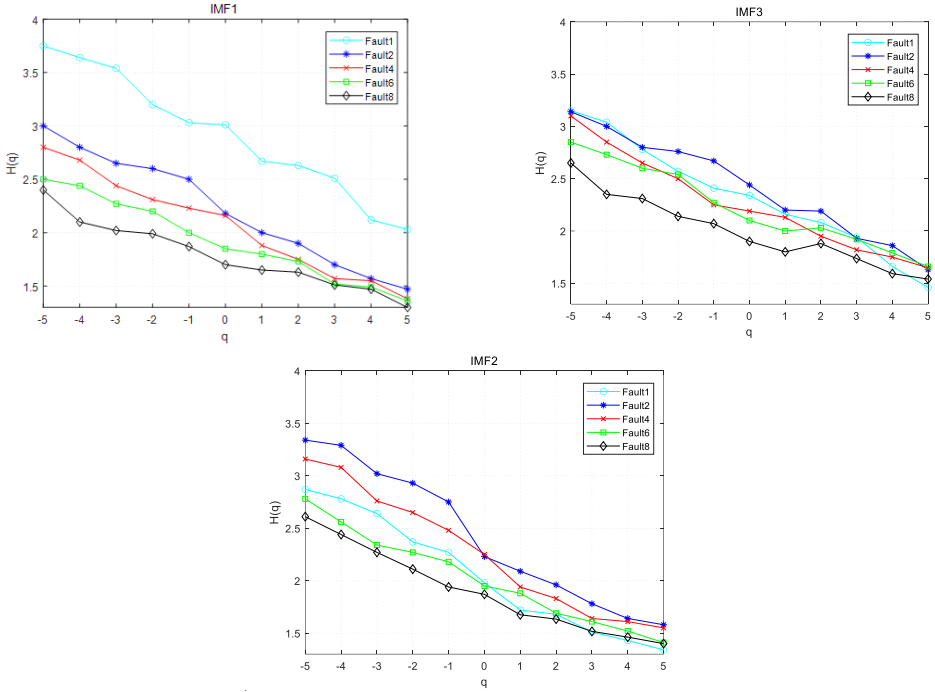


Figure 8. Hurst curves under five signals based on improved MF−DFA.

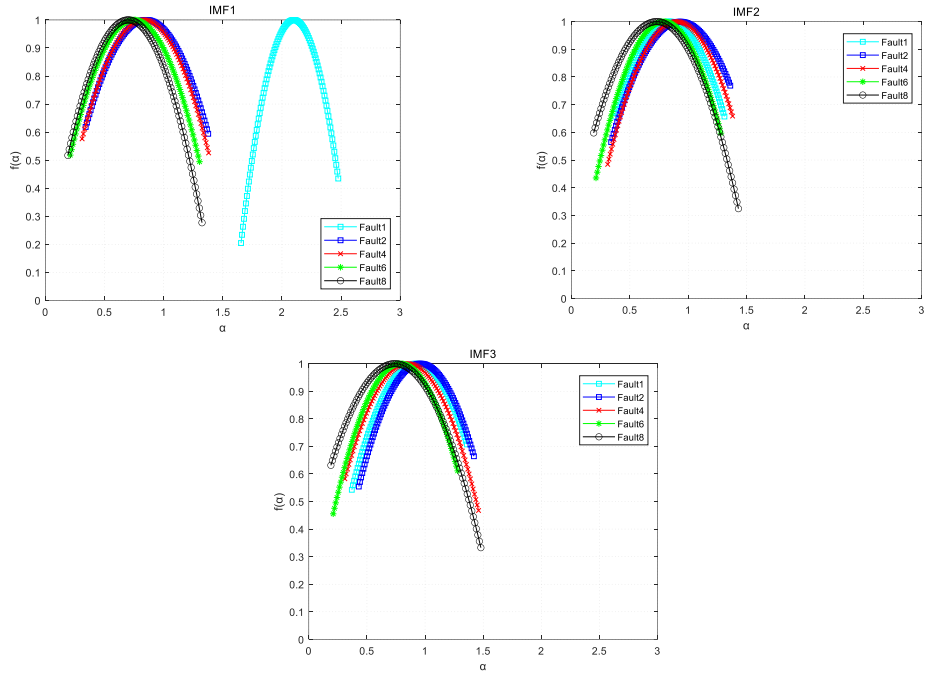


Figure 9. Multiple fractal spectral curves under five signals based on improved MF−DFA.

To further highlight the advantages of the improved MF−DFA, the traditional MF-DFA was used in this paper to process the sample signals in five states under three IMF components, with scalar values and order q being the same as the improved MF−DFA settings, resulting in the Hurst curve, and the results are shown in Figure 10.

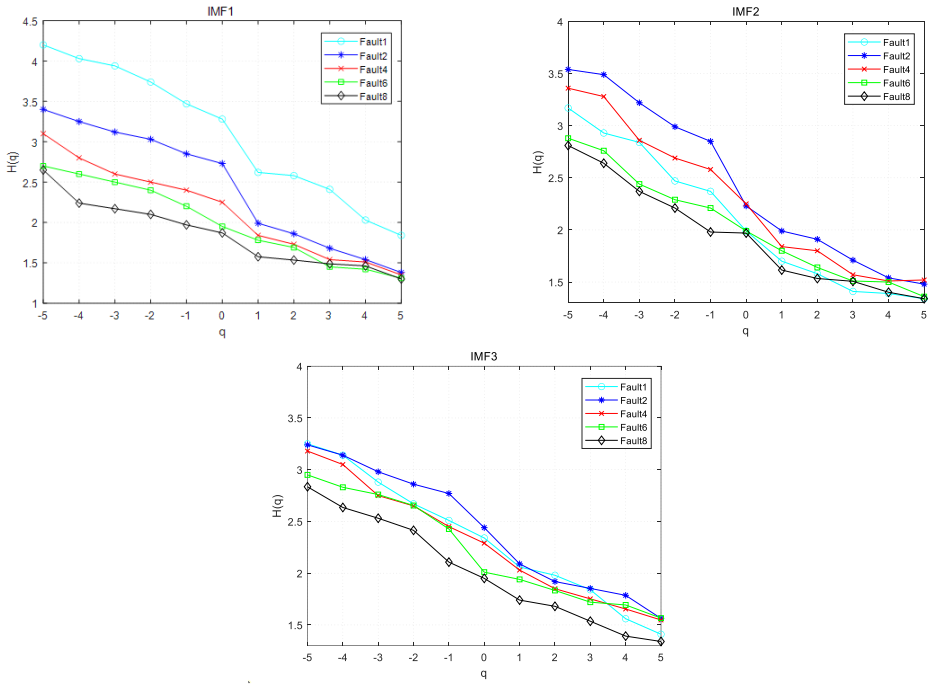


Figure 10. Five signal Hurst curves based on conventional MF−DFA.

From Figures 8 and 10, it can be seen that the generalized Hurst curve index decreases with increasing value of q for both MF−DFA and improved MF−DFA methods, indicating the existence of irregular multiple fractal characteristics of the circuit signal. At $q < 0$, the $h(q)$ value of the improved MF−DFA method is smaller than that of the conventional

method, while at $q > 0$, the $h(q)$ value of the improved MF−DFA method is greater than that of the conventional method, suggesting that the improved MF−DFA method can effectively reduce the pseudorandom fluctuation error caused by the discontinuity of time series segmentation point, while avoiding tail data loss and sequence reconstruction errors of the conventional method.

The average values of the multifractal spectral parameters for each of the five states of the circuit were calculated using the improved MF−DFA method, as shown in Table 2. Meanwhile, the distributions of Δa, a_0, Δf, and $\Delta\overline{h}(q)$ for the five states are shown in Figures 11–14. From the figure, it can be seen that Δf and $\Delta\overline{h}(q)$ have subtle crossover in different states, while Δa and a_0 were mixed severely, so Δf and $\Delta\overline{h}(q)$ were selected to form the eigenvectors in this paper.

Table 2. Multifractal spectral parameter averaging based on improved MF−DFA.

	Fault Code	Δa	a_0	Δf	$\Delta\overline{h}(q)$
IMF1	1	0.956	2.098	0.809	2.947
	2	0.996	0.815	0.386	2.315
	4	1.103	0.752	0.474	2.085
	6	1.201	0.697	0.641	1.929
	8	1.134	0.636	0.714	1.713
IMF2	1	1.083	0.832	0.324	2.179
	2	1.036	0.932	0.432	2.412
	4	1.072	0.894	0.531	2.297
	6	1.087	0.771	0.645	2.031
	8	1.155	0.694	0.725	1.765
IMF3	1	1.021	0.913	0.321	2.419
	2	1.132	0.986	0.431	2.534
	4	1.154	0.837	0.590	2.274
	6	1.076	0.759	0.521	2.116
	8	1.207	0.709	0.708	1.878

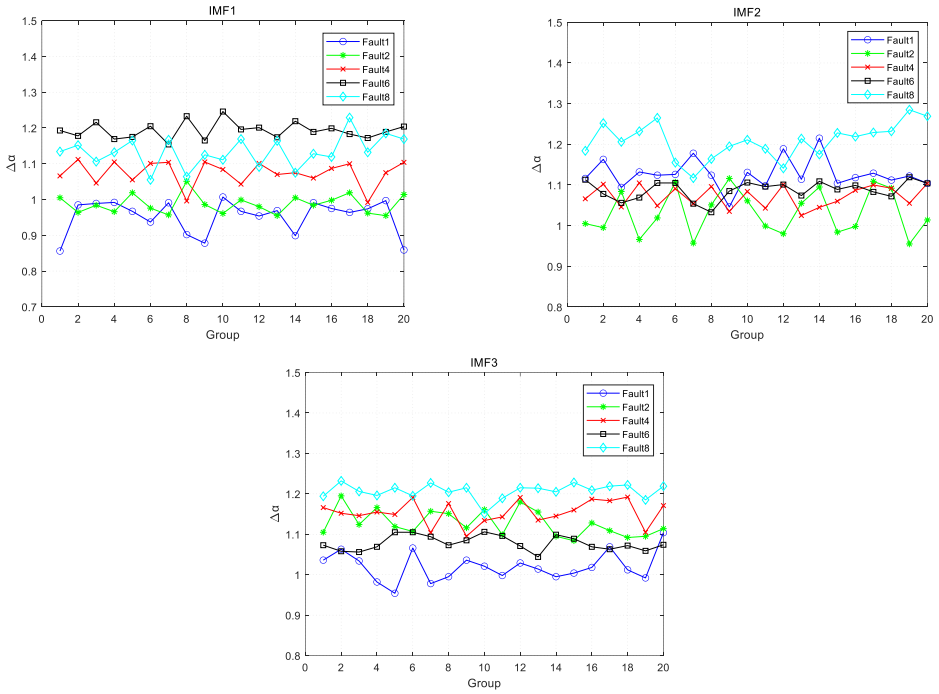


Figure 11. Distribution of Δa in five fault modes.

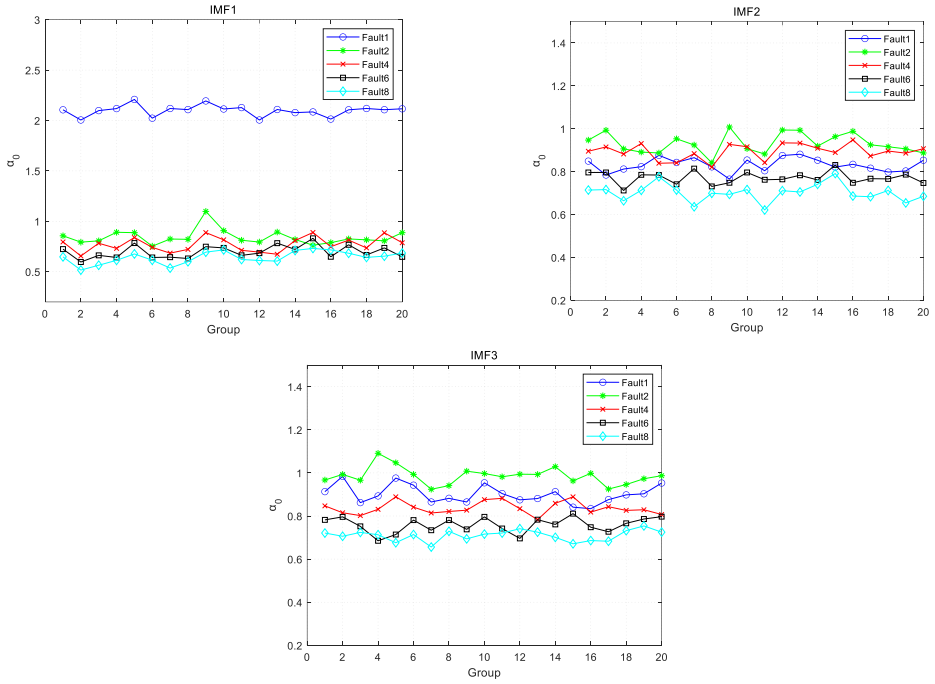


Figure 12. Distribution of a_0 in five fault modes.

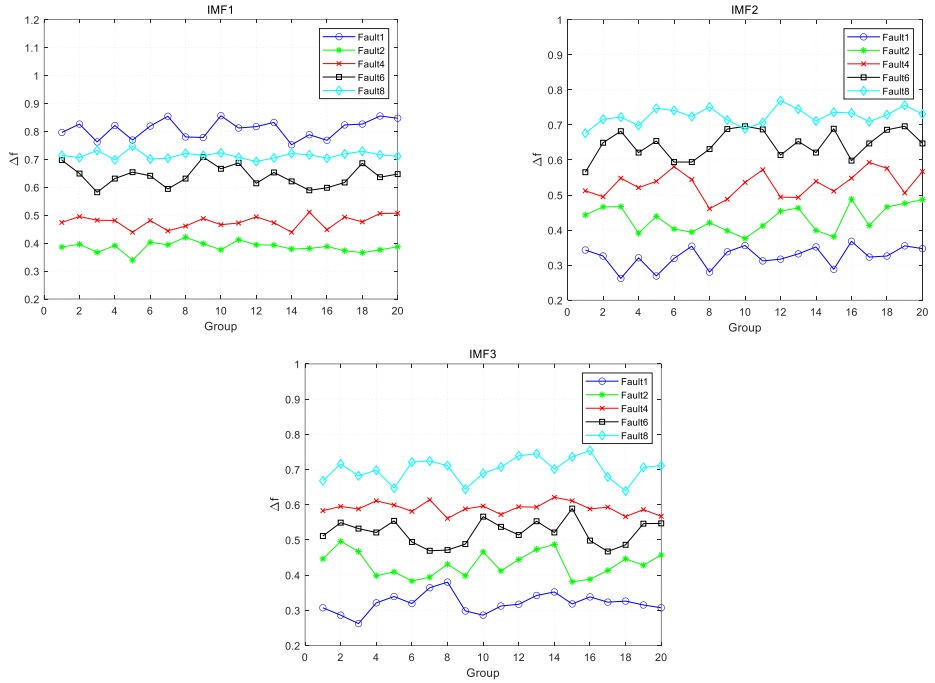


Figure 13. Distribution of Δf in five fault modes.

To further verify the effect of the proposed feature extraction method, SVM was used to model and classify the extracted features. Each state had 30 samples for SVM training and the remaining 20 samples for testing. We also compared with the wavelet packet energy spectrum and ICA (WP−ICA)-based analog circuit fault feature extraction method from the literature [15] and the EEMD multiscale entropy and LSSVM (EEMD−LSSVM) in the analog circuit fault diagnosis method from the literature [16]; the diagnosis results of the three methods were obtained, as shown in Table 3.

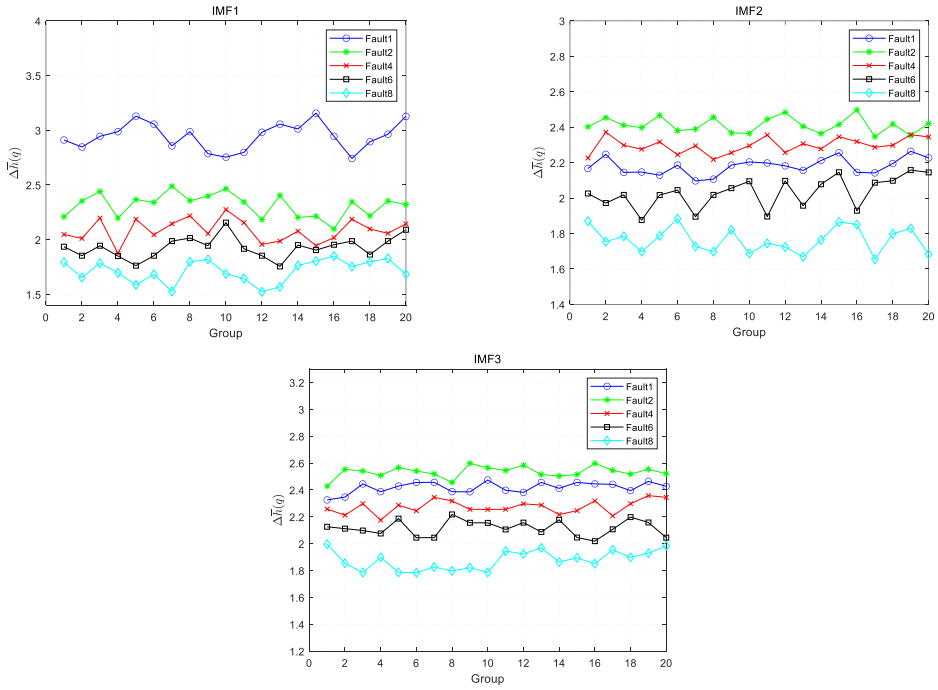


Figure 14. Distribution of $\Delta\overline{h}(q)$ in five fault modes.

Table 3. Fault diagnosis results of three methods.

	WP−ICA	EEMD−LSSVM	EEMD−Improved-MFDFA
Average detection rate/%	95.7	96.67	97.87

In this paper, Δf and $\Delta\overline{h}(q)$ were input to the support vector machine as feature vectors for classification. The classification results are shown in Table 3. In the same circuit environment, the recognition rate of the proposed method in this paper can reach 97.87%, while the recognition rates of the other two methods are 95.7% and 96.67%. This shows that the method proposed in this paper has more advantages in circuit soft fault identification.

In order to better verify the effectiveness of the method in this paper, the method in this paper is used in the fault feature extraction of three different circuits, namely, Sallen–Key lowpass filter, state-variable filter, and active dual second-order bandpass filter, and compared with the CBAM−CNN analog circuit fault diagnosis method from the literature [17], the FFT−CNNN−GRU analog circuit fault diagnosis method from the literature [18], and the MLLE−CHMM analog circuit fault diagnosis method from the literature [19]. The diagnosis results are shown in the following table.

From Table 4, it can be seen that the method in this paper is still very effective in circuit fault feature extraction even under different circuit models, while the diagnosis rate shows that the method in this paper is more advantageous than other methods with higher recognition rate.

Table 4. Diagnostic results of the corresponding methods under the three circuit models.

Circuit Models	Method	Diagnosis Rate (%)
Sallen–Key lowpass filter	EEMD-Improved MF-DFA	96.31
	CBAM-CNN	95.48
State-variable filters	EEMD-Improved MF-DFA	97.785
	FFT-CNN-GRU	97.778
Active dual second-order bandpass filter	EEMD-Improved MF-DFA	96.32
	MLLE-CHMM	95.04

5. Conclusions

Aiming at the difficulty of analog circuit fault diagnosis feature extraction caused by the nonlinearity and non-smoothness of analog circuit fault signal, an improved MFDFA fault diagnosis method based on EEMD was proposed. In order to avoid the problem of signal confusion, EEMD decomposition was used to preprocess each fault signal to obtain three IMF components without mode confusion, and then the improved MF-DFA processing of the obtained IMF components to obtain the multifractal spectrum. Because the multifractal spectra corresponding to different fault features have different shapes, distributions, and pole positions, they can be used to characterize different types of circuit faults. Subsequently, this paper used the Δf and $\Delta\bar{h}(q)$ obtained from the multiple fractal spectrum to form the feature vectors, and finally the support vector machine analysis method was used to classify the fault features. The results proved that the proposed EEMD-improved MFDFA-based fault feature extraction method showed good performance.

Author Contributions: Conceptualization, X.L.; methodology, X.L.; software, S.S.; validation, X.L., K.L., Q.W. and D.S.; writing—original draft preparation, C.Y., Z.L. and J.W.; writing—review and editing, C.Y.; project administration, X.L.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014.

Acknowledgments: The authors acknowledge Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khemani, V.; Azarian, M.H.; Pecht, M.G. Learnable Wavelet Scattering Networks: Applications to Fault Diagnosis of Analog Circuits and Rotating Machinery. *Electronics* **2022**, *11*, 451. [CrossRef]
2. El-Gamal, M.A.; Hassan, A.K.S.O.; Ibrahim, A.A.I. Analog Fault Diagnosis Using Conic Optimization and Ellipsoidal Classifiers. *J. Electron. Test.* **2014**, *30*, 443–455. [CrossRef]
3. Yang, Z.; Tian, Y.; Deng, N. Leave-One-Out Bounds for Support Vector Ordinal Regression Machine. *Neural Comput. Appl.* **2008**, *18*, 731–748. [CrossRef]
4. Wang, Y.P.; Mei, F.; Zheng, J.Y.; Mei, J. Research on Fault Diagnosis of High-Voltage Circuit Breaker in Electrical Power System Based on KPCA. *Adv. Mater. Res.* **2013**, *676*, 269–272. [CrossRef]
5. Cui, L.L.; Liu, Y.H.; Wang, X. Feature Extraction of Weak Fault for Rolling Bearing Based on Improved Singular Value Decomposition. *J. Mech. Eng.* **2022**, *58*, 156–169.
6. Narula, A.K.; Singh, A.P. Fault Diagnosis of Mixed-Signal Analog Circuit using Artificial Neural Networks. *Int. J. Intell. Syst. Appl.* **2015**, *7*, 11–17. [CrossRef]
7. Wang, X.L.; Li, C.; Zhang, C.L. Analog Circuit Fault Diagnosis Method Based on Deep Learning. *Chin. J. Electron Devices* **2019**, *42*, 674–678.
8. Sheng, P.; Xu, A.Q.; Xu, X.W.; Li, H. Fault Diagnosis Method of Analog Circuit Based on LCD and Fractal Dimension. *J. Nav. Aeronaut. Astronaut. Univ.* **2020**, *35*, 101–105+118.
9. Yao, Y.; Wang, N. Fault Diagnosis Model of Adaptive Miniature Circuit Breaker Based on Fractal Theory and Probabilistic Neural Network. *Mech. Syst. Signal Process.* **2020**, *142*, 106772. [CrossRef]
10. Mao, J.K.; Mao, X.B. Application of SVM Classifier and Fractal Feature in Circuit Fault Diagnosis. *Adv. Mater. Res.* **2012**, *490–495*, 942–945. [CrossRef]
11. Chen, K.; Zhou, X.C.; Fang, J.Q.; Zheng, P.; Wang, J. Fault Feature Extraction and Diagnosis of Gearbox Based on EEMD and Deep Briefs Network. *Int. J. Rotating Mach.* **2017**, *2017*, 9602650. [CrossRef]
12. Wu, Z.; Huang, N.E. Ensmble Empirical Mode De-composition: A Noise—Assisted Data Analysis Method. *Adv. Adapt. Data Anal.* **2009**, *1*, 1–41. [CrossRef]
13. Singh, S.K.; Kumar, S.; Dwivedi, J.P. Compound Fault Prediction of Rolling Bearing Using Multimedia Data. *Multimed. Tools Appl.* **2017**, *76*, 18771–18788. [CrossRef]
14. Liu, H.; Jing, J.; Ma, J. Fault Diagnosis of Electromechanical Actuator Based on VMD Multifractal Detrended Fluctuation Analysis and PNN. *Complexity* **2018**, *2018*, 9154682. [CrossRef]
15. Shan, S.S.; Ma, Q.F.; Xie, W.X. Fault Feature Extraction of Analog Circuit Based on Wavelet Packet Energy Spectrum and ICA. *Electron. Meas. Technol.* **2021**, *44*, 19–23.

16. Zhuang, C.C.; Yi, H.; Zhang, J. Application of EEMD Multi-Scale Entropy and LSSVM in Analog Circuit Fault Diagnosis. *Microelectron. Comput.* **2019**, *36*, 78–82.
17. Du, X.J.; Gong, B.; Yu, P.; Shi, Y.K. CBAM-CNN based analog circuit fault diagnosis. *Control Decis.* **2022**, *37*, 2609–2618.
18. Liu, Y.P.; Xu, W.Z.; Yang, Q. Diagnosis Method of Analog Circuit Fault Based on FFT-CNN-GRU. *J. Shenyang Ligong Univ.* **2021**, *40*, 27–30+65.
19. Chen, S.; Zhang, M.L.; Zhang, S.; Qiu, X.X.; Xu, C.W. MLLE-CHMM Based Fault Diagnosis Method for Analog Circuits. *Mod. Ind. Econ. Inf.* **2022**, *12*, 302–304+311.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Tone Mapping Method Based on the Least Squares Method

Lanfei Zhao [1], Guoqing Li [1] and Jun Wang [2,*]

[1] The Higher Educational Key Laboratory for Measuring & Control Technology and Instrumentations of Heilongjiang Province, Harbin University of Science and Technology, Harbin 150080, China
[2] School of Information Engineering, Quzhou College of Technology, Quzhou 324000, China
* Correspondence: 145688@qzct.edu.cn; Tel.: +86-181-0570-1939

Abstract: Tone mapping is used to compress the dynamic range of image data without distortion. To compress the dynamic range of HDR images and prevent halo artifacts, a tone mapping method is proposed based on the least squares method. Our method first uses weights for the estimation of the illumination, and the image detail layer is obtained by the Retinex model. Then, a global tone mapping function with the parameter is used to compress the dynamic range, and the parameter is obtained by fitting the function to the histogram equalization. Finally, the detail layer and the illumination layer are fused to obtain the LDR image. The experimental results show that the proposed method can efficiently restore real-world scene information while preventing halo artifacts. Therefore, tone mapping quality index and mean Weber contrast of the tone-mapped image are 8% and 12% higher than the closest competition tone mapping method.

Keywords: tone mapping; the least squares method; weights; HDR image; Retinex

1. Introduction

The real-world scene contains a wide range of dynamic range differences, the visible light dynamic range can reach 10^{10} : 1, and the dynamic range that can be perceived by the human eye is 10^5 : 1. However, conventional display devices can display a dynamic range of 10^3 : 1. The lack of dynamic range causes the loss of scene details and color information and cannot present the rich information of high dynamic range (HDR) images' content. Thus, a compression of the dynamic range for rendering HDR images on low dynamic range (LDR) devices is needed; this is commonly called tone mapping. In the process of tone mapping operation, it is necessary to retain the details, contrast, and other information of the image.

At present, HDR tone mapping methods are mainly divided into two categories: global tone mapping and local tone mapping. The method of global tone mapping mainly uses a mapping curve to map each pixel and adjusts the input brightness regardless of location; this method is easy to calculate and can maintain a good overall image effect, but it will cause a lot of loss of local details. On the contrary, local tone mapping takes into consideration the neighborhood of a pixel. Local tone mapping can effectively preserve more details of the image and prevent halo artifacts. Therefore, more and more researchers are paying attention to local tone mapping methods. Based on that, this paper proposes a local tone mapping method based on the least squares. The innovation of our method is first to use weights to obtain the detail layer through the Retinex model. The method of using weights for illumination estimation has two advantages. First, it has good edge-preserving and smoothing properties, which can obtain more accurate illumination information. Second, it can ensure that no gradient reversal is caused and prevent the generation of halo artifacts. The specific method is to assume that each pixel is weighted by the other pixels in its neighborhood; the equation listed for all pixel points is solved by the least squares method. The obtained weights are used for the estimation of the illumination, while the improved boundary-aware weights are introduced to prevent halo artifacts. The

results of the illumination estimation are used to obtain the detail layer by the Retinex model. Then, the dynamic range is compressed by a global tone mapping function with a parameter. Different images correspond to different parameters, so this method can achieve dynamic range compression for different images more effectively. Finally, the detail layer is fused with the adjusted illumination layer to obtain the LDR image. This method can compress the dynamic range and render HDR images effectively.

This paper is organized as follows. In Section 2, we briefly introduce the method of tone mapping. In Section 3, we introduce the method proposed in this paper in detail. In Section 4, the comparative tests we made are given. Finally, Section 5 concludes this paper.

2. Background

In recent years, many researchers have proposed different methods to solve the problems in the process of tone mapping. Common global tone mapping methods include histogram equalization, gamma correction, logarithmic correction, and so on. Many people have proposed improved methods of global tone mapping. Khan et al. presented a new histogram-based tone mapping method for HDR images which outperformed some existing state-of-the-art methods in terms of preserving both naturalness and structure [1]. Choi et al. proposed that the combination of the key value of the scene and visual gamma is used to improve the contrast in the entire resulting image as well as to scale the dynamic range using input luminance value, simultaneously [2]. Khan et al. used a histogram of luminance to construct a look-up table (LUT) for tone mapping [3]. Anas et al. proposed a modification of the histogram adjustment-based linear-to-equalized quantizer (HALEQ), developed for HDR images. The proposed modification preserves more details than the original version of the method in most parts of the HDR image [4].

Recent studies focus more on the local tone mapping method. Thai et al. used an adaptive powerful prediction step. The pixel distribution of the coarse reconstructed LDR image is then adjusted according to a perceptual quantizer concerning the human visual system using a piecewise linear function [5]. Yang et al. proposed an efficient method for image dynamic range adjustment with three adaptive steps [6]. Li et al. presented a clustering-based content and color-adaptive tone mapping method. Different from previous methods which are mostly filtering-based, this method works on image patches, and it decomposes each patch into three components: patch means, color variation, and color structure [7]. Thai and Mokraoui's method is a multiresolution approach with approximation and details separated, is weighted by entropy, and used an optimized contrast parameter. Then, the pixel distribution of the coarse reconstructed LDR image is adjusted according to a perceptual quantizer concerning the human visual system using a piecewise linear function [8]. Fahim and Jung proposed a two-step solution to perform a tone mapping operation using contrast enhancement. Our method improves the performance of the camera response model by utilizing the improved adaptive parameter selection and weight matrix extraction [9]. David et al. presented two new TMOs for HDR backward-compatible compression. The first TMO minimizes the distortion of the HDR image under a rate constraint on the SDR layer. The second TMO maintains the same minimization with an additional constraint to preserve the SDR perceptual quality [10].

In recent years, researchers have been focusing on the design of various edge-preserving filters for tone mapping. Liang et al. proposed a hybrid $\ell 1$-$\ell 0$ decomposition model. A $\ell 1$ sparsity term is imposed on the base layer to model its piecewise smoothness property. A $\ell 0$ sparsity term is imposed on the detail layer as a structural prior [11]. Many researchers have designed methods based on deep learning networks. Rana et al. presented an end-to-end parameter-free DeepTMO. Tailored in a cGAN framework, the model is trained to output realistically looking tone-mapped images that duly encompass all the various distinctive properties of the available TMOs [12]. Hu et al. proposed a joint multiscale denoising and tone mapping framework that is designed with both operations in mind for HDR images. The joint network is trained in an end-to-end format that optimizes both operators together [13].

3. Proposed Method

The proposed method includes two parts: illumination layer estimation based on image weights and a global tone mapping function with a parameter. These methods can effectively improve the image quality of the LDR after tone mapping. The flow chart of the method is shown in Figure 1, and its specific steps are as follows.

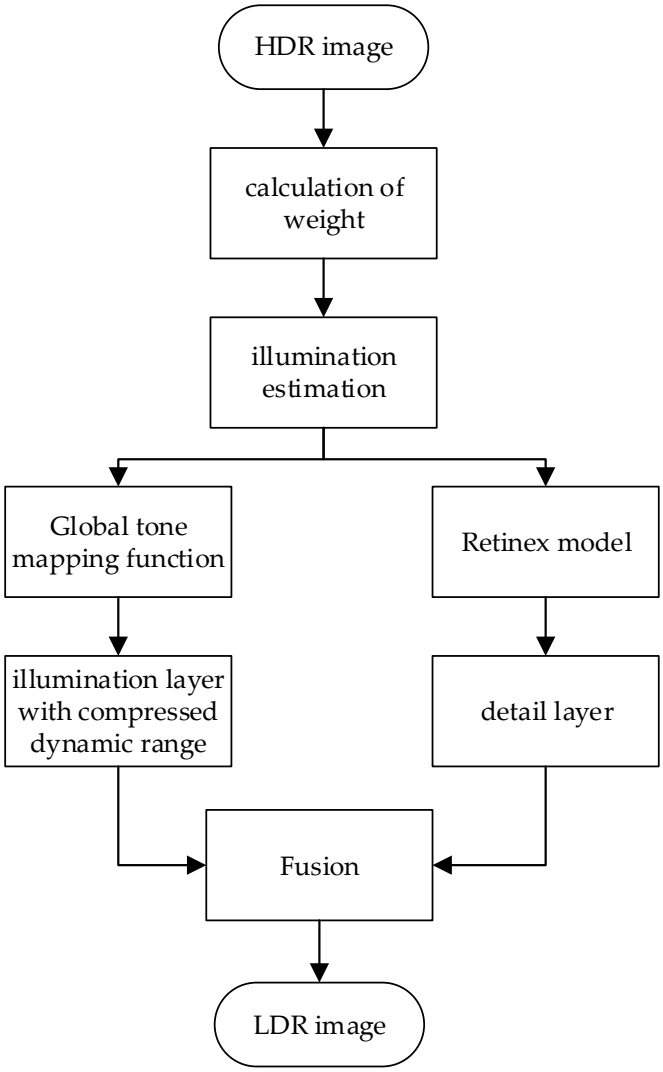


Figure 1. Flow chart of the method in this paper.

1. The weight between the original image pixel points and other pixel points in their neighborhood is obtained by the least squares method, and the boundary-aware weights are introduced to prevent the halo artifacts and are used to estimate the illumination of the original image.
2. The detail layer of the image is obtained by the Retinex model.
3. A global tone mapping function with a parameter is used to process the illumination layer obtained in step (1).

After the above processing, the illumination layer and detail layer are fused to obtain the final LDR image.

3.1. The Illumination Estimation

We chose to obtain the weight of all pixel points in the image with their 3×3 neighborhood pixel points; each region includes nine pixels. Figure 2 shows the weight matrix of a pixel and its 3×3 neighborhood:

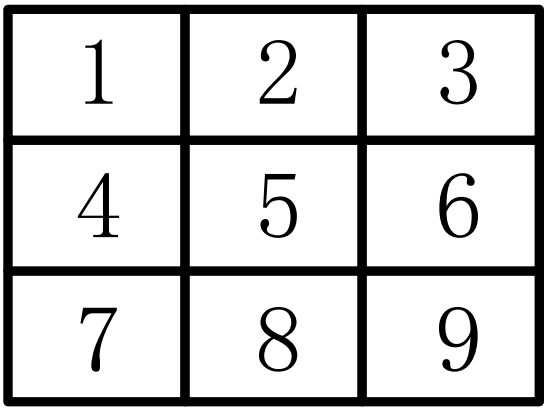


Figure 2. Pixel matrix in the 3×3 neighborhood.

The luminance of any pixel is assumed to be weighted by the luminance of other pixels in the 3×3 neighborhood, and different pixel points correspond to the same weights as their neighborhoods. For all the different pixels, a function $f_k(\omega_1, \omega_2, \ldots, \omega_9)$ is constructed concerning the 9-pixel weights, as shown in Equation (1).

$$f(\omega_1, \omega_2, \ldots, \omega_9) = \sum_{i=1}^{H} \sum_{j=1}^{W} \left(\omega_1 I_{i-1,j-1} + \omega_2 I_{i,j-1} + \omega_3 I_{i+1,j-1} + \cdots + \omega_9 I_{i+1,j+1} - I_{i,j}\right)^2 \tag{1}$$

where $\omega_1, \omega_2, \ldots, \omega_9$ is the weight of a pixel point with other pixels in its 3×3 neighborhood, $I_{i,j}$ denotes the brightness of a pixel with horizontal coordinate I and vertical coordinate j in the image pixel matrix, H is the image pixel matrix height value, and W is the width value of the image pixel matrix. According to the least squares criterion, the derivative function of any pixel weight ω_k of f is 0, i.e., Equation (2) holds.

$$\frac{\partial f(\omega_1, \omega_2, \cdots, \omega_9)}{\partial \omega_k} = 0 \tag{2}$$

Bringing all the weights into Equation (2) according to Equation (1), a linear system of equations on the pixel weights is obtained, as shown in Equation (3).

$$\begin{cases} \sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} \left(\omega_1 I_{i-1,j-1} + \omega_2 I_{i,j-1} + \cdots + \omega_9 I_{i+1,j+1} - I_{i-1,j-1}\right) \times I_{i-1,j-1} = 0 \\ \sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} \left(\omega_1 I_{i-1,j-1} + \omega_2 I_{i,j-1} + \cdots + \omega_9 I_{i+1,j+1} - I_{i,j-1}\right) \times I_{i,j-1} = 0 \\ \cdots \\ \sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} \left(\omega_1 I_{i-1,j-1} + \omega_2 I_{i,j-1} + \cdots + \omega_9 I_{i+1,j+1} - I_{i+1,j+1}\right) \times I_{i+1,j+1} = 0 \end{cases} \tag{3}$$

By solving Equation (3) through the Jacobi iteration method, the optimal solution of any pixel weight can be obtained, as shown in Equation (4):

$$\omega_k^{(t)} = \frac{\sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} \left(I_{i,j} - \left(\omega_1^{(t-1)} I_{i-1,j-1} + \omega_2^{(t-1)} I_{i,j-1} + \cdots + \omega_9^{(t-1)} I_{i+1,j+1}\right)\right) \times I_{i,j}}{\sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} I_{i,j}^2} - \omega_k^{(t-1)} \tag{4}$$

where $k \in [1, 9]$, and the superscript t denotes the number of iterations. Let Equation (5) hold constant for all pixels. We set an error value to determine the termination of the iteration. When the error value is less than 0.001, the image effect difference does not change much. To take into account the efficiency and the image quality, we set the error value to 0.001.

$$\left|\omega_k^{(t)} - \omega_k^{t-1}\right| \leq 0.001 \tag{5}$$

Then, the Jacobi iterative method converges; at this point, $\omega_k^{(t)}$ is the least squares estimate of the pixel weights. The obtained weights are used to estimate the illumination. Equation (6) shows the estimation result of the illumination.

$$I^*_{(i,j)} = \omega_1 I_{i-1,j-1} + \omega_2 I_{i,j-1} + \omega_3 I_{i+1,j-1} + \cdots + \omega_9 I_{i+1,j+1} \tag{6}$$

3.2. Improved Illumination Estimation

The experiment found that the difference between the brightness value of a pixel and other pixels in its neighborhood is too large, that is, if it is in the edge area, the artifact of a halo will occur. To prevent the artifact of halos, i.e., to achieve the function of edge detection, this paper introduces boundary-aware weights.

$$Z_G(I'_{i,j}) = N \times \frac{\sigma_G^2(I'_{i,j}) + \varepsilon}{\sigma_G^2(I_{i,j}) + \varepsilon} \tag{7}$$

where $i \in [1, H]$,$j \in [1, W]$, N is the total number of pixels in the 3×3 area, $\sigma_G^2(I'_{i,j})$ is the variance value of the pixel in the 3×3 neighborhood window, and $\sigma_G^2(I_{i,j})$ is the variance of all pixel points in the 3×3 neighborhood except for the selected pixel points. As $\sigma_G^2(I_{i,j})$ is the variance value whose minimal value can be 0 for the area where the luminance value remains unchanged, ε is a small constant value used to avoid the singularity that occurs with 0 values in these cases when taking division operation. It is used to ensure the validity of weights. We also should ensure that the value of ε is as small as possible while adapting to different images. So, when the input image is L, set it to $(0.001L)^2$. When $I'_{i,j}$ is in the edge region, the value $Z_G(I'_{i,j})$ is usually greater than 1. When $I'_{i,j}$ is in the smoothing region, the value $Z_G(I'_{i,j})$ is usually less than 1. We combined the interpixel weights obtained by the least squares method and obtained an improved illumination layer estimation for solving the halo artifacts, as shown in the following equation.

$$I^*_{i,j} = \omega_1 \frac{1}{Z_G(I'_{i-1,j-1})} I_{i-1,j-1} + \omega_2 \frac{1}{Z_G(I'_{i,j-1})} I_{i,j-1} + \cdots + \omega_9 \frac{1}{Z_G(I'_{i+1,j+1})} I_{i+1,j+1} \tag{8}$$

where $I^*_{i,j}$ is the improved illumination estimation. When a pixel is at the edge, causing a large value of ω_j, the introduction of boundary-aware weights can effectively reduce the impact of the brightness of other pixels, thus solving the halo artifacts caused by large differences in weights. When a pixel is in the smoothing region, introducing improved weights can enlarge the smoothing region and enhance the image details.

3.3. Dynamic Range Compression

To make full use of the advantages of histogram equalization and solve the problem of image over-enhancement caused by the method, in this paper, a global tone mapping function with a parameter was designed to compress the dynamic range. Based on the least squares, the parameter is calculated by fitting the global tone mapping function to the histogram equalization. The following equation is the global tone mapping function.

$$L_d(x, y) = \frac{I_{i,j}}{\alpha + I_{i,j}} \tag{9}$$

where $L_d(x, y)$ is the mapped luminance, $I_{i,j}$ is the input luminance value, and α is the parameter; the result of the mapping is shown in Figure 3.

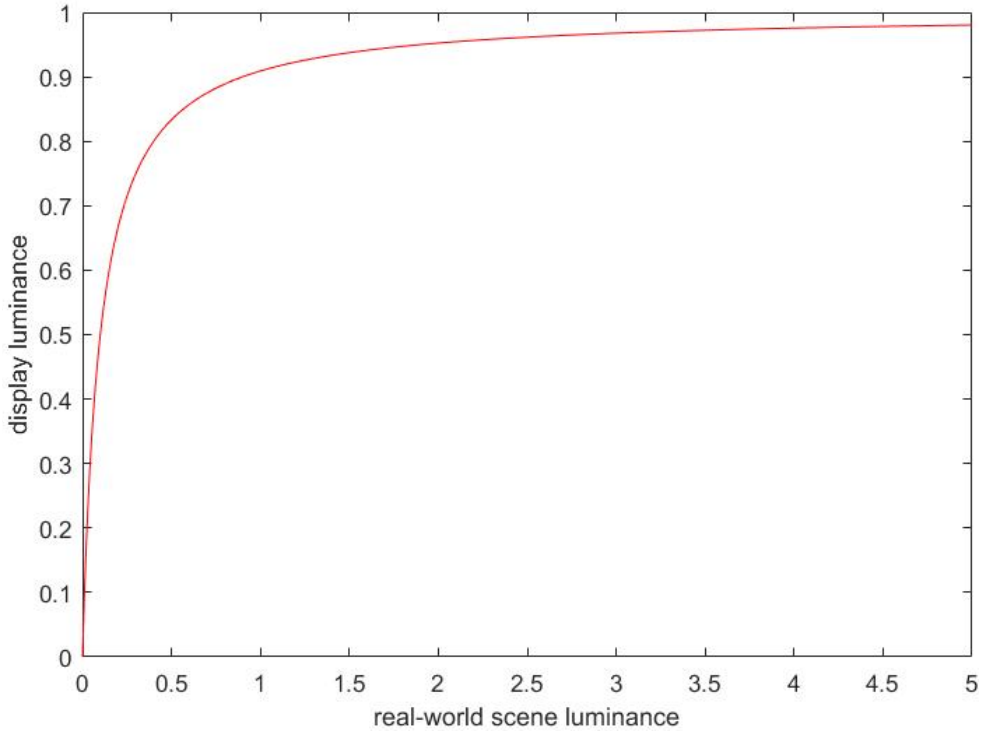


Figure 3. The dynamic range compression curve.

We take the reciprocal of the global tone mapping function and histogram equalization results at the same time and build a functional relationship between the global tone mapping function and histogram equalization based on the least squares criterion:

$$F(\alpha) = \sum_{i=1}^{H}\sum_{j=1}^{W}\left[\frac{\alpha + I_{i,j}}{I_{i,j}} - \frac{1}{f(I_{i,j})}\right]^2 \tag{10}$$

where $f(I_{i,j})$ is the calculation result of histogram equalization. According to the least squares criterion, the derivative of α is 0; this shows that Equation (11) holds, and Equation (12) is obtained:

$$\frac{\partial F(\alpha)}{\partial \alpha} = 0 \tag{11}$$

$$\sum_{i=1}^{H}\sum_{j=1}^{W}\left[\frac{(\alpha + I_{i,j})f(I_{i,j}) - I_{i,j}}{I_{i,j}f(I_{i,j})}\right] = 0 \tag{12}$$

By solving Equation (12), the parameter α of the global tone mapping function can be obtained:

$$\alpha = \sum_{i=1}^{H}\sum_{j=1}^{W}\frac{I_{i,j}\left[1 - f(I_{i,j})\right]}{f(I_{i,j})} \tag{13}$$

According to the obtained parameters, the global tone mapping functions corresponding to different images can thus be obtained, and the dynamic range compression of different images can be achieved.

3.4. LDR Image Generation

3.4.1. Detail Layer Estimation

The maximum value between each pixel RGB channel is set as the luminance component of the image, as shown in Equation (14).

$$L(x, y) = \max\{R(x, y), G(x, y), B(x, y)\} \tag{14}$$

where $L(x,y)$ is the luminance maximum for each pixel RGB channel, which is used to normalize the RGB colors. Then, calculate the color components for each channel, as shown in Equation (15).

$$r(x,y) = \frac{R(x,y)}{L(x,y)}, g(x,y) = \frac{G(x,y)}{L(x,y)}, b(x,y) = \frac{B(x,y)}{L(x,y)} \tag{15}$$

Based on the estimation of illumination obtained above, the detail layer of the image is obtained, as shown in the following equation

$$L^*(x,y) = \log(L(x,y)) - \log(I^*_{i,j}) \tag{16}$$

where $L^*(x,y)$ is the image detail layer.

3.4.2. Image Fusion and Color Retention

The detail layer and the illumination layer after dynamic range compression are fused to obtain the result, as shown in the following equation:

$$L_a(x,y) = \beta L_d(x,y) + \gamma e^{L^*(x,y)} \tag{17}$$

where $L_a(x,y)$ is the brightness value after fusion, β and γ are the reconstruction coefficients of the image fusion process, and the reconstruction coefficients take different values because the image illumination layer and the detail layer contain different information. The illumination layer contains the dynamic range of the image, which generally needs to be compressed, and the choice for β is generally less than 1. However, if the value is too small, it will cause problems such as a low overall brightness value and low contrast, which will affect the image quality. The detail layer contains the image detail information, which generally needs to be enhanced, and the choice γ is generally greater than 1. However, it should not be set too large because it can cause excessive detail enhancement while creating problems such as halo artifacts and unnatural overall image effects. Based on that, we set β and γ, respectively, to 0.8 and 1.2.

After the above processing, the color information needs to be recovered, as shown in the following equation.

$$\begin{cases} R^*(x,y) = L_a(x,y) \times [r(x,y)]^s \\ G^*(x,y) = L_a(x,y) \times [g(x,y)]^s \\ B^*(x,y) = L_a(x,y) \times [b(x,y)]^s \end{cases} \tag{18}$$

where $R^*(x,y)$,$G^*(x,y)$,$B^*(x,y)$ are the three-channel information after color recovery, and s is the gamma correction factor that determines the color saturation of the recovered image. In most cases, we can achieve close-to-ideal colors mostly by using a monitor with a gamma value of 2.2. Most LCD monitors are designed based on a gamma value of 2.2. So, we set $s = 1/2.2$, i.e., $s = 0.45$, in our method.

4. Experimental Results and Analysis

An experimental platform with an Intel i5-6300HQ processor, 8 GB memory capacity, 64-bit Windows 10 operating system, and Matlab 2016b simulation software was used to verify the feasibility of the proposed method. The LDR images were displayed on a Samsung LCD monitor model S32AM700PC. HDR images from the TMQI database [14] were selected for experimental comparison and analysis, and all images in the TMQI database are shown in Figure 4. The dataset contains some high-dynamic range scenes from the world, such as landscape photos, architecture photos, etc. A total of 15 selected HDR images from the database were used to analyze and evaluate the LDR images tone-mapped by TVI-TMO [3], the method of Li et al. [7], the method of Aziz et al. [1], the method of Gu et al. [15], the method of Shibata et al. [16], and the proposed method. Tone-mapped images were evaluated by subjective and objective assessment.

Figure 4. TMQI dataset.

4.1. Subjective Effect Analysis

The three sets of experimental results are shown in Figure 5, Figure 6, and Figure 7, respectively. From the experimental results, it can be seen that TVI-TMO retains the background information in the image, and the detail part is enhanced, but the overall brightness difference is large. The detail of the result of Li et al. is clearer, but the overall picture performance is not realistic. The result of Aziz et al. has good overall brightness performance, but there are incomplete details at extremely bright places. The result of Gu et al. is clear, but there is an artifact of the halo at the edge of the image. The result by Shibata et al. has overall good image perception, and the image details are clear, but there is the problem of low brightness. The proposed method has a good sense of image hierarchy, moderate brightness, and effectively preserves the edges of the original image and solves the halo artifacts, which is in line with the visual effect of the human eye.

(**a**) (**b**) (**c**)

Figure 5. *Cont.*

(**d**) (**e**) (**f**)

Figure 5. The tone-mapped images of the church. (**a**) Result of TVI-TMO; (**b**) Result of Li et al.; (**c**) result of Aziz et al.; (**d**) result of Gu et al.; (**e**) result of Shibata et al.; (**f**) result of the proposed method.

(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 6. The tone-mapped images of the landscape. (**a**) Result of TVI-TMO; (**b**) result of Li et al.; (**c**) result of Aziz et al.; (**d**) result of Gu et al.; (**e**) result of Shibata et al.; (**f**) result of the proposed method.

Figure 7. The tone-mapped images of the architecture. (**a**) Result of TVI-TMO; (**b**) result of Li et al.; (**c**) result of Aziz et al.; (**d**) result of Gu et al.; (**e**) result of Shibata et al.; (**f**) result of the proposed method.

Looking at Figures 6 and 7, it can be seen that, when compared to the method of this paper, the method of Gu et al., although it seems clearer, is not universal. The evaluation of LDR images generated by the method of Gu et al. for different scenes is significantly different. Moreover, the method of Gu et al. suffers from halo artifacts, an unnatural appearance, color distortion, and overexaggerated detail. For example, the edge between the leaf and sky in Figure 6d shows clear halos, the detail of the trunk and the sunroof in Figures 5d and 6d are overexaggerated, there are halo artifacts at the edges of the buildings and the sky in Figure 7d, and all three images produced by the method of Gu et al. suffer from low saturation. This demonstrates that the method of Gu et al. makes the output image visually unnatural to some extent.

We use the calculation of the mean opinion score (MOS) [7] for each comparison method to subjectively evaluate the tone mapping method. The method involves each participant in the experiment scoring each image on parameters such as brightness, contrast, and detail according to subjective visual effects. The method displayed the tone-mapped images on the monitor one by one. Eight experienced female and nine experienced male volunteers were selected, and each volunteer scored the images on the monitor subjectively in the range of [1,10]. The evaluation indexes include brightness, contrast, and detail. A rating of 1 represents the worst visual image, and a rating of 10 represents the best visual image. All the images in the TMQI database were scored, and the MOS values for each comparison method were counted. The obtained MOS score statistics are shown in Figure 8.

The high values of the indicators in Figure 8 indicate that the image quality after tone mapping is good.

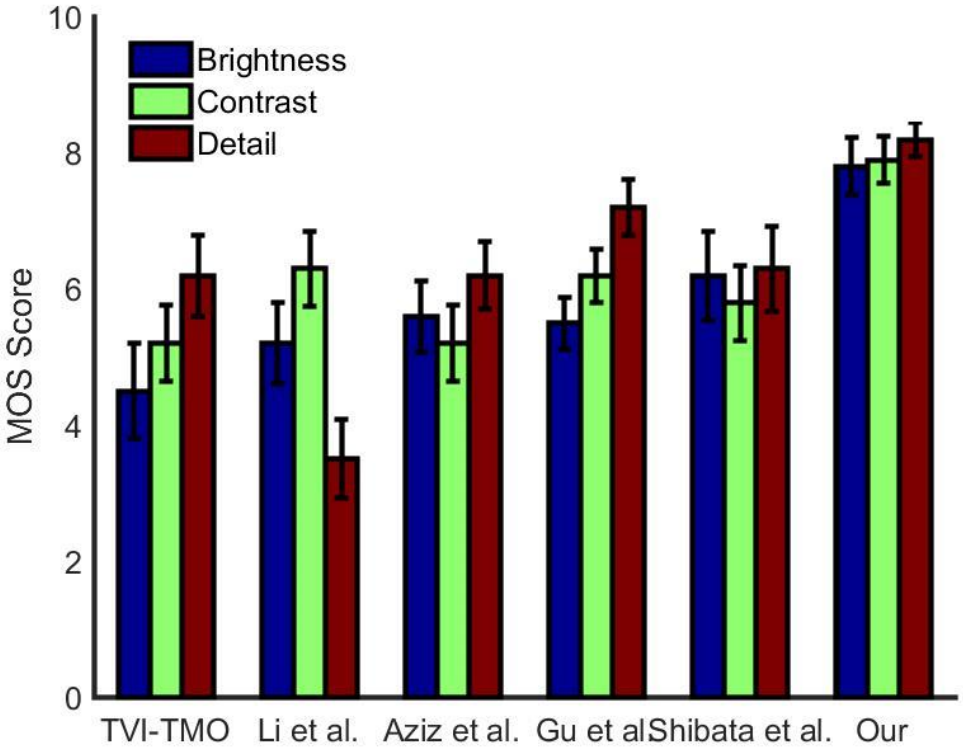


Figure 8. Statistical chart of MOS scores.

In the horizontal axis of Figure 8, the methods under comparison are displayed and the vertical coordinates are the MOS scores. The vertical line at the center of the upper boundary of the bar chart corresponds to the error line for each method, and the magnitude of the error line represents the standard deviation of the scores scored by different people.

Note that even though rare LDR images produced by the method of Gu et al. are cleaner than our method, the method of Gu et al. easily causes an unnatural effect by introducing halo artifacts, color distortion, and overexaggerated detail. More importantly, the method of Gu et al. is not a universal method. Thus, as an average evaluation, the MOS score of the image produced by the method of Gu et al. is a little lower than our method.

According to the scoring results, the results obtained by the proposed method were evaluated highly by most of the experimenters. The method has the smallest evaluation error, which indicates that the proposed method can adapt to the variability of images between different scenes, and the effect of the mapped images is consistent with the visual effect of most people and has strong stability for different individuals.

4.2. Objective Evaluation Effect Analysis

We used the objective evaluation method of tone mapping quality index (TMQI) [14] to evaluate the results of the method, after tone mapping, more comprehensively. The TMQI method consists of three evaluation metrics: structural fidelity, naturalness, and image quality score. Structural fidelity measures the extent to which the generated low-dynamic-range image maintains the structural information of the original image. Naturalness is a criterion designed by comparing the luminance statistics of a large number of natural images, and the larger the value, the more suitable it is for human eye observation. The image quality score is a composite score of the first two metrics. The TMQI scores are shown in Table 1.

Table 1. Tone mapping TMQI scores.

Image Index	TVI-TMO	Li et al.	Aziz et al.	Gu et al.	Shibata et al.	Proposed Method
1	0.725	0.735	0.785	0.822	0.796	0.896
2	0.693	0.705	0.728	0.832	0.826	0.908
3	0.753	0.723	0.722	0.831	0.785	0.886
4	0.659	0.695	0.732	0.796	0.801	0.876
5	0.698	0.698	0.702	0.852	0.793	0.894
6	0.682	0.712	0.712	0.804	0.811	0.898
7	0.632	0.722	0.710	0.832	0.786	0.901
8	0.689	0.709	0.722	0.862	0.822	0.876
9	0.705	0.708	0.709	0.821	0.815	0.903
10	0.688	0.725	0.774	0.832	0.813	0.893
11	0.668	0.731	0.765	0.845	0.809	0.901
12	0.675	0.724	0.755	0.833	0.795	0.897
13	0.685	0.706	0.742	0.826	0.782	0.882
14	0.690	0.730	0.738	0.836	0.821	0.902
15	0.675	0.726	0.729	0.842	0.796	0.896
Average	0.687	0.720	0.734	0.831	0.803	0.893

In addition, the local mean Weber contrast (MLWC) [17] metric, natural image quality evaluator (NIQE) [18], and blind/referenceless image spatial quality evaluator (BRISQUE) [19] are selected to further evaluate the performance of each method. The mean local Weber contrast index is a perceived contrast measure based on Weber's law, which takes into account the sensitivity of human visual effects to the spatial frequency content of an image. MLWC evaluates the image detail, and a higher the MLWC score indicates richer details of the LDR image. The natural image quality evaluator (NIQE) model extracts a set of local features from an image to evaluate image quality by using a multivariate Gaussian model. A smaller NIQE score indicates a higher image quality of the LDR image. BRISQUE compares the processed image to a default model computed from images of natural scenes with similar distortions. A small score indicates better perceptual quality. Several evaluation indexes' scores are shown in Tables 2 and 3.

Table 2. Tone mapping MLWC and NIQE scores.

Image Index	TVI-TMO		Li et al.		Aziz et al.		Gu et al.		Shibata et al.		Proposed Method	
	MLWC	NIQE	MLWC	NIQE	MLWC	NIQE	MLQC	NIQE	MLWC	NIQE	MLWC	NIQE
1	1.225	5.56	1.263	5.26	1.325	4.89	1.456	3.75	1.320	3.62	1.689	3.06
2	1.382	5.89	1.256	5.36	1.382	4.86	1.442	3.85	1.423	3.52	1.742	3.16
3	1.330	5.36	1.396	5.12	1.430	4.92	1.323	3.91	1.425	3.66	1.623	2.94
4	1.356	5.45	1.423	5.14	1.456	4.93	1.412	4.04	1.365	3.42	1.712	3.26
5	1.289	5.25	1.368	5.13	1.489	4.95	1.469	3.82	1.423	3.44	1.569	3.56
6	1.332	5.54	1.232	5.22	1.332	5.01	1.303	3.88	1.436	3.38	1.603	3.18
7	1.236	5.38	1.389	5.15	1.336	4.75	1.330	3.76	1.496	3.51	1.630	2.89
8	1.256	5.42	1.323	4.95	1.356	4.79	1.498	3.81	1.368	3.55	1.598	3.18
9	1.212	5.39	1.225	5.02	1.412	4.85	1.305	3.84	1.456	3.78	1.605	3.14
10	1.265	5.72	1.386	5.11	1.365	4.91	1.412	3.91	1.423	3.70	1.612	3.17
11	1.313	5.32	1.389	5.18	1.213	4.93	1.363	3.69	1.425	3.78	1.563	3.11
12	1.158	5.63	1.305	5.06	1.258	4.99	1.423	3.70	1.403	3.69	1.723	3.12
13	1.268	5.75	1.332	5.13	1.352	4.97	1.412	3.83	1.456	3.59	1.659	3.08
14	1.312	5.12	1.268	5.11	1.372	4.86	1.386	3.62	1.432	3.65	1.756	3.25
15	1.330	5.35	1.392	5.03	1.402	4.88	1.396	3.72	1.392	3.58	1.723	3.20
Mean	1.283	5.47	1.330	5.13	1.363	4.89	1.395	3.81	1.473	3.60	1.654	3.15

Table 3. Tone mapping BRISQUE scores.

Image Index	TVI-TMO	Li et al.	Aziz et al.	Gu et al.	Shibata et al.	Proposed Method
1	43.63	43.25	38.52	32.14	33.14	20.15
2	42.53	42.21	37.25	35.85	29.26	23.15
3	43.52	42.25	38.39	26.14	33.36	24.15
4	43.53	43.25	35.26	31.25	34.57	21.14
5	42.25	42.38	39.25	30.25	38.14	20.19
6	40.59	43.12	37.32	29.25	35.25	22.83
7	41.28	42.24	36.25	32.18	33.14	20.69
8	38.58	42.69	38.25	30.15	33.47	20.73
9	42.55	44.89	36.14	33.25	32.16	22.14
10	43.52	42.28	35.69	30.19	32.47	22.98
11	44.25	42.12	35.89	28.93	33.88	20.78
12	43.52	43.25	37.25	34.12	34.17	22.79
13	42.36	42.25	37.86	30.13	33.49	21.47
14	43.14	41.25	36.14	31.25	32.95	20.83
15	43.25	42.25	39.25	26.85	31.85	19.09
Mean	42.57	42.65	37.25	30.80	33.42	21.54

As can be seen from Table 1, the TMQI score of this method is higher than other methods, which indicates that the proposed method maintains a better structure and the visual effect is closer to the real scene. From Table 2, the MLWC score is higher than other methods, which indicates that the proposed method is better than other methods in terms of detail richness. The NIQE and BRISQUE scores are lower than those of several other methods, which indicates that the proposed method is better than other methods in terms of image quality.

5. Conclusions

In this paper, we proposed a tone mapping method based on the least squares. Our purpose is to reveal the local contrast of real-world scenes on a conventional monitor. Based on the classical Retinex model, we used the interimage weights by the least squares for illumination estimation, while we introduced boundary-aware weights to prevent halo artifacts. The detail layer is obtained by the above method. The method using weights takes into account the luminance intensity and pixel position, which can prevent the generation of halo artifacts while ensuring accurate estimation of illumination information. A global tone mapping function with the parameter is applied to the obtained illumination estimation to compress the dynamic range. The function is obtained by fitting the histogram equalization results, and different images correspond to different parameter values. This method can achieve dynamic range compression for different images more effectively. The detail layer and the illumination layer after dynamic range compression are fused to obtain the LDR image. The high dynamic range scenes in the TMQI dataset were selected for comparative analysis from both subjective and objective aspects. The experimental results show that details of both dark and highlighted regions are preserved, and halo artifacts do not appear. In the proposed method, the TMQI score of LDR images is 0.893 and the MLWC score is 1.654; the detail richness, structure fidelity, and visual effect are all better than other methods in different degrees. In this paper, we focused on solving the problems of illumination estimation and dynamic range compression in tone mapping, and we will focus on using tone mapping for video processing in the future.

Author Contributions: Conceptualization, L.Z.; methodology, L.Z. and J.W.; software, G.L.; validation, L.Z., G.L. and J.W.; formal analysis, G.L.; investigation, G.L.; resources, J.W.; data curation, L.Z.; writing—original draft preparation, G.L.; writing—review and editing, J.W.; visualization, J.W.; supervision, L.Z.; project administration, L.Z. and J.W.; funding acquisition, L.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Quzhou Science and Technology Plan Project, grant number 2022K108; Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014; and Basic Scientific Research Foundation Project of Provincial Colleges and Universities in Heilongjiang Province, grant number 2022KYYWF-FC05.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors acknowledge Quzhou Science and Technology Plan Project (grant number 2022K108), Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014), and Basic Scientific Research Foundation Project of Provincial Colleges and Universities in Heilongjiang Province (grant number 2022KYYWF-FC05).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, I.R.; Aziz, W.; Shim, S.O. Tone Mapping Using Perceptual-quantizer and Image Histogram. *IEEE Access* **2020**, *12*, 12–15. [CrossRef]
2. Choi, H.H.; Kang, H.S.; Yun, B.J. A Perceptual Tone Mapping HDR Images Using Tone Mapping Operator and Chromatic Adaptation Transform. *Imaging Sci. Technol.* **2017**, *61*, 15–20. [CrossRef]
3. Khan, I.R.; Rahardja, S.; Khan, M.M.; Movania, M.M.; Abed, F.A. Tone Mapping Technique Based on Histogram Using a Sensitivity Model of The Human Visual System. *IEEE Trans. Ind. Electron.* **2018**, *65*, 3469–3479. [CrossRef]
4. Husseis, A.; Mokraoui, A.; Matei, B. Revisited Histogram Equalization as HDR Images Tone Mapping Operators. *IEEE Comput. Soc.* **2017**, *15*, 144–149.
5. Chien, B.; Mokraoui, A.; Matei, B. Contrast Enhancement and Details Preservation of Tone Mapped High Dynamic Range Images. *Image Represent.* **2018**, *58*, 22–26.
6. Kai-Fu, Y.; Hui, L.; Hulin, K.; Chao-Yi, L.; Yong-Jie, L. An Adaptive Method for Image Dynamic Range Adjustment. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *29*, 15–18.
7. Hui, L.; Xixi, J.; Zhang, L. Clustering Based Content and Color Adaptive Tone Mapping. *Comput. Vis. Image Underst.* **2018**, *168*, 15–20.
8. Thai, B.C.; Mokraoui, A. HDR Image Tone Mapping Histogram Adjustment with Using an Optimized Contrast Parameter. In Proceedings of the International Symposium on Signal, Image, Video and Communications (ISIVC), Rabat, Morocco, 27–30 November 2018; Volume 12, pp. 15–20.
9. Fahim, M.A.N.I.; Jung, H.Y. Fast Single-Image HDR Tone Mapping by Avoiding Base Layer Extraction. *Sensors* **2020**, *20*, 4378. [CrossRef] [PubMed]
10. Gommelet, D.; Roumy, A.; Guillemot, C.; Ropert, M.; Julien, L. Gradient-Based Tone Mapping for Rate-Distortion Optimized Backward-Compatible High Dynamic Range Compression. *IEEE Trans. Image Processing* **2017**, *15*, 12–18. [CrossRef] [PubMed]
11. Liang, Z.; Xu, J.; Zhang, D. A Hybrid l1-l0 layer Decomposition Model for Tone Mapping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4758–4766.
12. Rana, A.; Singh, P.; Valenzise, G.; Dufaux, F.; Komodakis, N.; Smolic, A. Deep Tone Mapping Operator for High Dynamic Range Images. *IEEE Trans. Image Processing* **2019**, *29*, 18–25. [CrossRef] [PubMed]
13. Hu, L.; Chen, H.; Allebach, J.P. Joint Multi-Scale Tone Mapping and Denoising for HDR Image Enhancement. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; Volume 26, pp. 729–738.
14. Yeganeh, H.; Wang, Z. Objective Quality Assessment of Tone-mapped Images. *IEEE Trans. Image Processing* **2012**, *22*, 657–667. [CrossRef] [PubMed]
15. Gu, B.; Li, W.; Zhu, M. Local Edge-preserving Multiscale Decomposition for High Dynamic Range Image Tone Mapping. *IEEE Trans. Image Processing* **2012**, *22*, 70–79.
16. Shibata, T.; Tanaka, M. Gradient-domain Image Reconstruction Framework with Intensity-range and Base-structure Constraints. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27 June 2016; pp. 2745–2753.
17. Khosravy, M.; Gupta, N.; Marina, N. Perceptual Adaptation of Image Based on Chevreul–Mach Bands Visual Phenomenon. *IEEE Signal Processing Lett.* **2017**, *24*, 594–598. [CrossRef]
18. Mittal, A.; Soundararajan, R.; Bovik, A.C. Making a "completely blind" image quality analyzer. *IEEE Signal Process. Lett.* **2012**, *20*, 209–212. [CrossRef]
19. Mittal, A.; Moorthy, A.K.; Bovik, A.C. No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.* **2012**, *21*, 4695–4708. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

electronics

MDPI

Article

Three-Stage Tone Mapping Algorithm

Lanfei Zhao [1], Ruiyang Sun [2] and Jun Wang [3,*]

[1] The Higher Educational Key Laboratory for Measuring & Control Technology and Instrumentations of Heilongjiang Province, Harbin University of Science and Technology, Harbin 150080, China
[2] Chinese Martial Arts Department, Harbin Sport University, Harbin 150008, China
[3] School of Information Engineering, Quzhou College of Technology, Quzhou 324000, China
* Correspondence: 145688@qzct.edu.cn; Tel.: +86-181-0570-1939

Abstract: In this paper, a tone mapping algorithm is presented to map real-world luminance into displayed luminance. Our purpose is to reveal the local contrast of real-world scenes on a conventional monitor. Around this point, we propose a three-stage algorithm to visualize high dynamic range images. All pixels of high dynamic range images are classified into three groups. For the first stage, we introduce piecewise linear mapping as the global tone mapping operator to map the luminance of the first group, which provides overall impressions of luminance. For the second stage, the luminance of the second group is determined by the weighted average of its neighborhood pixels, which are derived from the first group's pixels. For the third stage, the luminance of the third group is determined by the weighted average of its neighborhood pixels, which are derived from the second group's pixels. Experimental results on several real-world images and the TMQI database show that our algorithm can improve the visibility of real-world scenes with about 12% and 9% higher scores of mean opinion score and tone-mapped image quality index than the closest competitive tone mapping methods. Compared to the existing tone mapping methods, our algorithm produces visually compelling results without halo artifacts and loss of detail.

Keywords: high dynamic range; tone mapping; piecewise linear mapping; local contrast

1. Introduction

The real-world scene covers a wide range of luminance levels, whose order of magnitude approximates about 10^{14}:1 [1]. Through the complicated process of self-adaption, the human visual system adapts over nine orders gradually. The dynamic range of modern image sensors is far less than real-world scenes and human visual systems. Nevertheless, by multi-exposure imaging [2], the limit of the image scene can be extended. Hence, we can obtain high dynamic range (HDR) images with conventional digital cameras.

The contrast ratio of some advanced conventional displays can reach up to 10^4:1 [3], although some high dynamic range display systems claim that they can achieve five orders of magnitude [4]. However, the dynamic range of display systems is much less than real-world scenes. Furthermore, the price of high contrast ratio displays is not affordable for household consumers. This flaw results in shadow and highlight that can not convey the appearance of the original scene. Tone mapping has been developed to convert an HDR image to a low dynamic range (LDR) image, whose dynamic range is compatible with conventional displays. A valid tone mapping algorithm can reveal both darkness and brightness pixels clearly on a conventional display. It is also able to preserve details in the original image and avoid common artifacts, such as halos, gradient reversals, or loss of local contrast [5].

Unfortunately, existing tone mapping algorithms suffer from some reduction in image quality, e.g., detail loss and unsuitable overall impression of brightness. To solve the rendering problem of HDR images, we present a three-stage tone mapping algorithm. The main idea is to map real-world luminance to displayable luminance while preserving

local contrast in the real-world scene. For mapping real-world luminance, previous tone mapping methods compress mainly high luminances [6]. Such tone-curves suffer from over-compression of the global contrast. To deal with this problem, we propose a piecewise linear mapping as the first stage to determine the brightness and enhance the global contrast of tone-mapped images. Each segment point of the piecewise linear mapping is selected by a tiny luminance threshold, and the remapped luminance is estimated by the cumulative probability of the luminance histogram. For visual detail protection, Ashikhmin [7] focused on preserving local contrast during compressing dynamic range. However, such a method could lead to insufficient performance of visual detail preservation. We introduce another two stages to ensure local contrast; both stages employ the weighted average of neighborhood pixels to estimate local contrast. The luminances derived from each stage are related to the previous stage by reviewing the local contrast expression. Experiments were conducted on radiance maps from various real-world scenes and show that the proposed algorithm can produce pleasing images.

2. Related Work

In an early attempt, global tone mappings designed a single curve to compress dynamic range and was designed to save computational efficiency [8]. A practical general framework to match real-world brightness and display brightness was built by Tumblin and Rushmeier [9]. They reviewed the sigmoid responses to light in the film's encoding process. Using this kind of power-law relation, they created an observer model that converts world luminance to perceived brightness; meanwhile, the coefficient and exponent are determined by Stevens's experiment. Ward [10] applied a designated multiplier to relate the minimum discernible difference on the display and in a real-world scene. Note that this method deals with the perceived contrast, consequently, it provides the same contrast visibility both in bright scenes and in dark scenes. Larson et al. [11] suggested that the eye is sensitive to region brightness, i.e., adaptation levels, rather than absolute luminance. To estimate adaptation levels, they averaged real-world luminance over a $1\bigcirc$ visual angle. Histogram equalization with a linear ceiling was applied to compress the dynamic range while the limit region contrast could not exceed the original image. Khan et al. [12] presented a histogram-based tone mapping algorithm to visualize HDR images on an LDR display. The algorithm restricts the pixel counts in the histogram to solve over-compression and over-enhancement problems introduced by classic histogram equalization.

Recent tone mapping research focuses on local tone mapping operators. Meylan et al. [13] rendered real-world images with a Retinex-based adaptive filter. This filter processes the luminance and chrominance in two parallel procedures. The chrominance processing employs principal component analysis to provide unaffected color rendition immunity while luminance is compressing. Gu et al. [14] developed a local edge-preserving (LEP) filter and applied the LEP filter to decompose real-world scene images into several detail layers and one base layer based on the Retinex model. Kuang et al. [15] proposed a valid image appearance modeling to reproduce the same visual perception across media based on the iCAM framework. This model decomposes an image into a base layer and a detail layer by the bilateral filter. The Cone response function and the rod's response function are applied to the base layer to compress the dynamic range. A power-function adjustment is applied to detail layer to predict the Stevens effect. Fattal et al. [5] took the luminance gradient as the local metric to preserve details and compress the dynamic range. The work attempts to attenuate the large magnitudes of gradients at each pixel while small magnitudes of gradients remain unchanged. Their work also introduces a Gaussian pyramid to avoid halo artifacts.

In recent years, deep-learning-based tone mapping algorithms have been presented to generate LDR images. Rana et al. [16] proposed a fast, parameter-free, and scene-adaptable deep tone mapping operator (DeepTMO) based on a conditional generative adversarial network. DeepTMO explores four possible combinations of Generator-Discriminator architectural designs to specifically address some prominent issues in HDR related deep-learning

frameworks like blurring, tiling patterns, and saturation artifacts. Panetta et al. [17] designed a deep-learning-based tone mapping operator (TMO-Net), which offers an efficient and parameter-free method capable of generalizing effectively across a wider spectrum of HDR content. Patel et al. [18] proposed a novel generative adversarial network to learn a combination of several classic tone mapping operators. The proposed method uses a deep network to keep the best TMQI [19] score tone-mapped image generated by the classic tone mapping operators.

3. Algorithm

3.1. Global Luminance Mapping

We apply the scene's key value proposed by Reinhard [20] to make an analogy to camera exposure. The approximation to the key of the scene is given by

$$\overline{L} = \frac{1}{M} \exp\left(\sum_{x,y}\left(\log(\delta + \widetilde{L}_{x,y})\right)\right) \tag{1}$$

where M is the total number of pixels, δ is a small value, $\widetilde{L}$ is world luminance, (x, y) is the pixel location and $\overline{L}$ is the key of the scene. The scaled luminance is computed by

$$L_{x,y} = \frac{a}{\overline{L}} \times \widetilde{L}_{x,y} \tag{2}$$

where a is the key value to map the log-average world luminance to an appropriate value and a is estimated adaptively by Reinhard's work [21].

We found that the tone distribution curve of a great number of radiance maps presents a similar shade, partially shown in Figure 1a, which is obtained from a well-known radiance map called "memorial church".

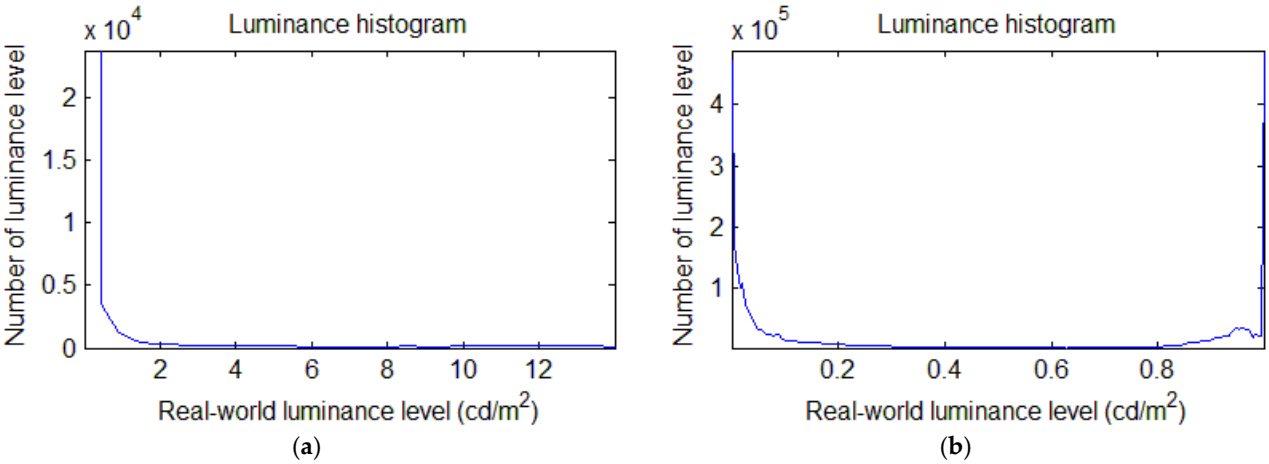


Figure 1. Luminance histogram of memorial church image. (**a**) Memorial Church; (**b**) Moto.

It can be seen that most luminance levels distribute at the beginning of the luminance level, and the curve plots a very long and low trailing. This indicates that it is necessary to make the compression for high luminance levels. Meanwhile, a majority of luminance levels, i.e., the low luminance levels, should be stretched to increase the global contrast.

Not all of the luminance histograms look like Figure 1a. The luminance levels of some scenes occur in both low-level bins and high-level bins, as shown in Figure 1b. This histogram comes from another radiance map called "moto". Here, we can see that it has equal importance both at low-level and high-level bins. This type of curve suggests that low-level and high-level bins should be stretched together.

Essentially, to pick a luminance mapping operator is to choose a proper shade of the mapping curve. If employing linear mapping as the luminance mapping, the visual of the remapping image is supposed to be consistent with the reconstructed image by the optic

nerve practically. However, the linear mapping is incapable of compressing dynamic range within the range of HDR images. Due to this, a piecewise linear mapping is advisable to replace linear mapping.

The next issue is to determine segment points for piecewise linear mapping. Since the bins possess very few numbers of luminance levels that should be compressed, we use the empirical value $T_a = 10^{-4}$ as the threshold to determine the location of segment points. When the probability of the previous bin is larger than the empirical value, while the probability of the current bin is not greater than the empirical value, the current bin is a segment point. When the probability of the previous bin is not greater than the empirical value, while the probability of the current bin is larger than the empirical value, the current bin is also a segment point. Moreover, the remapping value of a segment point is determined by the cumulative probability between the previous segment point and the current segment point. Let $B(\cdot)$ represent the luminance bin and $(\cdot)$ represent the number of a bin. The remapping value of the l-th bin between two segment points $B(i)$ and $B(j)$ is given by

$$L(B(l)) = \frac{L(B(j)) - L(B(i))}{B(j) - B(i)} \times (B(l) - L(B(i))) + L(B(i)) \tag{3}$$

where $L(B(l))$, $L(B(i))$, and $L(B(j))$ are the remapping value of the l-th bin, i-th bin, and j-th bin, respectively. Note that the remapping values of the first and last bin are 0 and 1, respectively. Thus, the remapping value of every segment point is known by formula (3). Using (3) after segmenting the histogram by an empirical threshold, we obtain the luminance mapping curves shown in Figure 2. Due to the slow varying specifically at the end of the curve shown in Figure 1a, the piecewise linear function mainly compresses the range of high luminance for memorial church image. Meanwhile, the low luminance level is stretched to enhance the contrast in dark areas. Both low luminance levels and high luminance levels are stretched and the dynamic range of the middle luminance level is compressed since most bins are densely distributed both at the beginning and end of the luminance histogram which is shown in Figure 1b.

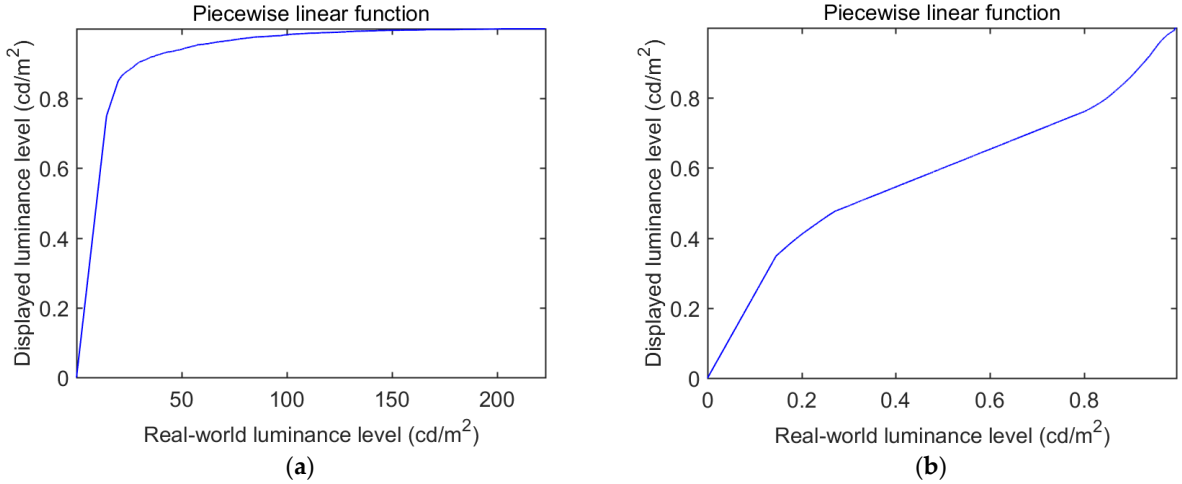


Figure 2. Piecewise linear function. (**a**) Memorial Church; (**b**) Moto.

3.2. Local Contrast Preservation

To preserve local contrast, all pixels of the input image are divided into three groups, i.e., U_1, U_2, and U_3. Figure 3 shows a classification procedure for an image with a size of 9×9. Both row and column numbers of the first group's pixels are odd, as shown in Figure 3a. There is only one spatial coordinate of the second group's pixels that is odd, as shown in Figure 3b. Both row and column numbers of the third group are even, as shown in Figure 3d. The second group pixel is divided into two subgroups, as shown in Figure 3c.

The row number of the first subgroup is even. On the contrary, the row number of the second subgroup is odd.

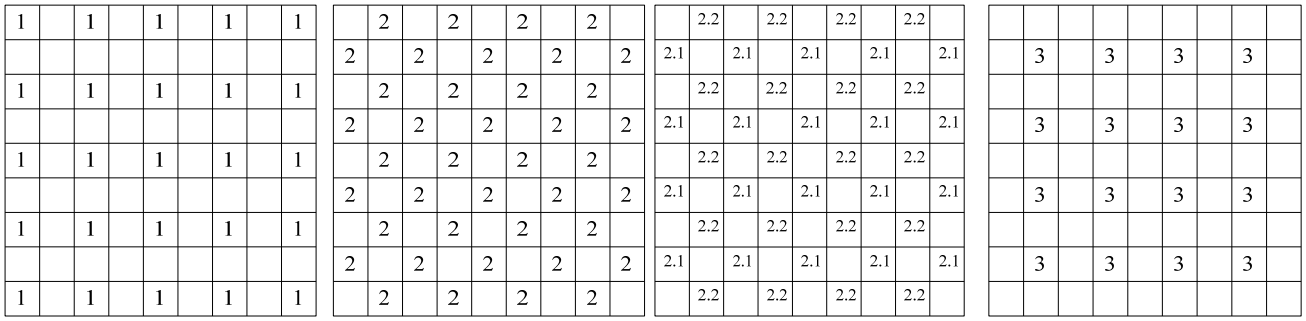


Figure 3. Pixel classification. (**a**) Pixels of group one; (**b**) Pixels of group two; (**c**) Subgroups for group two; (**d**) Pixels of group three.

We use the global luminance mapping described in Section 3.1 to compress the dynamic range for the first group's pixel. Then, the local contrast of the two subgroups is computed by

$$\begin{cases} c_{x,y}^{(1)} = \frac{3I_{x,y}}{I_{x-1,y}+I_{x,y}+I_{x+1,y}}, & (x,y) \in U_2^{(1)} \\ c_{x,y}^{(2)} = \frac{3I_{x,y}}{I_{x,y-1}+I_{x,y}+I_{x,y+1}}, & (x,y) \in U_2^{(2)} \end{cases} \tag{4}$$

where $(\cdot)$ represents the number of subgroups, c represents local contrast, I represents displayed luminance, and the superscripts of c and U represent the number of the subgroup. For local contrast preservation, the local contrast value of every pixel should be equal to the tone-mapped image, which can be formulated as:

$$\begin{cases} c_{x,y}^{(1)} = \frac{3L_{x,y}}{L_{x-1,y}+L_{x,y}+L_{x+1,y}}, & (x,y) \in U_2^{(1)} \\ c_{x,y}^{(2)} = \frac{3L_{x,y}}{L_{x,y-1}+L_{x,y}+L_{x,y+1}}, & (x,y) \in U_2^{(2)} \end{cases} \tag{5}$$

Since the luminance of the first group's pixels has been computed by global luminance mapping, i.e., $L_{x-1,y}$, $L_{x+1,y}$, $L_{x,y-1}$, and $L_{x,y+1}$ are known, the luminance of the second group's pixels derived by joint (4) and (5) is computed by:

$$I_{x,y} = \begin{cases} \frac{c_{x,y}^{(1)} \times (L_{x-1,y}+L_{x+1,y})}{3-c_{x,y}^{(1)}}, & (x,y) \in U_2^{(1)} \\ \frac{c_{x,y}^{(2)} \times (L_{x,y-1}+L_{x,y+1})}{3-c_{x,y}^{(2)}}, & (x,y) \in U_2^{(2)} \end{cases} \tag{6}$$

The local contrast of the third group's pixels is computed by:

$$c_{x,y} = \frac{5I_{x,y}}{I_{x-1,y} + I_{x+1,y} + I_{x,y} + I_{x,y-1} + I_{x,y+1}} \tag{7}$$

Similar to the second group's pixels, the luminance of the third group's pixels is computed by:

$$I_{x,y} = \frac{c_{x,y}^{(3)} \times \left(L_{x-1,y} + L_{x+1,y} + L_{x,y-1} + L_{x,y+1}\right)}{5 - c_{x,y}^{(3)}} \tag{8}$$

Figure 4 shows the flowchart of the three-stage approach.

All the pixels of the input real-world image are first classified into three groups. In Stage 1, the luminances of the pixels in group one are remapped by piecewise linear mapping. Both Stage 2 and Stage 3 compute the local contrast of every pixel of group two and group three. After piecewise linear mapping, i.e., Stage 1, Stage 2 computes the

luminance of all the pixels in group two, according to Equation (6). After Stage 2, Stage 3 computes the luminance of all the pixels in group three, according to Equation (7). All pixels of the three groups are collected to generate the tone mapped image.

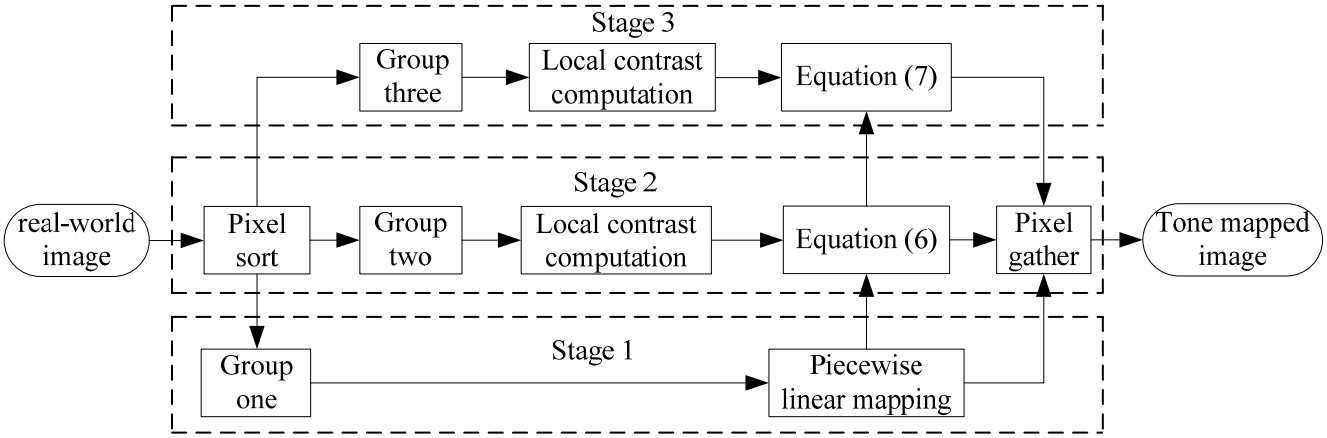


Figure 4. The flowchart of the three-stage algorithm.

3.3. Color Image

We separate lightness and color features by converting RGB images into HSV space. Figure 5 shows the data flow diagram for dealing with a color image.

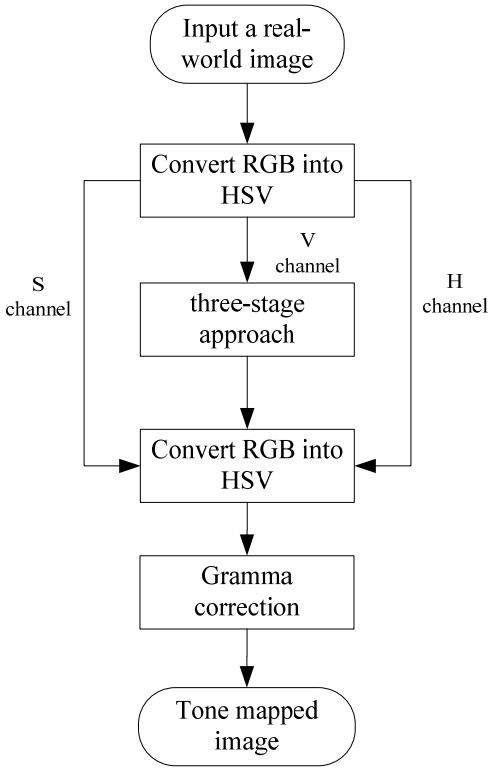


Figure 5. Data flow diagram for dealing with a color image.

Three channels are separated into H, S, and V channels, where the V channel is only used to compress dynamic range, and the H and S channels are constant. The last part is Gamma correction, which is used for overcoming the non-linear response of the display device.

4. Results

Our method is simulated by Matlab 2016a on a desktop computer with an Intel i5-7400 CPU and Kingston 2400 MHz 8G-DDR4 internal storage. Tone-mapped images are shown on a Dell S2721DGF digital display whose gamma parameter approximates 2.2. The test radiance maps contain three famous scenes and the TMQI database [19] with a very high dynamic range.

4.1. Subjective Performance Evaluation

In this part, we use three famous radiance maps to make a comparison among several tone mapping algorithms. The compared tone mapping algorithm include DeepTMO [16], Thai [22], TMO-Net [17], L1-L0 [23], and Khan [12]. The three test radiance maps are shown directly in Figure 6. Figures 7–9 demonstrate a comparison of tone-mapped images.

Figure 6. The picture shows the test radiance maps rendering on standard LCD. The left image, called "memorial church", is an HDR image from TMQI database [19]. The middle image, called "moto", and the right image, called "apartment", are reprinted with permission from Ref. [24]. 2022 Elsevier B.V. (Reprinted from Signal Processing: Image Communication, Vol (29), Authors (Narwaria, M., Da Silva, M.P., Le Callet, P. and Pépion, R.), Title of article (Tone mapping based HDR compression: Does it affect visual experience?), Pages (257–273), Copyright (2013), with permission from Elsevier).

The images derived by our algorithm show visually pleasing results due to the clear details and distinguishable information both in dark areas and bright areas. The compared tone mapping algorithms suffer from some reduction in image quality. TMO-Net provides a good, clear result in the darker areas, while the contrast in the brighter area is decreased which leads to a problem of overexposure. DeepTMO improves the global contrast of the original real-world image. The content of the dark area is revealed and the brightness becomes more apparent. However, the reproduced image was blurred while the details are reduced both in low-light and bright areas. Both Thai and Khan have the problem of contrast loss in the low-light area, which leads to over-compression of some intensity levels. There, some image quality is lost, particularly in darker areas. The visibility of the dark area is improved by L1-L0 and the tone-mapped images make a good balance between detail enhancement and visual naturalness for both indoor and outdoor scenes. However, the LDR image obtained by L1-L0 has problems with halo artifacts and the contrast is over-enhanced. Meanwhile, the tone-mapped images show a little brightness distortion.

Another experiment was conducted to test the validity of the tone mapping algorithm. We selected the TMQI database [19] that contains 15 radiance maps to compute the mean opinion score (MOS). TMQI database is shown in Figure 10. It can be seen that the TMQI database includes six indoor scenes and nine outdoor scenes. Due to the unequal resolution, we resize the images without aspect ratio preservation to favor the demonstration. The image displayed in Figure 10 leads to unsatisfactory visibility which is caused by extremely high contrast. The algorithms compared and the algorithm proposed were adopted to convert every radiance map of the TMQI database to a displayed image. Thus, we obtained 90 LDR images. Figure 11 shows four groups of LDR images. It can be seen from Figure 11 that the visibility of our method is improved with better performance than the other competitive tone mapping methods. The subjective visual quality of each tone-mapping image share quality consistent with the reproduced image shown in Figures 7–9.

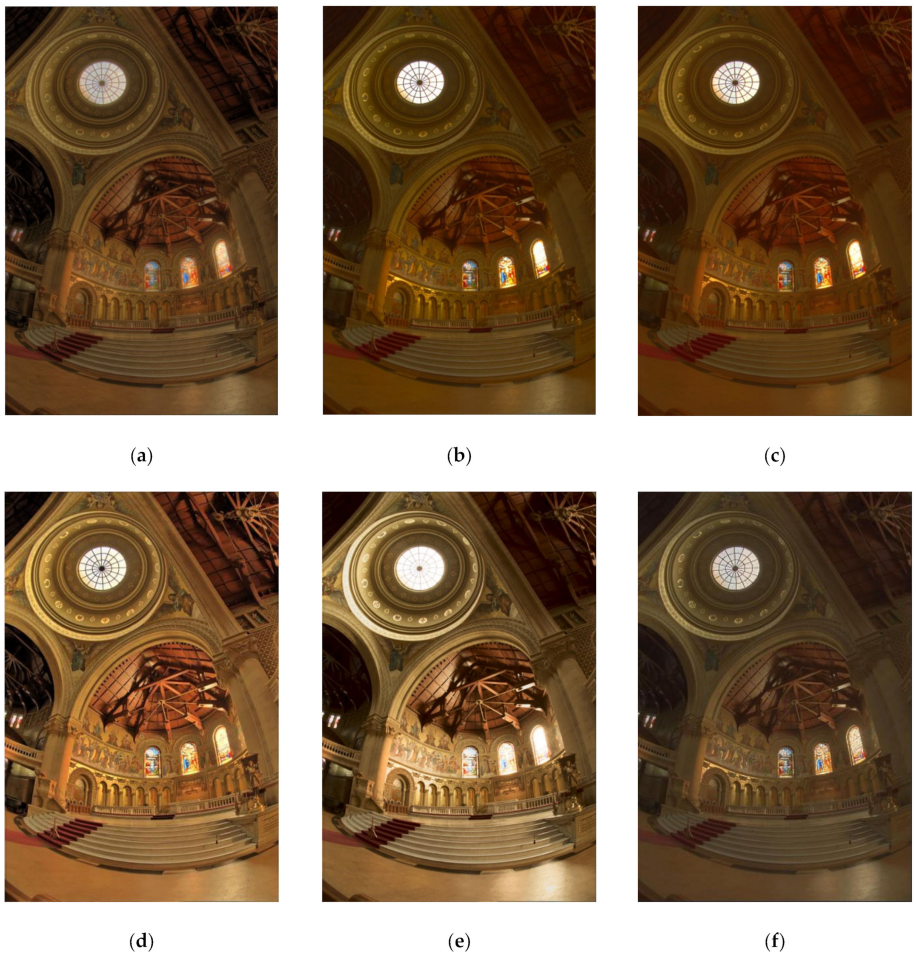


Figure 7. A comparison of tone mapping algorithms on the memorial church image. (**a**) Result by TMO-Net. (**b**) Result by DeepTMO. (**c**) Result by Thai. (**d**) Result by L1-L0. (**e**) Result by Khan. (**f**) Result by our algorithm.

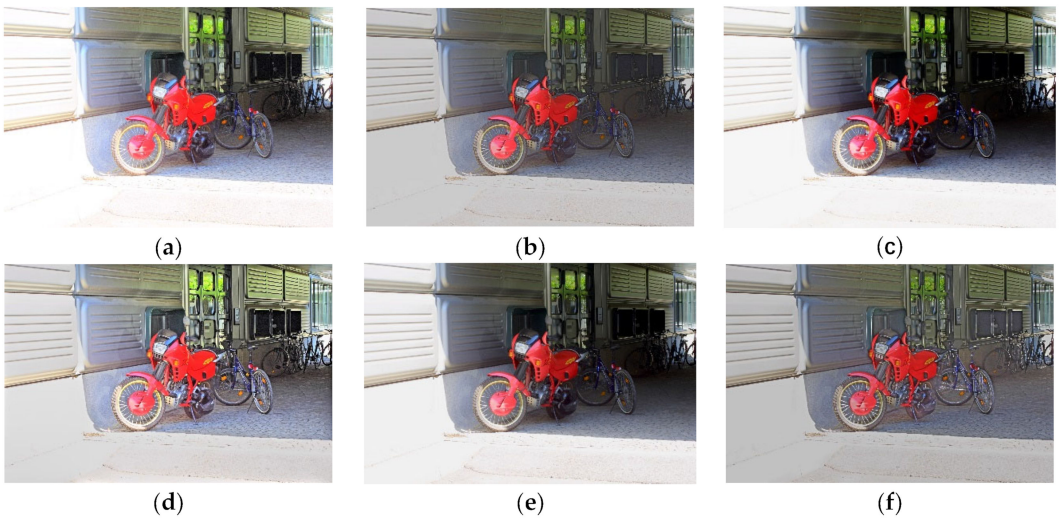


Figure 8. A comparison of tone mapping algorithms on the moto image. (**a**) Result by TMO-Net. (**b**) Result by DeepTMO. (**c**) Result by Thai. (**d**) Result by L1-L0. (**e**) Result by Khan. (**f**) Result by our algorithm.

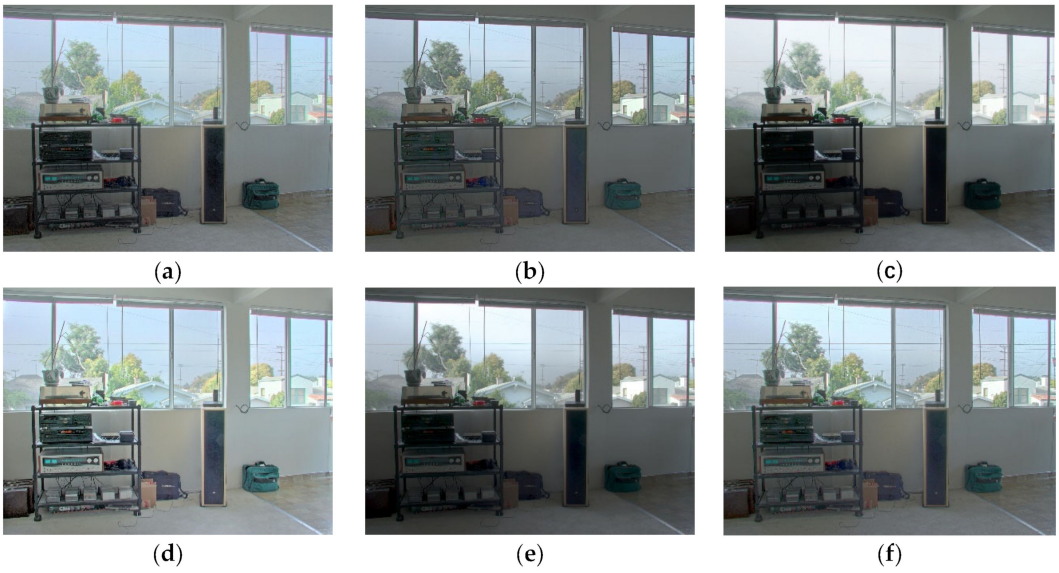


Figure 9. A comparison of tone mapping algorithms on the apartment image. (**a**) Result by TMO-Net. (**b**) Result by DeepTMO. (**c**) Result by Thai. (**d**) Result by L1-L0. (**e**) Result by Khan. (**f**) Result by our algorithm.

Figure 10. The 15 radiance maps from the TMQI database.

Then, the tone-mapped images of the TMQI database were shown on an S2721DGF Display. To determine MOS, 31 volunteers, including 16 females and 15 males aged between 23 and 34, were invited to give the score for each image shown on the display produced by the above tone mapping algorithms. The score range is from 1 to 10, where 1 means the worst visual quality and 10 means the best visual quality. The mean and std of MOS values are shown in Figure 12.

It can be seen that our method has a higher score (8.9) than other compared methods. Meanwhile, the lowest std value (0.175) means the performance of our algorithm is widely recognized among these volunteers. The mean score and standard deviation for other tone mapping methods are DeepTMO (7.9, 0.3), Thai (5.8, 0.6), Khan (5.1, 1.1), TMO-Net (7.2, 0.25), and L1-L0 (7.7, 0.4). According to the mean scores, our method achieves about 12% higher than the closest tone mapping method (DeepTMO).

Figure 11. Four groups of tone-mapped images were obtained by our algorithm and the compared algorithms. The order of sub-photographs keeps in line with Figure 7.

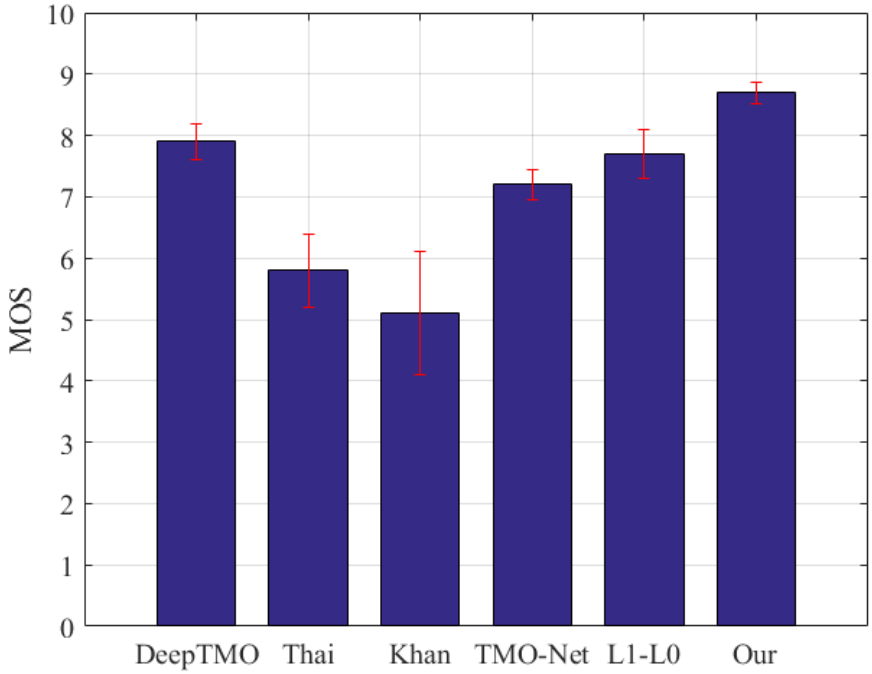


Figure 12. Mean and std of subjective rankings of the six tone mapping algorithms.

4.2. Objective Performance Evaluation

Besides subjective performance evaluation, we employed a tone-mapped image quality index (TMQI) [19] and natural image quality evaluator (NIQE) [25] as objective evaluations to further evaluate the performance of related tone mapping algorithms. The TMQI score ranges from 0 (worst quality) to 1 (best quality), and a smaller NIQE indicates a higher image quality. Tables 1 and 2 show the TMQI and NIQE scores for related tone mapping algorithms. According to Tables 1 and 2, our algorithm results in a higher score than the compared tone mapping algorithms. The mean TMQI score for our tone mapping method is higher by about 9%, 30%, 12%, 8%, and 41% as compared to DeepTMO, Thai, TMO-Net, L1-L0, and Khan tone mapping methods. The mean NIQE score for our tone mapping method is lower than the competitive tone mapping methods. The highest TMQI and lowest NIQE scores indicate the tone-mapped image obtained by our algorithm provides better structural fidelity and less statistical naturalness than the other methods.

Table 1. Objective evaluations using 6 tone mapping algorithms and 3 radiance maps.

Algorithm	Memorial Church		Moto		Apartment	
	TMQI	NIQE	TMQI	NIQE	TMQI	NIQE
L1-L0	0.893	3.62	0.876	3.73	0.834	3.82
DeepTMO	0.905	3.35	0.904	3.59	0.798	3.31
Khan	0.652	5.29	0.622	6.21	0.649	5.45
TMO-Net	0.852	4.28	0.814	5.07	0.824	4.34
Thai	0.711	6.67	0.685	6.75	0.659	5.75
Our	0.929	3.21	0.919	3.42	0.854	3.03

Table 2. Objective evaluations using 6 tone mapping algorithms and 15 radiance maps from the TMQI database.

Image Index	DeepTMO		Thai		TMO-Net		L1-L0		Khan		Our	
	TMQI	NIQE	TMQI	NIQE	TMQI	NIQE	TMQI	NIQE	TMQI	NIQE	TMQI	NIQE
1	0.801	3.21	0.733	6.66	0.810	3.60	0.850	3.15	0.788	5.35	0.900	3.11
2	0.829	3.24	0.701	5.65	0.844	3.97	0.796	3.13	0.553	5.82	0.888	3.06
3	0.843	3.87	0.710	5.69	0.875	4.14	0.879	3.81	0.654	5.93	0.891	3.32
4	0.885	3.84	0.671	5.82	0.854	4.33	0.863	3.58	0.721	6.36	0.897	3.46
5	0.812	3.84	0.681	4.75	0.760	4.47	0.837	3.57	0.619	5.53	0.904	3.47
6	0.831	3.55	0.684	6.94	0.722	3.99	0.850	4.13	0.632	4.68	0.888	2.94
7	0.825	3.12	0.705	6.38	0.781	3.47	0.839	3.74	0.552	5.24	0.880	3.14
8	0.784	3.57	0.684	5.84	0.851	3.71	0.779	3.71	0.541	6.20	0.887	2.93
9	0.817	3.92	0.699	6.91	0.753	3.61	0.838	4.19	0.586	5.14	0.901	2.86
10	0.787	3.48	0.656	5.36	0.877	4.74	0.816	3.36	0.684	6.03	0.901	3.11
11	0.846	3.69	0.682	6.17	0.799	3.76	0.810	3.88	0.664	5.66	0.901	3.60
12	0.807	3.57	0.675	6.10	0.860	4.21	0.803	3.93	0.621	6.24	0.897	3.17
13	0.829	3.17	0.663	6.38	0.744	3.90	0.828	3.56	0.705	4.76	0.899	3.23
14	0.815	3.40	0.696	6.38	0.760	4.25	0.769	3.71	0.638	5.52	0.894	3.05
15	0.834	3.39	0.692	5.79	0.736	3.71	0.888	3.24	0.584	5.08	0.905	3.47
Mean	0.823	3.52	0.689	6.05	0.800	3.96	0.830	3.65	0.636	5.57	0.896	3.19

4.3. Hardware Platform Test

Generally, our method is purely software-based, and no specific hardware is required. Hence, our method is easily ported to many hardware platforms. In this section, our method was implemented on three hardware platforms based on ARM CPU, FPGA, and DSP architectures to analyze the feasibility of our method on embedded platforms. The ARM CPU is Qualcomm's Snapdragon 865 mobile platform. The CPU and operating system in this platform are a quadcore Kryo@ 2.84 GHz and Android 10.0. Our target FPGA platform is the Xilinx Virtex-7 XC7VX690T FPGA with 5 million gates and 52-Mbit SRAM memory for data exchange. The DSP is the DaVinci digital media processor DM648 with a 1.1 GHz maximum processing clock rate. Three implementations of the proposed method were conducted for these platforms: Java for ARM CPU, VHDL for FPGA, and C language for DSP. Table 3 summarizes the average computational time cost for different hardware platforms when running our method on the TMQI database with six kinds of resolution.

Table 3. The average computational time cost when running implementations on different hardware platforms.

Resolution	357 × 535	512 × 380	401 × 535	720 × 480	713 × 535	803 × 535
ARM CPU	27.24 ms	27.75 ms	30.60 ms	49.29 ms	54.41 ms	61.28 ms
FPGA	11.31 ms	11.53 ms	12.71 ms	20.47 ms	22.60 ms	25.45 ms
DSP	10.48 ms	10.69 ms	11.78 ms	18.97 ms	20.95 ms	23.59 ms

This is an expected result and the proposed method can be implemented on all three platforms. It can be seen that both FPGA and DSP platforms achieve real-time performance compared with ARM CPU. For the largest spatial resolution (803 × 535), our method runs on FPGA and DSP platforms with a speed of about 39 FPS and 42 FPS. Since our method is very dependent on the multiplier, the runtime of the ARM CPU is markedly lower than the other platforms.

5. Conclusions

This paper focuses on reproducing real-world luminance on a conventional display. The radiance map is obtained by global luminance mapping and local contrast preserving. The piecewise linear mapping is introduced to achieve global dynamic range compression. For preserving local contrast, this work computes each pixel by a weighted average of its neighborhood pixels, which luminance values are known by the piecewise linear mapping. We compare the performance of our algorithm and 5 existing state-of-the-art tone mapping algorithms. Each tone mapping algorithm was performed on 3 famous HDR images and the TMQI database, respectively. The objective metric TMQI and subjective evaluation MOS are employed to evaluate every tone-mapped image. The proposed algorithm obtained the best TMQI, MOS, and NIQE among all the tone mapping algorithms, which indicates that our method can keep the quality of tone-mapped images close to the real-world scene and produce high visual quality and natural-looking images under various real-world scenes.

Author Contributions: Conceptualization, L.Z.; methodology, L.Z. and J.W.; software, R.S.; validation, L.Z., R.S. and J.W.; formal analysis, R.S.; investigation, R.S.; resources, J.W.; data curation, L.Z.; writing—original draft preparation, R.S.; writing—review and editing, J.W.; visualization, J.W.; supervision, L.Z.; project administration, L.Z. and J.W.; funding acquisition, L.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Quzhou Science and Technology Plan Project, grant number 2022K108 and Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014.

Acknowledgments: The authors acknowledge Quzhou Science and Technology Plan Project (grant number 2022K108), Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014), and Basic Scientific Research Foundation Project of Provincial Colleges and Universities in Heilongjiang Province (2022KYYWF-FC05).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kalantari, N.K.; Ramamoorthi, R. Deep high dynamic range imaging of dynamic scenes. *ACM Trans. Graph.* **2017**, *36*, 144. [CrossRef]
2. Ou, Y.; Ambalathankandy, P.; Takamaeda, S.; Motomura, M.; Asai, T.; Ikebe, M. Real-time tone mapping: A survey and cross-implementation hardware benchmark. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 2666–2686. [CrossRef]
3. Yue, G.; Yan, W.; Zhou, T. Referenceless quality evaluation of tone-mapped HDR and multiexposure fused images. *IEEE Trans. Ind. Inform.* **2019**, *16*, 1764–1775. [CrossRef]
4. Jiang, M.; Shen, L.; Hu, M.; An, P.; Gu, Y.; Ren, F. Quantitative measurement of perceptual attributes and artifacts for tone-mapped HDR display. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–11. [CrossRef]
5. Fattal, R.; Lischinski, D.; Werman, M. Gradient domain high dynamic range compression. In Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, 23–26 July 2002.

6. Eilertsen, G.; Mantiuk, R.K.; Unger, J. A comparative review of tone-mapping algorithms for high dynamic range video. *Comput. Graph. Forum* **2017**, *36*, 565–592. [CrossRef]
7. Ashikhmin, M. A tone mapping algorithm for high contrast images. In Proceedings of the Eurographics Workshop on Rendering, Pisa, Italy, 26–28 June 2002.
8. Zai, G.J.; Liu, Y. An improved tone mapping algorithm for high dynamic range images. In Proceedings of the International Conference on Computer Application and System Modeling, Taiyuan, China, 22–24 October 2010.
9. Tumblin, J.; Rushmeier, H. Tone reproduction for realistic images. *IEEE Comput. Graph. Appl.* **1993**, *13*, 42–48. [CrossRef]
10. Ward, G. A contrast-based scalefactor for luminance display. *Graph. Gems* **1994**, *4*, 415–421.
11. Larson, G.W.; Rushmeier, H.; Piatko, C. A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Trans. Vis. Comput. Graph.* **1997**, *3*, 291–306. [CrossRef]
12. Khan, I.R.; Aziz, W.; Shim, S.O. Tone-mapping using perceptual-quantizer and image histogram. *IEEE Access* **2020**, *8*, 31350–31358. [CrossRef]
13. Meylan, L.; Süsstrunk, S. High dynamic range image rendering using a retinex-based adaptive filter. *IEEE Trans. Image Process.* **2006**, *15*, 2820–2830. [CrossRef] [PubMed]
14. Gu, B.; Li, W.; Zhu, M.; Wang, M. Local edge-preserving multiscale decomposition for high dynamic range image tone mapping. *IEEE Trans. Image Process.* **2012**, *22*, 70–79. [PubMed]
15. Kuang, J.; Johnson, G.M.; Fairchild, M.D. iCAM06: A refined image appearance model for HDR image rendering. *J. Vis. Commun. Image Represent.* **2007**, *18*, 406–414. [CrossRef]
16. Rana, A.; Singh, P.; Valenzise, G.; Dufaux, F.; Komodakis, N.; Smolic, A. Deep tone mapping operator for high dynamic range images. *IEEE Trans. Image Process.* **2019**, *29*, 1285–1298. [CrossRef] [PubMed]
17. Panetta, K.; Kezebou, L.; Oludare, V.; Agaian, S.; Xia, Z. TMO-Net: A parameter-free tone mapping operator using generative adversarial network, and performance benchmarking on large scale HDR dataset. *IEEE Access* **2021**, *9*, 39500–39517. [CrossRef]
18. Patel, V.A.; Shah, P.; Raman, S. A generative adversarial network for tone mapping hdr images. In Proceedings of the National Conference on Computer Vision, Pattern Recognition, Image Processing, and Graphics, Mandi, India, 16–19 December 2017.
19. Yeganeh, H.; Wang, Z. Objective quality assessment of tone-mapped images. *IEEE Trans. Image Process.* **2012**, *22*, 657–667. [CrossRef] [PubMed]
20. Reinhard, E.; Stark, M.; Shirley, P.; Ferwerda, J. Photographic tone reproduction for digital images. *ACM Trans. Graph.* **2002**, *22*, 267–276. [CrossRef]
21. Reinhard, E. Parameter estimation for photographic tone reproduction. *J. Graph. Tools* **2002**, *7*, 45–51. [CrossRef]
22. Thai, B.C.; Mokraoui, A.; Matei, B. Contrast enhancement and details preservation of tone mapped high dynamic range image. *J. Vis. Commun. Image Represent.* **2019**, *58*, 589–599. [CrossRef]
23. Liang, Z.; Xu, J.; Zhang, D.; Cao, Z.; Zhang, L. A hybrid l1-l0 layer decomposition model for tone mapping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
24. Narwaria, M.; Da Silva, M.P.; Le Callet, P.; Pépion, R. Tone mapping based HDR compression: Does it affect visual experience? *Signal Process. Image Commun.* **2014**, *29*, 257–273. [CrossRef]
25. Mittal, A.; Soundararajan, R.; Bovik, A.C. Making a "completely blind" image quality analyzer. *IEEE Signal Process. Lett.* **2012**, *20*, 209–212. [CrossRef]

electronics

MDPI

Article

Furniture Image Classification Based on Depthwise Group Over-Parameterized Convolution

Han Ye [1], Xiaodong Zhu [1], Chengyang Liu [2], Linlin Yang [2] and Aili Wang [2,*]

[1] College of Materials Science and Engineering, Northeast Forestry University, Harbin 150040, China
[2] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China
* Correspondence: aili925@hrbust.edu.cn

Abstract: In this paper, an improved VGG16 combined with depthwise group over-parameterized convolution (DGOVGG16) model is proposed to realize automatic furniture image classification. Firstly, depthwise over-parameterized convolution combined with group convolution is combined to construct depthwise group over-parameterized convolution, which is introduced to the VGG 16 model for reducing the number of parameters of the overall model while extracting more sufficient semantic features of furniture images. Then, this paper uses the ReLU activation function in the former part of the neural network to reduce the correlation between parameters and accelerate the weight update speed of the former part of the model. Meantime, the proposed model applies Leaky-ReLU activation function in the last layer to avoid the problem that some neurons do not update. Compared with the six furniture image classification methods based on MobileNetV2, AlexNet, ShuffleNetv2, GoogleNet, VGG 16 and GVGG16, the experimental results show the proposed DGOVGG16 with average accuracy (AA) of 95.51% has better classification performance.

Keywords: furniture image classification; deep learning; VGG16; group convolution; depthwise over-parameterized convolution

1. Introduction

With the development of the Internet technology and big data, the number of e-commerce websites has increased dramatically. In the face of such a huge transaction volume, furniture images have become the main information carrier of commodity information transmission. Facing a large number of furniture products, automatic recognition of furniture has become an inevitable requirement for intelligent information extraction in e-commerce [1]. The maturity and development of this technology help businesses automatically identify uploaded furniture images according to color and style. Saving labor consumption can help users quickly search for desired furniture to improve the efficiency of the website and can also help businesses more accurately recommend furniture to improve the sales of furniture. At present, most scholars in the research of furniture image recognition task classify the furniture according to the types of tables, chairs, beds, etc, especially in complex background [2,3].

Traditional image classification methods focus on the extraction of texture, color and other features of the image and select an appropriate classifier to classify the extracted features. With the development of big data technology and the computer hardware performance, deep learning (DL) has become a research hotspot of computer theory [4]. The image classification methods based on convolutional neural network (CNN) are deeply loved by industry and academia [5,6].

The realization of image classification technology is divided into traditional image classification methods and those based on CNN. The two types of recognition methods are composed of two stages: image processing and recognition classification. In the image processing stage, the image to be recognized is input into the system, and the

image is preprocessed appropriately to improve the image quality. Traditional methods usually use special feature extraction algorithms to extract image features and reduce the dimensions of the features to obtain digital features that can accurately describe the image. In the recognition and classification stage, the corresponding statistical learning and other classification decision-making methods are mainly used to match, identify and classify the extracted features. Finally, the classification results are output.

At present, people have tried to use DL to classify furniture images. The convolution machine network does not require complicated feature extraction operations. It uses its own convolution layer and pooling layer to extract features and reduce dimensions, and uses the full connection layer as a classifier to complete image recognition tasks. Zhu Bin [7] et al. studied the emotional intention recognition of chairs based on convolutional neural network technology and built a product image recognition model. Wang Y et al. [8] proposed a metric learning algorithm combined with CNN to remove duplicate and irrelevant samples in the furniture image database. Zhu Haipeng [9] et al. took the chair as an example, and summarized four furniture styles of the East, the West, modern and new technology through expert evaluation and cluster analysis. They used the classic ReSNet50 model to recognize the chair image and obtained good recognition accuracy.

In this paper, we combine the advantages of group convolution and depthwise over-parameterized convolution (DO-Conv), proposed depthwise group over-parameterized convolution, which is introduced into VGG16 to improve the classification performance of furniture images. In order to fully mine the semantic information of furniture under complex background, we have carried out many experiments and analyses to prove which layer of VGG16 can get the best classification accuracy. Section 2 introduces the related work of group convolution and depthwise over-parameterized convolution and Section 3 gives the furniture image classification model based on DGOVGG16. In Section 4, the proposed DGOVGG16 model is validated compared to six other classification models. Finally, we gave the conclusions at the end of this paper.

2. The Related Work

2.1. Group Convolution

The group convolution (GC) first appeared in AlexNet, further optimized in ResNeXt network, and solved the problem of insufficient video memory [10]. The first advantage of group convolution is efficient training. Because the convolution is divided into multiple paths, each path can be processed separately by different GPUs, so the model can be trained on multiple GPUs in parallel. Compared with completing all tasks on a single GPU, this model parallelization on multiple GPUs enables the network to process more images at each step. It is generally believed that model parallelization is better than data parallelization. The latter is to divide the dataset into multiple batches, and then train each batch separately. However, when the batch size becomes too small, we essentially perform random gradient descent, not batch gradient descent. This results in slower and sometimes worse convergence results. When training very deep neural networks, packet convolution is very important, as in ResNeXt.

Meantime, with the same convolution operation, GC requires fewer parameters than ordinary convolution to avoid overfit [11]. The difference between ordinary convolution (upper part) and GC (lower part) is as shown in Figure 1, which means the model parameters will decrease as the number of filter groups increases.

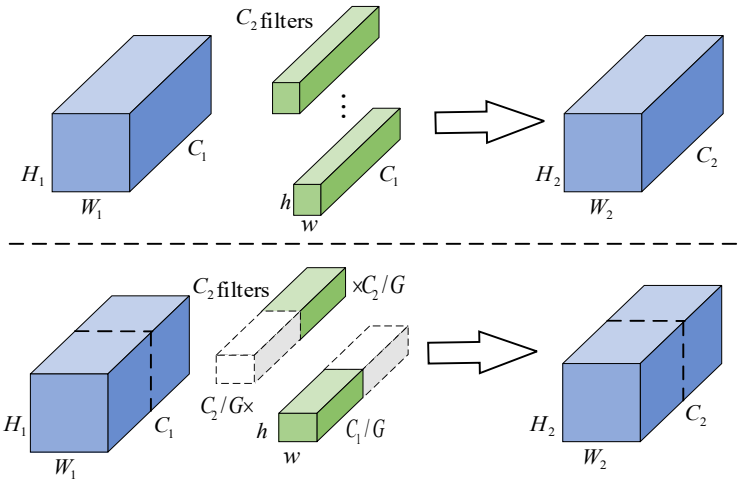


Figure 1. The structure of group convolution.

The GC first groups the input data, and then convolves each group of data separately. The input feature map size of C_1 are divided into G groups as follows:

$$\begin{bmatrix} H_1(x) \\ H_2(x) \\ \vdots \\ H_G(x) \end{bmatrix} = \begin{bmatrix} W_{11}^P & 0 & 0 & 0 \\ 0 & W_{11}^P & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & W_{GG}^P \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_G \end{bmatrix} \tag{1}$$

$$F(x) = \sum_{i=1}^{G} H_i(x) \tag{2}$$

where, $[x_1, x_2, \cdots, x_G]$ is the input feature map which is divided into groups size of G, each group has G_1/G feature map size of $H_1 \times W_1$; W_{ii}^P is the convolution kernel of each group with the number of $P = C_2/G$ and size of $h \times w$; $H_i(x)$ is the output feature maps of each group size of $H_2 \times W_2$. Then, all the output feature maps are merged by channel to get the final output of the convolution layer.

If $C_1 = C_2 = C$ which means the input and output feature maps of the above GC are equal, T_{GC} is the number of parameters, which can be calculated as follows:

$$T_{GC} = G \times (\frac{C}{G} \times h \times w \times \frac{C}{G}) = \frac{h \times w \times C^2}{G} \tag{3}$$

In the same case, the number of parameters of ordinary convolution named as T_{RC} can be calculated as follows:

$$T_{RC} = C \times h \times w \times C = h \times w \times C^2 \tag{4}$$

By comparison of T_{GC} and T_{RC}, the parameter amount of the group convolution is 1/G of the standard convolution, which means G is larger, the complexity is lower and the parameter efficiency is higher.

2.2. Depthwise Over-Parameterized Convolution

Li et al. proposed DO-Conv to speed up the convergence of the model, while adding learnable parameters to the model without increasing the computational complexity [12]. DO-Conv is a combination of an ordinary convolution $W \in R^{C_{out} \times D_{mul} \times C_{in}}$ and a deep convolution $D \in R^{(M \times N) \times D_{mul} \times C_{in}}$. Figure 2 shows the flowchart of DO-Conv. P is the size of the input feature map. M and N are the spatial dimensions of P, C_{in} is the number of input feature maps, D_{mul} is the number of deep convolutions, C_{out} is the number of output feature maps, $D^T \in R^{D_{mul} \times (M \times N) \times C_{in}}$ is the transposition of $D \in R^{(M \times N) \times D_{mul} \times C_{in}}$,

and W' is the convolution kernel of. First, combine the deep convolution kernel D^T and conventional convolution kernel W to form W', and then output O by convolution of W' and P, as shown in Formula (5).

$$\begin{aligned} O &= (D, W) \circledast P \\ &= (D^T \circ W) * P \end{aligned} \tag{5}$$

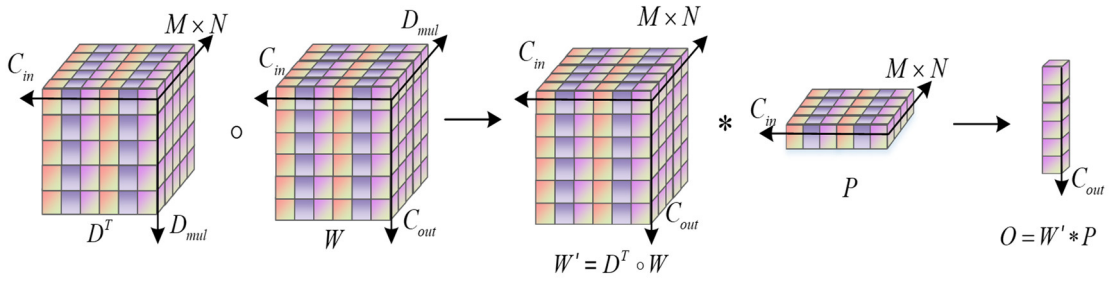


Figure 2. The flowchart of DO-Conv.

2.3. Activation Function

In the DGOVGG16 model, in order to ensure that the weight of the model is updated quickly, alleviate the problem of over fitting, and at the same time avoid losing part of the characteristic information in the negative signal, the ReLU and Leaky ReLU activation function are used in the model to activate neurons. The ReLU activation function is introduced in the previous part of the neural network, so that the output of some neurons is 0, reducing the correlation between parameters and accelerating the weight's update speed of the previous part of the model. The ReLU activation function written is as follows:

$$\text{ReLU} = \begin{cases} x_i & x_i > 0 \\ 0 & x_i \leq 0 \end{cases} \tag{6}$$

However, if all of the ReLU activation functions are replaced with Leaky ReLU activation functions, the updating speed of the whole model will be reduced. Therefore, this paper only introduces a Leaky value into the last layer of activation functions. The formula of Leaky ReLU function is as follows:

$$\text{Leaky-ReLU} = \begin{cases} x_i & x_i > 0 \\ a_i x_i & x_i \leq 0 \end{cases} \tag{7}$$

where $a_i = 0.01$ is a very small weight, that is, the negative input information is retained without increasing the calculation cost of model training, and the input information is retained more completely.

3. Proposed Methods

The improved VGG16 based on depthwise group over-parameterized convolution (DGOVGG16) model for furniture classification is shown in Figure 3.

First, the furniture data are put into the DGOVGG16 model, and the first 11 layers of the model are used to extract the deep features of the furniture data. The last two layers are divided into depthwise group over-parameterized convolution and conventional convolution layers, respectively. Through depthwise group over-parameterized convolution, more precise features are captured while reducing the number of parameters to ensure the accuracy of the network model to the maximum extent. Secondly, the output features are mapped to the full connection layer through the conventional convolution layer, and after each full connection layer, the dropout layer is introduced, which temporarily discards some neurons from the network at a certain rate, thus eliminating or weakening the con-

nection between neuron nodes, reducing the phenomenon of over fitting, and outputting the final category through the last full connection layer.

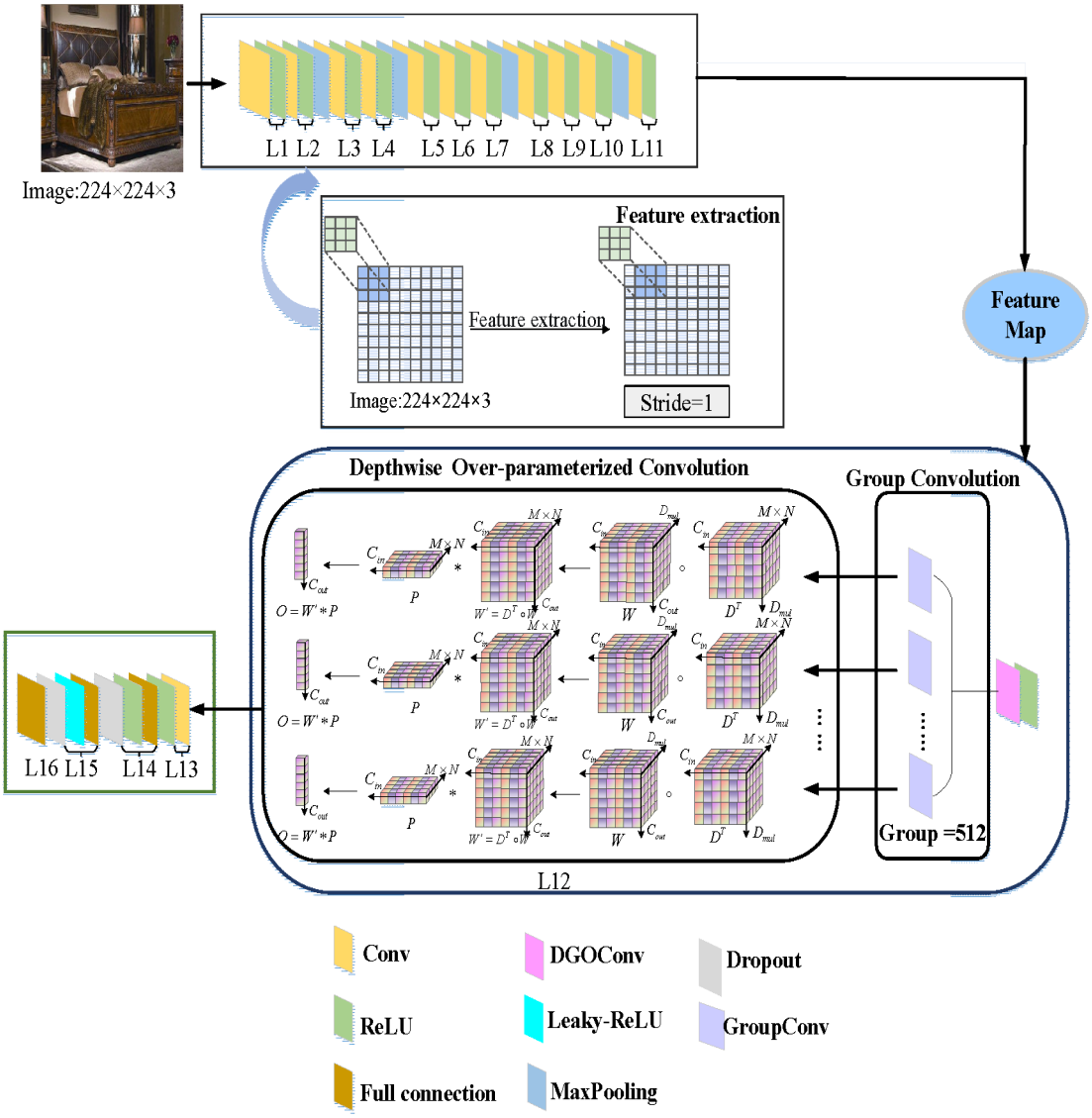


Figure 3. The proposed DGOVGG16 model of furniture image classification.

3.1. DGOVGG16 Model

For the selection of convolutional neural network model, the recognition accuracy and the training convergence efficiency of the model must be comprehensively considered. Generally speaking, the deeper the network model is, and the larger the scale is, the higher the accuracy of image recognition will be. However, the more the corresponding model parameters are, the lower the recognition efficiency will be. Especially when the data set is insufficient, not only the risk of over fitting exists, but also the difficulty of model training will increase difficulty in convergence and other problems. The architecture of the proposed DGOVGG16 network is shown in Figure 4a, and the convolution processing is shown in Figure 4b.

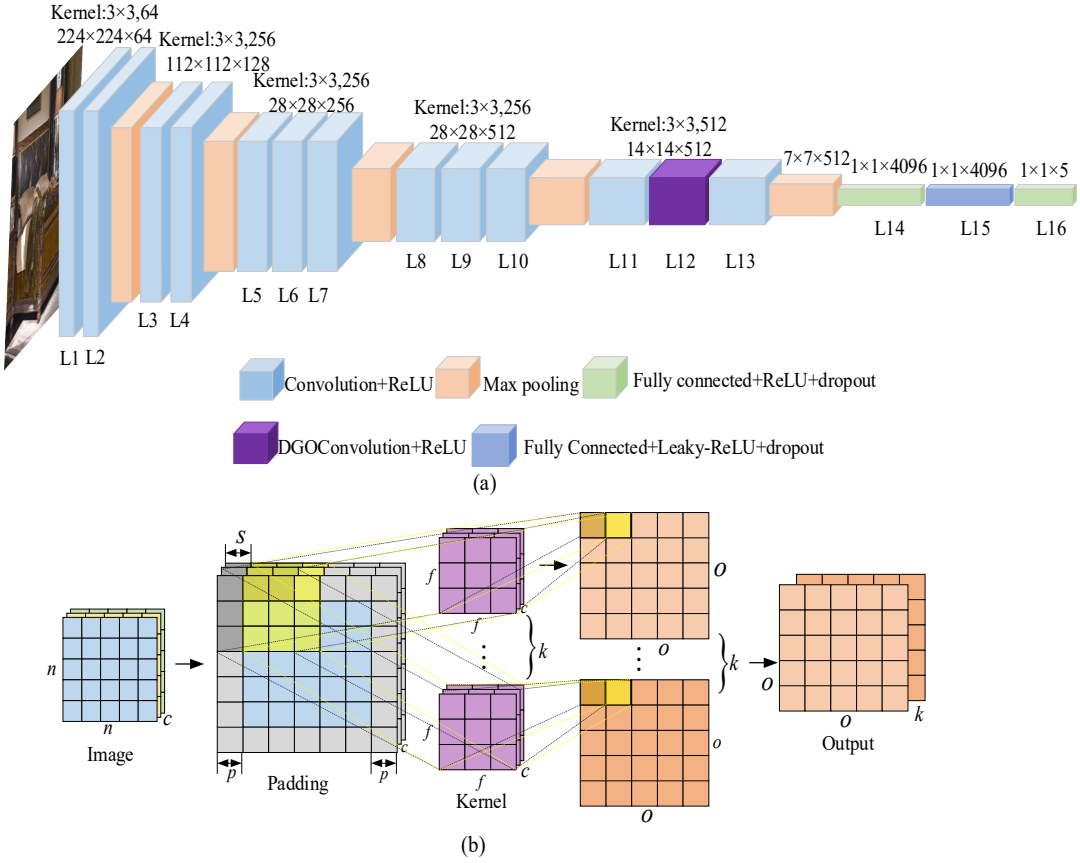


Figure 4. The designed DGVGG16 model details of each convolutional layer (**a**) DGOVGG16 structure. (**b**) convolution processing.

As there is no large-scale image public data set for furniture data set, the accuracy and efficiency of actual furniture classification are also considered. Therefore, this paper chooses to improve the VGG16 network model, which has the advantages of smaller convolution kernel, fewer parameters and strong nonlinear fitting ability. DGOVGG16 model stacks 3×3 small size convolution kernel to replace large convolution kernel for achieving the purpose of using fewer parameters to extract more features, and 2×2 to retain more image information. The number of convolution kernel in the 15 convolution layers is 64, 64, 128, 128, 256, 256, 256, 512, 512, 512, 512, 512, 512, 512, 512, respectively. The number of neurons in the two fully connected layers are 4096, and the number of neurons in the last output layer is 5. The pooling layer adopts the maximum pooling method, and the two full connection layers adopt ReLU and Leaky ReLU activation function respectively. Among them, the ReLU function is easy to derive and fast to calculate, and make the output of some neurons zero, thus reducing the correlation between parameters, which can effectively alleviate the occurrence of overfitting problems. At the same time, the introduction of Leaky ReLU activation function in the last layer can effectively prevent some neurons from being invalid when the final input is negative and ensure the final accurate output results.

At the same time, in order to further prevent overfitting, a dropout layer is added to the DGOVGG16 model. Its main idea is to temporarily discard some neurons from the network according to a certain ratio, so as to eliminate or weaken the connection between neural nodes. The dropout process reduces the size of the parameters involved in training. If the total number of neural nodes in the convolutional network structure is n, the combination size of 2^n in theory can be constructed. Such a combination can not only increase the size of the parameters, but also improve the generalization of the network. Moreover, dropout has a great probability to create different combinations of neurons working together each time.

The combination of these different neurons reduces the node correlation, thus enhancing the generalization ability of the entire model.

3.2. Depthwise Group Over-Parameterized Convolution

Although VGG16 model itself has the advantages of smaller convolution kernel and fewer parameters, there is still a risk of over fitting with the deepening of network layers.

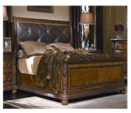
Therefore, in the 12th layer of VGG16 model, this paper introduces depthwise group over-parameterized convolution as shown in Figure 3, which is composed of two parts, namely, group convolution and DO-Conv. Among them, the number of parameters of the network model is minimized by grouping to avoid overfitting. The reduction of the number of parameters means that the complexity of the model is reduced. The lower the complexity, the lower the accuracy of the model will also decrease. Therefore, this paper conducts depthwise over-parameterized convolution for each input channel, which can fully extract the semantic features of the furniture image ensuring that the overall model to extract more sufficient semantic features while reducing the amount of parameters.

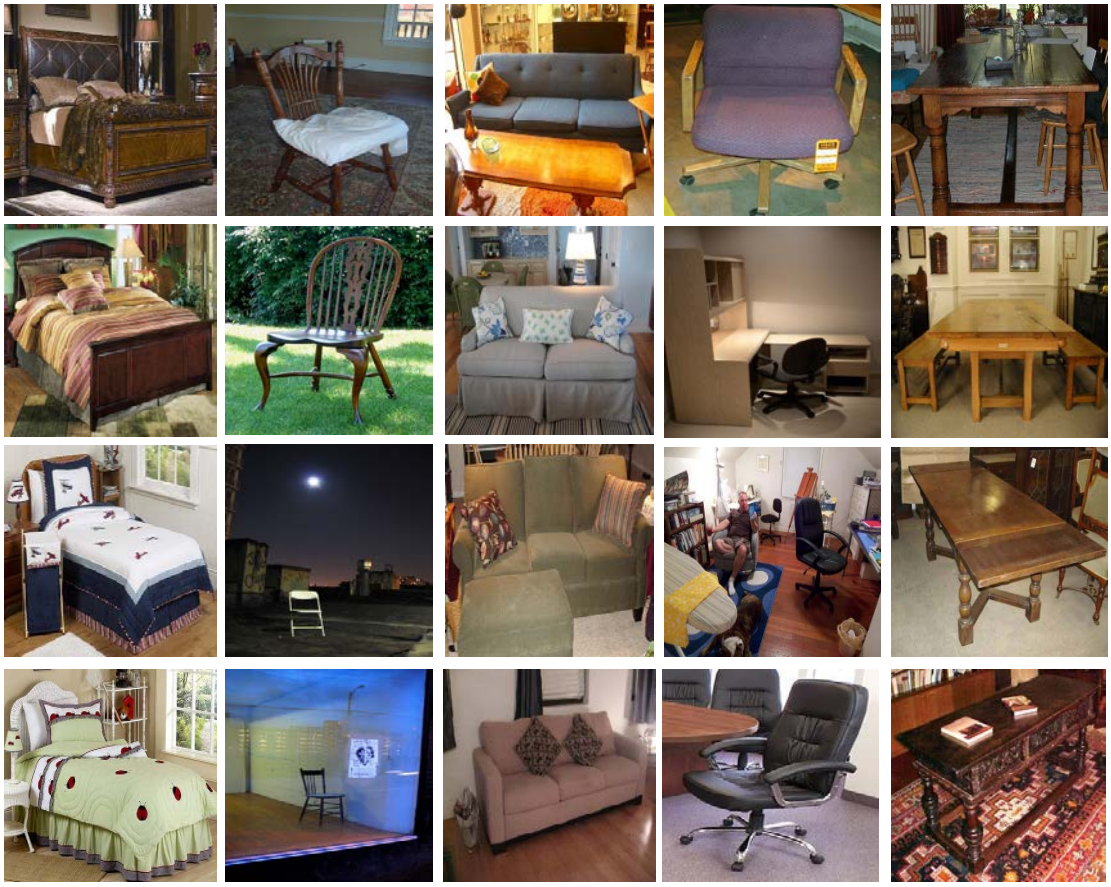
4. Experimental Results and Analysis

4.1. Furniture Dataset Description

The furniture dataset used in this paper is the open data set of Kaggle competition, which was released by Nikhil Akki at Deloitte, a data science engineer from Mumbai, India, in 2018. Some images of furniture data are shown in Figure 5, which can be divided into five categories: bed, chair, sofa, swivelhair, and table. It can be seen from Figure 5 that the background of furniture is usually complex, including invalid information unrelated to furniture such as walls, windows, curtains, external environment, people, etc. Among them, there also exists that chair appears in the table sample as shown the last column. Therefore, when classifying furniture images, it is necessary to take into account the complexity of its background and the diversity of furniture features and design an accurate classification model of furniture in a complex background. The specific number of each type of furniture is shown in Table 1.

Table 1. The description of furniture dataset.

Sample Image	Class	Number
	bed	1100
	chair	1100
	sofa	1100
	swivelchair	1100
	table	470



(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 5. The samples of the furniture dataset. (**a**) bed. (**b**) chair. (**c**) sofa. (**d**) swivelchair. (**e**) table.

4.2. Experimental Platform Parameter Settings

The experimental environment was Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10 GHz processor, 128 GB running memory (RAM), NVIDIA GeForce RTX 2080Ti GPU. In addition, the deep learning framework was Pytorch. In the experiments, the size of the input data is set to 224, the number of training epochs is set to 30, and SGD optimization algorithm is used for optimization. The momentum is set to 0.9, and the learning rate is set to 0.1. All experimental training samples are randomly selected from furniture dataset and the ratio of training set to test set is 10:1. Meanwhile, this paper uses six different methods for comparison and analysis, including MobileNetV2, AlexNet, ShuffleNetv2, GoogLeNet [13], VGG16 and VGG16 network based on group convolution (GVGG16). The average accuracy (AA) commonly is used in furniture image classification as the evaluation index to better evaluate the proposed DGOVGG16 model's classification performance of each category.

4.3. Design of DGOVGG16 Model

This paper takes DGOVGG16 as the main feature extraction network, which consists of 15 ReLU layers, 12 convolution layers, 5 MaxPool layers, 3 full connection layers, 1 over-parameterized convolution layer, 1 Leaky-ReLU layer, and 2 dropout layers. In order to make DGOVGG16 model use fewer parameters to represent more features, this paper sets the size of convolution kernel as 3×3 to replace the large convolution kernel by stacking small convolution kernels, and the size of the largest pooling layer is 2×2 to retain more image information. The number of neurons in the three full connection layers was 4096, 4096 and 5, respectively. When the dropout rate is 0.5, the experimental accuracy is the highest. Table 2 shows the parameter settings of DGOVGG16 network model.

Table 2. The parameter settings of DGOVGG16 model.

Layer	Kernel	MaxPool	ReLU	Dropout	Leaky-ReLU
Conv1	3 × 3 (stride = 1)	No	Yes	No	No
Conv2	3 × 3 (stride = 1)	2 × 2 (stride = 2)	Yes	No	No
Conv3	3 × 3 (stride = 1)	No	Yes	No	No
Conv4	3 × 3 (stride = 1)	2 × 2 (stride = 2)	Yes	No	No
Conv5	3 × 3 (stride = 1)	No	Yes	No	No
Conv6	3 × 3 (stride = 1)	No	Yes	No	No
Conv7	3 × 3 (stride = 1)	2 × 2 (stride = 2)	Yes	No	No
Conv8	3 × 3 (stride = 1)	No	Yes	No	No
Conv9	3 × 3 (stride = 1)	No	Yes	No	No
Conv10	3 × 3 (stride = 1)	2 × 2 (stride = 2)	Yes	No	No
Conv11	3 × 3 (stride = 1)	No	Yes	No	No
DOConv12	3 × 3 (stride = 1) (groups = 512)	No	Yes	No	No
Conv13	3 × 3 (stride = 1)	2 × 2 (stride = 2)	Yes	No	No
FC	Neurons = 4096	No	Yes	Yes	No
FC	Neurons = 4096	No	No	Yes	Yes
FC	Neurons = 5	No	No	No	No

4.4. The Selection of Depthwise Group Over-Parameterized Convolutional Layer

The group convolution in depthwise group over-parameterized convolution applied over-parameterized convolution to the input grouped channels. In order to minimize the parameter amount of the proposed model, this paper sets the number of groups divided as the number of input channels. From the number of channels input at each layer, it can be found that layers 2 to 3 are equal, layers 4 to 5 are equal, layers 6 to 8 are equal, and layers 9 to 13 are equal. Moreover, the features extracted from the next layer are more accurate than those extracted from the previous layer. Thus, this paper introduces depthwise group over-parameterized convolution into layer 3, layer 5, layer 8 and layer 12 respectively for comparison, and selects the most appropriate number of layers. It can be seen from Table 3 that when the depthwise group over-parameterized convolution is put into the 12th layer, the AA growth reaches the peak, for 12th layer extracts more refined furniture semantic features with stronger discrimination. Thus, this paper sets the location of depthwise group over-parameterized convolution as the 12th layer.

Table 3. The classification accuracy comparison of depthwise group over-parameterized convolution in different layer.

Classes \ Layer	3	5	8	12
bed	81.00	96.00	83.00	96.00
chair	91.00	87.00	90.00	98.00
sofa	84.00	93.00	94.00	95.00
swivelchair	85.00	88.00	88.00	94.00
table	91.30	82.61	86.96	94.56
AA(%)	86.46	89.32	88.40	95.51

4.5. Comparison of Experimental Results

This paper makes a comparative analysis of the two evaluation indicators: AA and running time. Table 4 shows that DGOVGG16 model is superior to other six classification

models. The comparison between DGOVGG16 and GVGG16 shows that DGOVGG16 performs better than the latter. The DO-Conv is introduced to reduce the number of model parameters while fully extracting the semantic features of furniture, reducing the problem of model overfitting. At the same time, Leaky-ReLU activation function is introduced to retain the update weight speed of the whole model. Therefore, the DGOVGG16 model proposed in this paper has a stronger classification accuracy compared with other classification methods.

Table 4. The comparison of furniture classification results based on different network models.

Classes	MobileNetV2	AlexNet	ShuffleNetv2	GoogLeNet	VGG16	GVGG16	DGOVGG16
bed	94.40	88.60	94.00	95.40	93.40	91.00	96.00
chair	94.60	95.20	97.00	95.80	96.40	93.00	98.00
sofa	88.00	88.60	92.00	95.20	92.20	92.00	95.00
swivelchair	89.60	91.00	89.00	88.00	95.00	93.00	94.00
table	88.04	91.30	91.30	89.56	95.65	91.30	94.56
AA (%)	90.92	90.94	92.66	92.79	94.53	92.06	95.51
Time(s)	729	675	688	722	769	746	747

It can be seen from Table 4 that in the furniture dataset classification experiment of DGOVGG16, AA reached 95.51% at the highest, and increased by 3.45%, 0.98%, 2.72%, 2.85%, 4.57% and 4.59% respectively compared with GVGG16, VGG16, GoogLeNet, ShuffleNetv2, AlexNet and MobileNetV2. AlexNet model is the classification method with the shortest running time. It can be found that the running time of DGOVGG16 is 72 s longer than that of AlexNet, but the accuracy is increased by 4.57%, thus a slight increase in time is acceptable. Figures 6 and 7 show the classification results of different methods and each category on the furniture dataset respectively.

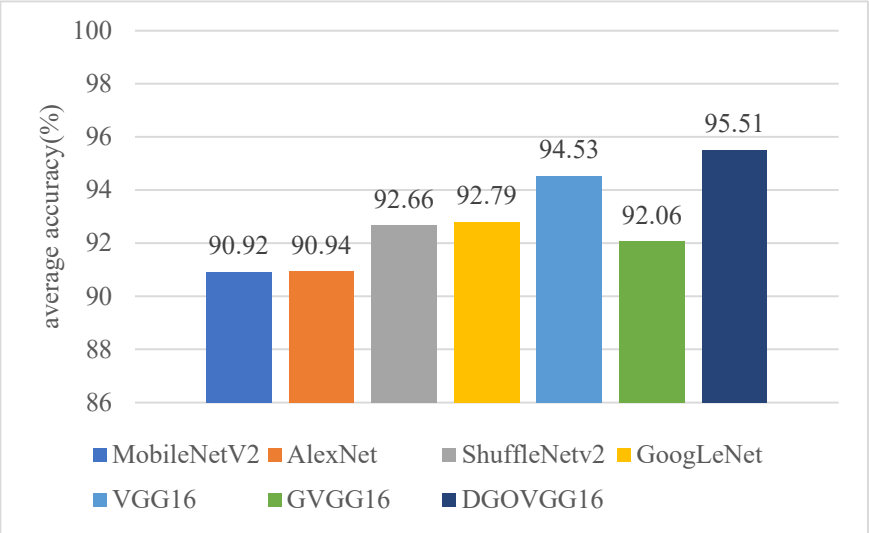


Figure 6. Classification accuracy comparison for different models on the furniture dataset.

It can be seen from Figure 7 that the accuracy of each category of DGOVGG16 model has been significantly improved compared with that of GVGG16 and VGG16, GoogLeNet, AlexNet, MobileNetV2 and ShuffleNetv2 methods. Although some categories are not of the highest accuracy, the overall performance of DGOVGG16 is the best.

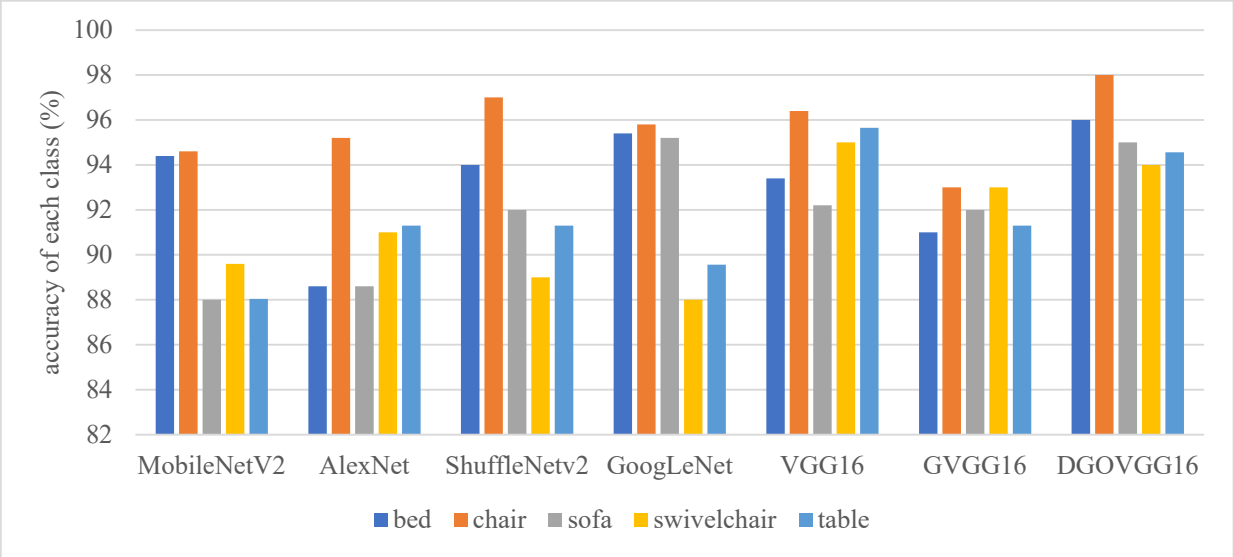


Figure 7. Classification accuracy of each class in the furniture dataset.

5. Conclusions

This paper proposes a furniture classification algorithm based on VGG16 network combined with depthwise group over-parameterized convolution. In VGG16 model, depthwise group over-parameterized convolution is introduced to reduce the number of parameters of the overall model while extracting more sufficient semantic features, which reduces the risk of model overfitting and improves the furniture image classification accuracy of the model, especially in complex backgrounds situation. In addition, in the selection of activation function, this paper used ReLU activation function in the former part of the neural network, reduced the correlation between parameters, accelerated the weight update speed of the former part of the model, and applied Leaky ReLU activation function in the last layer to avoid the problem that some neurons do not update.

This paper conducts experiments on the furniture public data set and compares DGOVGG16 with other six popular classification network methods. The AA of the proposed DGOVGG16 in this paper reaches 95.51%, which is superior to other classical methods, which fully shows that our model can accurately classify furniture images under complex backgrounds with stronger competitiveness. In future work, we intend to mine and fuse the global and local features of furniture at different scales improve the proposed DGOVGG16 model, so that the classification accuracy between similar furniture images can be further improved.

Author Contributions: Conceptualization, H.Y. and X.Z.; methodology, H.Y., X.Z. and A.W.; software, H.Y., C.L. and L.Y.; validation, X.Z., H.Y., C.L. and L.Y.; writing—review and editing, X.Z. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research is funded by "Belt and Road Initiative" Innovative Talent Exchange Foreign Experts Project (G2022012010).

Data Availability Statement: https://www.kaggle.com/datasets/akkithetechie/furniture-detector/download?datasetVersionNumber=1 (accessed on 2 November 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
2. Mo, K.; Zhu, S.; Chang, A.X.; Yi, L.; Tripathi, S.; Guibas, L.J.; Su, H. Partnet: A large-scale benchmark for fifine-grained and hierarchical part-level 3D object understanding. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

3. Varvadoukas, T.; Giannakidou, E.; Gomez, J.V.; Mavridis, N. Indoor Furniture and Room Recognition for a Robot Using Internet-Derived Models and Object Context. In Proceedings of the 10th International Conference on Frontiers of Information Technology, Islamabad, Pakistan, 17–18 December 2012.
4. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
5. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
6. Xiao, J.; Hays, J.; Ehinger, K.A.; Oliva, A.; Torralba, A. SUN database: Large-scale scene recognition from abbey to zoo. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 3485–3492.
7. Zhu, B.; Yang, C.; Yu, C.; An, F. Product Image Recognition Based on Deep Learning. *J. Comput. -Aided Des. Comput. Graph.* **2018**, *30*, 1778–1784. [CrossRef]
8. Wang, Y.; Gao, W.; Wang, Y. Application of furniture images selection based on neural network. In Proceedings of the 6th International Conference on Computer-Aided Design, Manufacturing, Modeling and Simulation (CDMMS 2018), Busan, Republic of Korea, 14–15 April 2018.
9. Zhu, H.; Li, X.; Huang, W.; Li, C. Chair Style Recognition and Intelligent Design Method Based on Deep Learning. *Furniture* **2021**, *42*, 37–41.
10. Qin, B.; Gu, N.; Zhang, X. Image verification code recognition based on convolutional neural network. *Comput. Syst. Appl.* **2018**, *27*, 144–150.
11. Yan, X.; Huang, S. Lightweight target detection network based on grouped heterogeneous convolution. *Comput. Sci.* **2020**, *47*, 114–117.
12. Cao, J.; Li, Y.; Sun, M.; Yi, C.; Lischinski, D.; Cohen-Or, D.; Chen, B.; Tu, C. DO-Conv: Depthwise Over-parameterized Convolutional Layer. *IEEE Trans. Image Process.* **2022**, *31*, 3726–3736. [CrossRef] [PubMed]
13. Xie, S.; Zheng, X.; Chen, Y.; Xie, L.; Liu, J.; Zhang, Y.; Yan, J.; Zhu, H.; Hu, Y. Artifact Removal using Improved GoogLeNet for Sparse-view CT Reconstruction. *Sci. Rep.* **2018**, *8*, 1–9. [CrossRef] [PubMed]

electronics

MDPI

Article

Multi-Task Learning for Scene Text Image Super-Resolution with Multiple Transformers

Kosuke Honda [1], Masaki Kurematsu [1], Hamido Fujita [2,3,4,5,*] and Ali Selamat [3]

[1] Faculty of Software and Information Science, Iwate Prefectural University, Takizawa 020-0693, Japan
[2] Faculty of Information Technology, HUTECH University, Ho Chi Minh City 72308, Vietnam
[3] Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, Kuala Lumpur 54100, Malaysia
[4] Regional Research Center, Iwate Prefectural University, Takizawa 020-0693, Japan
[5] i-SOMET Inc., Morioka 020-0104, Japan
* Correspondence: h.fujita@hutech.edu.vn or fujitahamido@utm.my or hfujita-799@acm.org

Abstract: Scene text image super-resolution aims to improve readability by recovering text shapes from low-resolution degraded text images. Although recent developments in deep learning have greatly improved super-resolution (SR) techniques, recovering text images with irregular shapes, heavy noise, and blurriness is still challenging. This is because networks with Convolutional Neural Network (CNN)-based backbones cannot sufficiently capture the global long-range correlations of text images or detailed sequential information about the text structure. In order to address this issue, this paper proposes a Multi-task learning-based Text Super-resolution (MTSR) Network to approach this problem. MTSR is a multi-task architecture for image reconstruction and SR. It uses transformer-based modules to transfer complementary features of the reconstruction model, such as noise removal capability and text structure information, to the SR model. In addition, another transformer-based module using 2D positional encoding is used to handle irregular deformations of the text. The feature maps generated from these two transformer-based modules are fused to attempt improvement of the visual quality of images with heavy noise, blurriness, and irregular deformations. Experimental results on the TextZoom dataset and several scene text recognition benchmarks show that our MTSR significantly improves the accuracy of existing text recognizers.

Keywords: scene text image super-resolution; multi-task learning; scene text recognition; transformer; attention mechanism

1. Introduction

Today, information in images is essential in various situations, including daily life, business scenes, and medical settings. However, the information is often lost due to several factors, such as complex backgrounds, noise, blurriness, and low-resolution images. In recent years, there have been many studies to obtain high visual quality images without noise, including approaches in bio-imaging [1,2] using a scientific camera such as CMOS and approaches using super-resolution (SR) processing [3–6]. In scene text recognition (STR) which we address in this study, the visual quality of images is also an important factor because recognition performance is significantly affected by them. STR aims to convert text images into computer-readable and editable symbols, a fundamental and vital task in computer vision. This technique has been widely applied to studies for automation or efficiency, such as license plate recognition, text retrieval, and ID card recognition [7–9]. The performance of STR has improved significantly in recent years with the development of deep learning. However, images with various quality degradations, such as low resolution (LR) or blurred structures, cause significant difficulties in text recognition. Against this background, recently, there have been many studies to improve the resolution and visual quality of text images for STR.

Single image super-resolution (SISR) is a fundamental task in computer vision that reconstructs fine detail from a degraded LR image and generates a high-resolution (HR) image. Similar to STR, SISR techniques have also improved significantly with the development of deep learning. Researchers have developed many SISR methods for scene text images in the past few years, as the task named scene text image super-resolution (STISR). Several studies [10,11] adopt the deep learning-based method designed for generic SISR to STISR, proving the effectiveness of using super-resolution methods as preprocessing for STR. However, these methods are insufficient for text images with heavy quality degradation since they can not distinguish between generic and text images and take into account the specific properties of text images. Then, the deep learning based-methods specific to text images [12–15] were developed, providing significant performance improvements.

Most STISR methods [12–15] specific to text images train using TextZoom [12]. TextZoom is the first dataset for STISR containing pairs of degraded text images with high-quality text images. A representative STISR method of TSRN [12] captures sequential features in text images using Bidirectional LSTM (BLSTM) [16] and text-specific loss using a gradient profile. Another representative method, TBSRN [15], attempts to restore the text shapes using pre-recognized text semantic information. However, despite these technical advances, restoring text images that are heavily blurred or noisy, have crooked or distorted text, or used unique fonts is still challenging. One of the main reasons for this is that networks using CNN-based backbones cannot sufficiently capture global long-range correlations in text images and detailed sequential information about text structure.

In this paper, we propose a new STISR network, termed Multi-task learning based Text SR (MTSR) Network, which employs multi-task architecture for SR and image reconstruction to approach this problem. Image reconstruction is a close task to SR in computer vision, which generates a high-visual-quality image of the same pixels from an input image. The models of this task tend to have more robust noise-removal capability and information on the correct text structure than SR models. Therefore, we employed multi-task learning (MTL) architecture for image reconstruction and SR to take advantage of these properties. Our MTSR consists of the SR and Reconstruction branches, which simultaneously train end-to-end. The features extracted in the reconstruction branch are sent SR branch for feature representation sharing/transfer using the Feature Sharing Transformer (FST). In addition, we use a backbone in the SR branch containing the Feature Enhancement Transformer (FET) with adaptive 2D positional encoding and multi-head attention (MHA) for capturing global long-range correlations and sequential spatial information in arbitrary orientation. Features from these two transformers are fused and output as a final feature map for SR. In this work, we employ two types of 2D positional encoding for FET, such as Absolute Positional Encoding (APE) and Relative Positional Encoding (RPE), to analyze the effectiveness of these by comparison. Examples of generated SR images using MTSR and the predicted results are shown in Figure 1. The main contributions of our work are as follows:

- We propose the MTSR network that simultaneously performs text image reconstruction and super-resolution using two transformers to recover text shapes.
- The FST transfers the complementary features extracted from the reconstruction branch as an attention map to the SR branch for obtaining robust noise-removal capability and information on the correct text structure. Furthermore, the FET in the SR branch captures global long-range correlations and sequential spatial information in arbitrary orientation by adaptive 2D PE and MHA.
- Evaluation results on the TextZoom dataset [12] and popular STR benchmarks [17–20] prove the effectiveness of the proposed method. In addition, the proposed model exhibits outstanding generalization performance and learning efficiency due to the MTL of image reconstruction and super-resolution.

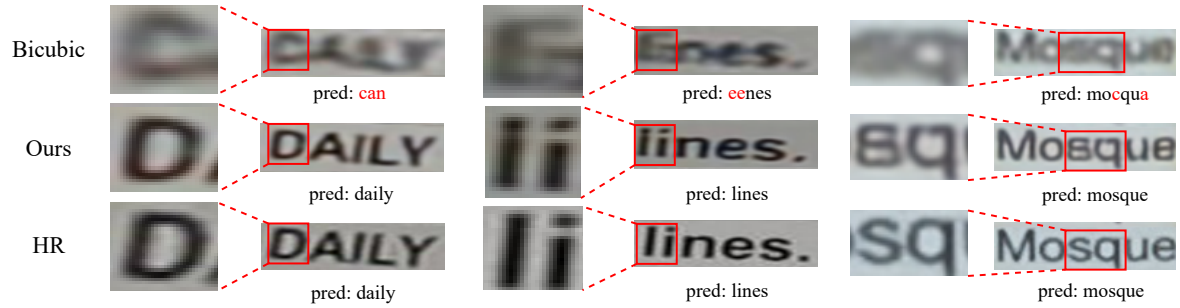


Figure 1. Examples of SR images generated using our MTSR and prediction using ASTER [21] on the TextZoom [12]. Red texts mean the results of misrecognition.

2. Related Works

2.1. Single Image Super-Resolution (SISR)

SISR aims to reconstruct fine details from degraded LR images to generate HR images. Most recent deep learning-based SISR methods use Convolutional Neural Networks (CNNs) as the backbone and achieve significantly improved performance compared to earlier work. SRCNN [3], with its simple structure using a three-layer CNN, is the leading general-purpose SISR method. Since its proposal, more complex and higher-performance SISR architectures have been developed. As other computer vision tasks have developed, SISR methods have been developed that employ Residual blocks [4], attention mechanisms [5], generative adversarial networks [6], and transformer [22]. Among the SISR methods, TTSR [22], a reference-based SR method, uses transformers for texture transfer between LR and reference images and has achieved significant performance gains.

2.2. Scene Text Recognition (STR)

Recently, scene text recognition techniques have made significant progress with the development of deep learning. Some traditional and early approaches [20,23,24] recognize text bottom-up, from character to word and word to text. However, this approach needs to be revised for Improved readability of scene text images because these methods cannot capture sequential information in the text image. CRNN [25] combines CNNs and the Recurrent Neural Network (RNN) to capture semantic and sequential information from the whole text image. In ASTER [21], the input images are rectified using Spatial Transformer Network (STN) [26], and then the feature maps are enhanced using the attention mechanism. Furthermore, advanced methods, such as Attention-based methods in the 2D direction [27] and Transformer-based STR [28], have significantly improved STR benchmarks in recent years. However, although recent significant improvements in performance, it is still challenging to recognize LR text images.

2.3. Scene Text Image Super-Resolution (STISR)

STISR is an SR task specific to scene text images. Its main difference from generic SR for natural scene images is that it aims not only to improve resolution and visual quality but also to recover the shape of the text and reconstruct the LR text images to the computer and human-recognizable images. Most early approaches [10,11] to STISR have extended generic SR methods for text images. For example, Dong et al. employed SRCNN [3] as a backbone network for text SR and achieved state-of-the-art performance in the ICDAR 2015 benchmark. Another example is the approach in [11], which uses a Laplacian pyramid network as a backbone network to capture text details by fusing multiple features from the middle layer.

These early approaches used LR images downsampled from HR images using interpolation methods such as BICUBIC and Bilinear for the training dataset. On the other hand, TSRN [12] uses the dataset TextZoom, which contains pairs of LR and HR text images of real-world scenes, for training, resulting in a significant improvement in performance on real-world scenes. In addition, TSRN utilizes a backbone network with BLSTM [16] to

attempt to capture sequential horizontal and vertical information. Since TextZoom [12] was proposed, it has become the standard for STISR training datasets and has given rise to many follow-up studies. TSRGAN [14] is a GAN-based STISR network that improves the representativeness of feature maps by introducing triplet attention. Chen et al. [15] proposed a training method that uses transformer-based networks and text-specific loss functions to improve STISR performance. TPGSR [13] pre-recognizes the semantic information of text and utilizes it in STISR to recover semantically correct text images with convincing visual quality.

3. Methodology

3.1. Overall Architecture

In this paper, we propose the MTSR network for STISR that MTL architecture of SR and reconstruction. The overall architecture of our MTSR is shown in Figure 2. The input image is $h \times w \times 4$ (h and w are the height and width) shape concatenated with an RGB image and a binary mask following [12]. MTSR consists of an image reconstruction branch and an SR branch in parallel, each processing LR text images rectified by the Spatial Transformer Network (STN) [26]. First, a shallow feature map is extracted by 1×1 convolution in each branch, and then features are extracted by SRBlock and Reconstruction Block (RecBlock) based on EDSR [29] consisting of CNNs and BLSTM [16], shown in Figure 3. More details of SRBlock are introduced in Appendix A. The extracted features from the reconstruction branch are sent as a key and value to the FST module for sharing/transferring the complementary characteristics of the reconstruction branch to the SR branch. On the other hand, the features extracted in the SR branch are sent to two modules, the FET, which enhances the feature using MHA and adaptive 2D positional encoding, and FST. The feature maps generated from the two transformer modules are fused and output as the final feature map. The model is trained to minimize the loss function between the images generated from each branch and the target images. Here, the target image for the SR branch is a pre-prepared HR image in TextZoom [12], and the target image for the reconstruction branch is a high-quality small image downsampled from the HR image.

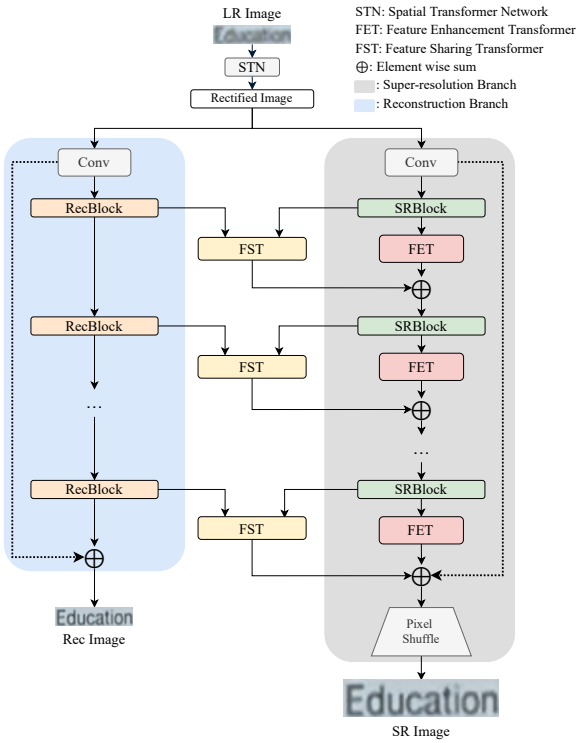
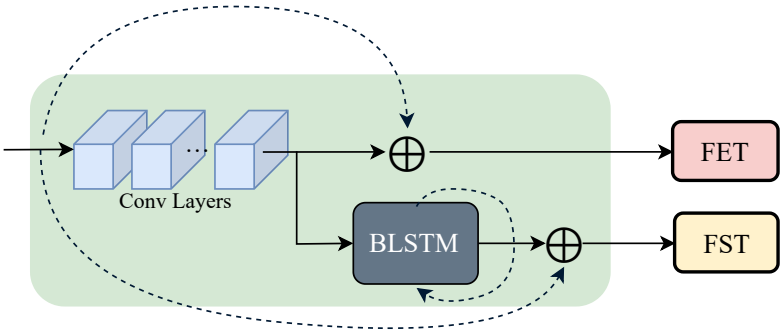
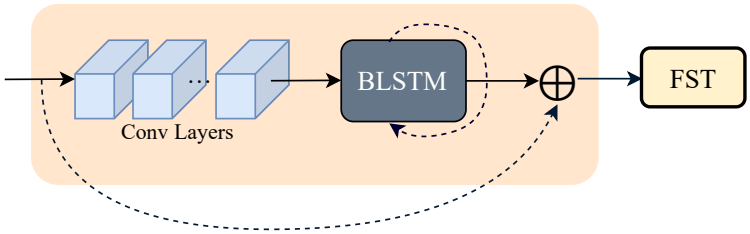


Figure 2. The overall architecture of our proposed MTSR Network for STISR. FET and FSR stand for Feature Enhancement Transformer and Feature Sharing Transformer, respectively.



(a) Super-resolution Block (SRBlock)



(b) Reconstruction Block (RecBlock)

Figure 3. SRBlocks and RecBlocks are feature extractors for the Reconstruction and SR branches, respectively. The Convolutional layers in the SRBlock and RecBlocks are both based on EDSR [29].

3.2. Feature Sharing Transformer (FST) for MTL

The reconstruction branch aims to remove noise from LR text images and generate visually high-quality images without up-sampling. The image reconstruction task is very close to SR in computer vision. However, it tends to be superior in removing noise and capturing structural features of objects because they clean the image without up-sampling. To exploit these characteristics for text reconstruction, we adopt the MTL architecture of SR and image reconstruction, and the features extracted reconstruction branch are shared/transmitted through the FST module to the SR branch. The architecture of the FST module is inspired by the transformer [22,30], which shares/transfers features between different inputs or tasks. The overall diagram of the FST module is shown in Figure 4. Here, feature maps extracted from RecBlock are treated as key (K) and value (V), and from SRBlock as query (Q).

First, feature maps extracted from RecBlock as key (K) and value (V) and from SRBlock as query (Q) are sent to the FST module, respectively. Then, for each patch q_i and k_j of Q and K, the relevance $r_{i,j}$ between these two patches is calculated with the normalized inner product as shown in Equation (1):

$$r_{i,j} = \left\langle \frac{q_i}{||q_i||}, \frac{k_j}{||k_j||} \right\rangle. \tag{1}$$

Next, Feature Transfer Attention is calculated using this $r_{i,j}$ to transfer the structural features of the text image. In order to transfer the feature with the most relevant position in V for each query q_i, we generate an index map F that represents the most relevant position of Q and K. The definition is shown in the following Equation (2):

$$f_i = \arg\max_j(r_{i,j}) \qquad i \in [1, \frac{h}{2} \times \frac{w}{2}], \tag{2}$$

where h and w are the height and width of the input, and f_i is the index representing the i-th element of the feature map from the SRBlock and the most relevant position of the feature map from the RecBlock. Then, the feature T transferred from the feature map of the RecBlock is obtained using f_i as the index and applying the index selection operation to V. Here, the Feature Transfer Attention map T contains the structural features of the text image and is denoised.

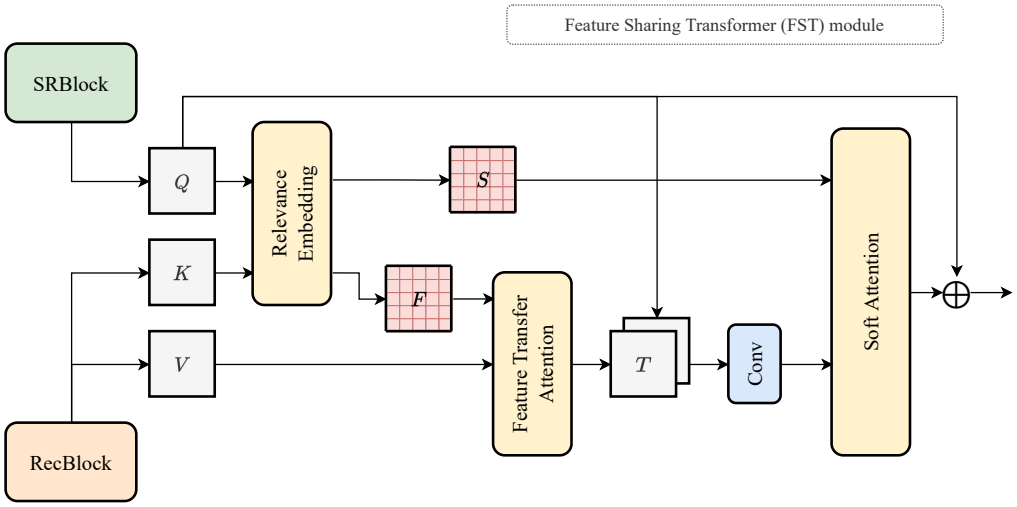


Figure 4. The architecture of the FST module. Feature maps extracted from the reconstruction branch are fused with the feature maps from the SR branch to utilize complementary characteristics, such as noise removal capability and text structure information.

Finally, Soft Attention is calculated to synthesize the features from Q and Feature Transfer Attention map T. In order to transfer relevant features, a soft attention map S is calculated from $r_{i,j}$ to estimate the confidence of the transferred texture features for each position of T as shown in Equation (3):

$$s_i = \max_j(r_{i,j}), \tag{3}$$

where s_i is the i-th position of the Soft Attention map S. Then, the feature map concatenating T and Q and the Soft Attention map S are multiplied element by element and output as the final feature map F_{out}. The definition is shown in the following Equation (4):

$$F_{out} = Q \oplus Conv(Concat(Q, T)) \otimes S. \tag{4}$$

Here, $\otimes$ is the element-wise multiplication, and $\oplus$ is the element-wise summation.

3.3. Feature Enhancement Transformer (FET)

FET is a transformer designed to capture sequential information for arbitrary directions in text images tilted or curved using 2D positional encoding. BLSTM [16], which has been used to capture sequential information in our previous work [31] and TSRN [12], is effective for horizontal and vertical sequential information but is insufficient for text images of real scenes with various shapes and orientations. Therefore, inspired by the results of studies [15,27,28] on the text image using 2D attention and recent achievements of the transformer in computer vision [32–34], we attempt to use a transformer for a feature enhancement to capture sequential information in arbitrary orientations.

The attention mechanism in typical transformers used for natural language processing and time series data processing cannot recognize spatial location information or context because the input is processed in parallel in one dimension. Thus, positional encoding for 2D orientation considers spatial positional information to enable the transformer for text

images. Furthermore, we propose two transformers, one using 2D APE and the other using 2D RPE, and compare them in an ablation study.

3.3.1. 2D Absolute Positional Encoding (APE)

Unlike sequence data, text in images appears in various orders. In particular, text in real scenes often appears not only horizontally and vertically but also tilted or curved. Therefore, based on the feature maps, positional encoding is adaptively determined for the vertical and horizontal directions. For an index p to the vertical and horizontal position of the input feature map, its positional encoding is defined as in Equations (5) and (6):

$$P_{p,2i} = sin(p/10000^{2i/D}), \tag{5}$$

$$P_{p,2i+1} = cos(p/10000^{2i/D}), \tag{6}$$

where $2i$ is the $2i$-th element in each axis p of the positional encoding and D is the number of dimensions in the depth direction. This is calculated for each of the vertical and horizontal directions. These positional encodings and feature maps are concatenated and flattened into a one-dimensional sequence, which is then input to the attention layer. Thus, the input X in FET using APE is defined as in Equation (7):

$$X = Concat(X, P_h, P_w), \tag{7}$$

where P_h is the vertical positional coding and P_w is the horizontal positional coding. The architecture of the FET with 2D APE is shown in Figure 5.

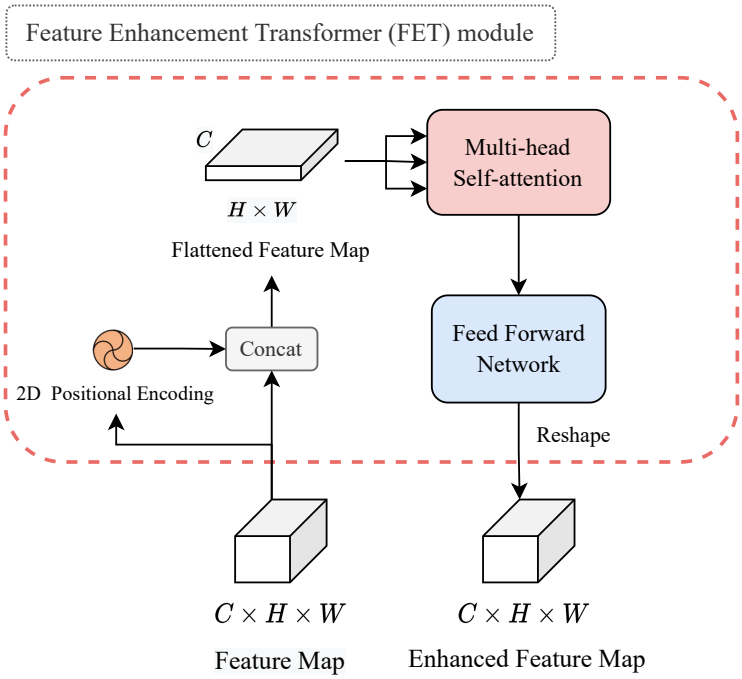


Figure 5. Illustration of FET module with self-attention modules using 2D absolute positional encoding.

In FET with 2D APE, for n elements of input $x = (x_1, \ldots, x_n)$, the output $z = (z_1, \ldots, z_n)$ of self-attention is computed as a weighted sum of the input elements, as in the general transformer, as shown in Equation (8):

$$z_i = \sum_{j=1}^{n} \alpha_{ij}(x_j W^V), \tag{8}$$

where the projection W^V is a parameter matrix and α_{ij} is each weight coefficient. α_{ij} is calculated using softmax and defined by Equation (9):

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{n} exp(e_{ik})}, \tag{9}$$

where e_{ij} is the value calculated using scaled dot-product attention. It is defined in Equation (10):

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T}{\sqrt{d_z}}. \tag{10}$$

Here, the projections W^Q and W^K are parameter matrices and are unique for each layer. d_z is the number of dimensions of z. In MHA, self-attention is calculated multiple times in parallel and then concatenated to produce the output. The output attention map is then sent to the Feed-Forward Network and finally reshaped to the same size as the input feature map.

3.3.2. 2D Relative Positional Encoding (RPE)

In recent studies [33,34], approaches that consider the relative position between each input element in the transformer have been proposed. Information such as the relative sequence and distance between elements is essential for tasks involving images. Following the RPE designed for images [33], we use a contextual RPE for text images in our FET to validate the effectiveness of the RPE. In RPE, encoding vectors are embedded in a self-attention module, and the relative positions between elements are trained during training for the transformer. The positional encoding vector p_{ij} in 2D RPE is defined for Equation (11):

$$p_{ij} = (x_i W^K) {rw_{ij}}^T, \tag{11}$$

where rw_{ij} is a trainable vector that denotes the relative positional weights between each position i and j of the input and interacts with the query embedding. In order to incorporate the encoding vector into the self-attention module, Equations (8) and (10) are reformulated as in Equations (12) and (13).

$$z_i = \sum_{j=1}^{n} \alpha_{ij}(x_j W^V + rw_{ij}^V). \tag{12}$$

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T + p_{ij}}{\sqrt{d_z}}, \tag{13}$$

Following [33], the rw_{ij} is weighted by the Euclidean method based on the Euclidean distance between the elements and the Cross method, which considers the pixel position direction. These details are introduced in Appendix B. A self-attention module based on 2D RPE is shown in Figure 6. As in the case of using APE, the self-attention module is processed multiple times in parallel as the MHA, and then the output attention map is sent to the Feed-Forward Network.

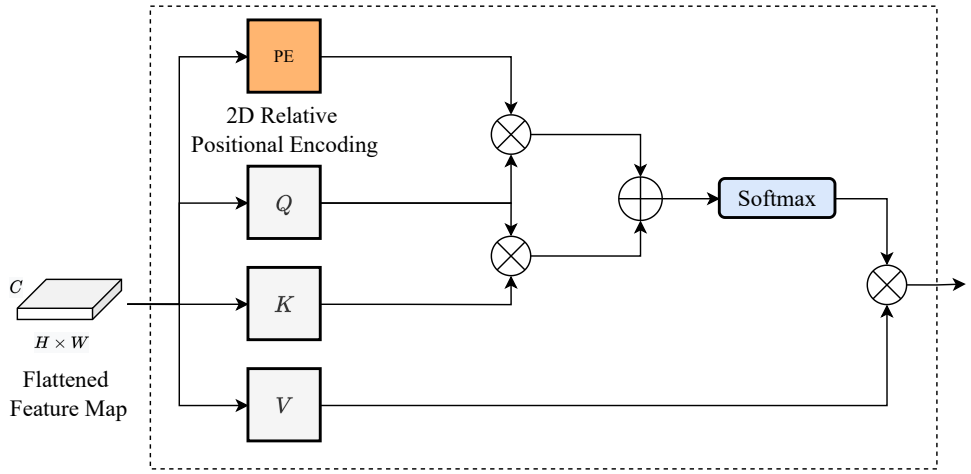


Figure 6. Illustration of self-attention modules with 2D relative position encoding.

3.4. Loss Function

In our MTSR, we attempt to optimize the model by minimizing the loss function for each SR and reconstruction branch, respectively. Following TSRN [12], we employed a combination of Gradient Profile Prior (GPP) [35] loss and L2 loss as the base loss function for each branch. The GPP loss L_{GP} is defined as in Equation (14):

$$L_{GP}(x) = \mathbb{E}_x||\nabla I_{gt}(x) - \nabla I(x)||_1 \quad (x \in [x_0, x_1]), \tag{14}$$

where I_{gt} is the ground truth image, I is the generated image, and ∇ is the gradient fields. Then, L_{sr} and L_{rec}, which are the loss functions for each branch, are defined as the sum of GPP loss and L2 loss, respectively. The final overall loss function L is defined as in Equation (15):

$$L = \lambda_{sr} L_{sr} + \lambda_{rec} L_{rec}. \tag{15}$$

Here, λ_{sr} and λ_{rec} are arbitrary values and hyperparameters. The sum of λ_{sr} and λ_{rec} is set to a maximum value of 1.0.

4. Experiment

In this section, we first analyze the effectiveness of each module, such as FSTs and FETs, the effect of hyperparameters on the loss function, and the performance of two types of positional encoding, as ablation studies. Then, the performance of MTSR on the TextZoom dataset and its ability as a preprocessor in text recognition benchmarks are evaluated by comparing it with state-of-the-art methods. In all tables, the bolded value means the maximum number.

4.1. Implementation Details

All experiments are conducted on a PC with an Intel Core i7-9700k CPU and two NVIDIA GeForce RTX2070 super GPUs. Our MTSR consists of 8 blocks and uses a 4-heads MHA for FET. The model is implemented in Python and Pytorch and trained using Adam optimizer with a learning rate of 2×10^{-4} on the TextZoom dataset. The batch size is 16, and the epoch is 300 in training. The images in the TextZoom dataset are text images culled from the SISR datasets RealSR [36] and SR-RAW [37]. These SISR datasets contain LR-HR pairs captured using digital cameras in real scenes. TextZoom consists of a training set of 17,367 images and a test set of 4373 images, with the test set divided into three subsets based on focal length: easy, medium, and hard. All input LR and target HR images are resized to 16×64 and 32×128, respectively.

The primary evaluation metric is the recognition accuracy for the generated images using the pre-trained text recognizers ASTER [21], MORAN [38], and CRNN [25]. In this process, the recognized text has all punctuation removed and uppercase letters converted to lowercase. As supplementary evaluation indices, we also use the Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index Measure (SSIM), which are general SR image quality evaluation indices.

4.2. Ablation Study

In this section, we will evaluate and analyze the effectiveness of each component for STISR, including each transformer module, the loss function, and the positional encoding. We conduct these experiments on TextZoom [12] and use the accuracy of pre-trained ASTER [21] for evaluation.

4.2.1. Ablation Study on Each Module in the Backbone

We analyze the effect on accuracy with and without each module in the backbone to prove the effectiveness of BLSTM [16], FST, and FET for text images. Here, we use 2D APE for positional encoding in the FETs. The hyperparameters λ_{sr} and λ_{rec} of the loss function are set to 0.6 and 0.4, respectively. The comparison results are shown in Table 1. "Inference FPS" means the FPS of ASTER with the proposed model as preprocessing, and "Params" means the total number of network parameters. FPS is a metric for processing speed, the number of frames per second a model processes. Our previous study [31,39] corresponds to the configurations in row 5 of the table. The results in Table 1 show that the configurations using FET and FST, respectively, are more accurate than those using only BLSTM. It can further be seen that the configuration using all three modules achieves the highest accuracy. These results indicate that FET can capture sequential information in arbitrary directions that BLSTM cannot capture. We observe that FST can also transfer the noise-reduced structural features of the reconstruction branch. On the other hand, simultaneous use of these three modules inevitably increases the number of parameters and decreases the FPS. However, the FPS of 21.4 generally maintains real-time performance and balances accuracy and processing speed.

Table 1. Ablation study on each module. The comparison results on the accuracy of ASTER with and without each module in the backbone are shown. "Inference FPS" means the number of frames the ASTER can process per second with the proposed model as preprocessing, and "Params" means the total number of network parameters.

BLSTM	FET	FST	Easy	Medium	Hard	Inference FPS	Params
✓	-	-	74.3%	56.6%	38.8%	25.3	3,072,972
-	✓	-	73.8%	54.6%	39.8%	24.3	3,339,856
-	-	✓	72.1%	55.6%	39.5%	23.2	3,568,116
✓	✓	-	74.2%	57.6%	40.9%	22.7	3,935,732
✓	-	✓	74.1%	57.9%	40.3%	22.1	4,075,088
-	✓	✓	73.5%	55.9%	40.8%	21.7	4,202,616
✓	✓	✓	**75.6%**	**59.8%**	**43.4%**	21.4	4,937,848

4.2.2. Ablation Study on Loss Function

To determine the best ratio of SR loss to reconstruction loss, we explore λ_{sr} and λ_{rec} from {0.0, 0.2, 0.4, 0.5, 0.6, 0.8, 1.0}. The comparison results are shown in Table 2. Here, we conduct experiments in a configuration with only the FST module to analyze which branch should be focused on in MTL. The results in the table show that settings of 0.4 to 0.6 for both λ_{sr} and λ_{rec} tend to have higher average accuracy. The average accuracy is lowest when either λ_{sr} or λ_{rec} is trained with a setting of 1.0 or 0.0, focusing on only one of the branches. These results indicate the effectiveness of balancing training in both SR and reconstruction branches.

Table 2. Ablation study on the loss function for MTL.

λ_{SR}	λ_{Rec}	Easy	Medium	Hard
0.0	1.0	70.78%	55.07%	38.05%
0.2	0.8	71.02%	55.12%	38.13%
0.4	0.6	71.15%	55.75%	39.14%
0.5	0.5	71.07%	**56.06%**	39.27%
0.6	0.4	71.44%	55.43%	**39.88%**
0.8	0.2	**72.02%**	54.50%	38.69%
1.0	0.0	71.82%	53.50%	38.89%

4.2.3. Ablation Study on Positional Encoding

In this section, we analyze the effects on the performance of two positional encoding approaches APE and RPE, for the FET module. Table 3 shows the comparison results. It can be seen that RPE can improve FPS with reduced parameters, while APE can improve accuracy. However, the effect is subtle. The results do not prove the significant effectiveness of RPE based on the Euclidean and Cross methods set up in this study.

Most previous studies [33,34] using RPE for computer vision tasks have indicated its effectiveness for tasks involving high-resolution images, such as object detection and image classification, and are not designed for low-resolution tasks, such as STISR. Therefore, the main reason for this experimental result may be that the model can not benefit from the relative positional information due to the tiny images handled in this task.

Table 3. Ablation study on the positional encoding approaches.

PE Method	Easy	Medium	Hard	Inference FPS	Params
APE	75.6%	**59.8%**	**43.4%**	21.4	4,937,848
RPE-Euclidean	74.3%	58.2%	41.25%	21.9	4,484,328
RPE-Cross	**75.8%**	58.6%	42.30%	22.2	4,370,253

4.3. Comparison with State-of-the-Art Methods

4.3.1. Results on TextZoom

To prove the effectiveness of our MTSR, we compared it on TextZoom with seven other SR methods, including SRCNN [3], SRResNet [4], EDSR [29], LapSRN [40], TSRN [12], TSRGAN [14], and TBSRN [15]. Here, TSRN [12], TSRGAN [14], and TBSRN [15] are recent SR models specific to text images. As the configuration in this experiment, the positional encoding for FET, λ_{sr}, and λ_{rec} are set to APE, 0.6 and 0.4, respectively. The results are shown in Table 4. It can be seen that our model achieves competitive performance with the current STISR methods, with some partially outperforming results. Especially at the medium and hard levels, the improvement in accuracy is comparatively significant, indicating that MTSR is effective for noisy and blurred images. TBSRN [15], a state-of-the-art method, uses a loss function with a pre-trained text recognition model for considering character position and recognition accuracy during learning. Benefiting from this learning method, TBSRN builds models specifically for text images and achieves high performance. On the other hand, although the loss function used to train our model is simple, it achieves comparable performance. Furthermore, our model significantly improves accuracy compared to EDSR [29], the baseline for the MTSR backbone. This result proves the effectiveness of MTL and the two transformers, and further performance improvement can be expected by specializing the learning method to text image as future work.

Comparison results of visual quality using PSNR and SSIM as complementary assessments are shown in Table 5. As with recognition accuracy, our model improves visual quality more for more challenging images such as medium and hard. However, the improvement in visual quality is less pronounced than in recognition accuracy compared to general-purpose SR methods. The reason is that PSNR and SSIM consider all image pixels,

and the recognition accuracy in Table 4 and visual quality in Table 5 are not necessarily proportional. Therefore, this is not an appropriate evaluation metric for STISR, which aims to reconstruct text regions.

Table 4. Comparison of recognition accuracy on the TextZoom dataset [12]. "Avg" indicates average accuracy in all subsets.

Method	ASTER [21]				MORAN [38]				CRNN [25]			
	Easy	Medium	Hard	Avg	Easy	Medium	Hard	Avg	Easy	Medium	Hard	Avg
BICUBIC	64.7%	42.4%	31.2%	47.2%	60.6%	37.9%	30.8%	44.1%	36.4%	21.1%	21.1%	26.8%
SRCNN [3]	69.4%	43.4%	32.2%	49.5%	63.2%	39.0%	30.2%	45.3%	38.7%	21.6%	20.9%	27.7%
SRResNet [4]	69.4%	47.7%	34.3%	51.3%	60.9%	42.9%	32.6%	46.3%	39.7%	27.6%	22.7%	30.3%
EDSR [29]	72.3%	48.6%	34.3%	53.0%	63.6%	45.4%	32.2%	48.0%	42.7%	29.3%	24.1%	32.2%
LapSRN [40]	71.5%	48.6%	35.2%	53.0%	64.6%	44.0%	32.2%	48.3%	46.1%	27.9%	23.6%	33.2%
TSRN [12]	75.1%	56.3%	40.1%	58.3%	70.1%	53.3%	37.9%	54.8%	52.5%	38.2%	31.4%	41.4%
TSRGAN [14]	**75.7%**	57.3%	40.9%	59.1%	72.0%	54.6%	39.3%	56.3%	56.2%	42.5%	32.8%	44.6%
TBSRN [15]	75.6%	59.5%	41.7%	**59.4%**	**74.1%**	57.0%	40.8%	**58.4%**	**59.6%**	**47.1%**	35.2%	**47.7%**
MTSR (Ours)	75.6%	**59.8%**	**43.4%**	58.9%	73.9%	**57.2%**	**41.8%**	56.0%	56.2%	47.0%	**35.3%**	45.4%

Table 5. Comparison of visual image quality based on PSNR and SSIM between the proposed method and representative STISR and SISR methods on the TextZoom dataset.

Method	PSNR			SSIM		
	Easy	Medium	Hard	Easy	Medium	Hard
BICUBIC	22.35	18.98	19.39	0.7884	0.6254	0.6592
SRCNN [3]	23.13	19.57	19.56	0.8152	0.6425	0.6833
SRResNet [4]	20.65	18.90	19.53	0.8176	0.6324	0.7060
EDSR [29]	24.26	18.63	19.14	0.8633	0.6440	0.7108
LapSRN [40]	24.26	18.63	19.14	0.8633	0.6440	0.7108
TSRN [12]	**25.07**	18.86	19.71	**0.8897**	0.6676	0.7302
TSRGAN [14]	24.22	19.17	**19.99**	0.8791	0.6770	0.7420
TBSRN [15]	23.82	19.17	19.68	0.8660	0.6533	**0.7490**
MTSR (Ours)	23.55	**19.88**	19.64	0.8734	**0.6843**	0.7476

To evaluate the processing speed of the proposed model, we compare the inference FPS with and without super-resolution processing as preprocessing with the three methods EDSR [29], TSRN [12], and TBSRN [15]. The results are shown in Table 6. We can see that the speed effect of adding the proposed model to ASTER [21] as preprocessing is tiny and comparable in speed to the recent STISR method. For MORAN [38] and CRNN [25], adding MTSR results in a relatively lower FPS. However, our model is appropriate as a preprocessing method considering the balance with the improvement in accuracy. On the other hand, the proposed model tends to decrease the FPS compared to other STISR and SISR methods, although it maintains practicality. This is due to the increased computational cost caused by the multi-task architecture with two network branches in parallel, which is an issue that needs to be improved for the proposed model.

Table 6. Comparison of processing speed with and without SR as preprocessing.

Method	Inference FPS				
	Without Preprocess	EDSR [29]	TSRN [12]	TBSRN [15]	MTSR (Ours)
ASTER [21]	24.3	24.2	24.2	19.4	21.4
MORAN [38]	124.4	99.8	110.9	72.8	72.2
CRNN [25]	765.1	208.6	348.7	168.0	143.1

4.3.2. Results on Scene Text Recognition Benchmarks

To evaluate the effectiveness of MTSR as a generalized STR preprocessor, we validate it using four STR benchmarks, including CUTE-80 [17], IIIT5K [18], SVTP [19], and SVT [20]. These four datasets contain text images with spatial deformations such as tilting or curving in real scenes. All 4580 images in the test set (288 from CUTE-80 [17], 3000 from IIIT5K [18], 645 from SVTP [19], and 647 from SVT [20]) in the four STR benchmarks were resized to 16×64 LR images. We further degrade the LR images by applying noise and blur processing to validate the effectiveness of our method. Specifically, Gaussian noise of $\sigma = 150$ was added to the original images for noise processing, and the original images were convolved with a 3×3 Gaussian kernel for blurring processing. Sample images are shown in Figure 7.

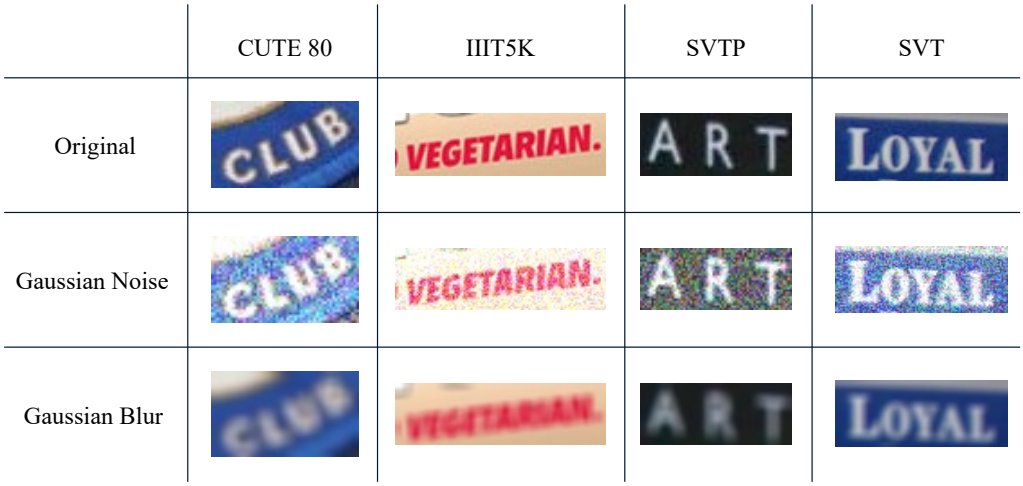


Figure 7. Sample images from the STR benchmarks processed by Gaussian blur and Gaussian noise. Gaussian noise of $\sigma = 150$ is added to the original image as noise processing, and a 3×3 Gaussian kernel is convolved with the original image as blurring processing.

In this experiment, we evaluate our model in recognition accuracy using ASTER [21] and compare our model with EDSR [29], TSRN [12], and TBSRN [15]. The results are shown in Table 7. It can be seen that both degradation processes achieve better accuracy than the compared methods on most of the datasets.

Table 7. Comparison results on scene text recognition benchmarks including CUTE-80 [17], IIIT5K [18], SVTP [19], and SVT [20]. "GN" and "GB" refer to Gaussian noise and Gaussian blurring.

Degradation	Method	Accuracy of ASTER [21]			
		CUTE-80 [17]	IIIT5k [18]	SVTP [19]	SVT [20]
GN	EDSR [29]	58.2%	76.2%	32.2%	60.1%
	TSRN [12]	60.7%	78.0%	36.9%	61.5%
	TBSRN [15]	63.5%	80.2%	**39.0%**	**64.0%**
	MTSR (Ours)	**63.9%**	**80.4%**	37.2%	62.7%
GB	EDSR [29]	59.2%	51.5%	20.1%	62.15%
	TSRN [12]	61.4%	53.8%	22.33%	63.8%
	TBSRN [15]	62.3%	54.2%	**23.1%**	65.1%
	MTSR (Ours)	**62.5%**	**54.6%**	22.2%	**65.7%**

4.4. Limitations and Failure Cases

In this section, we describe the limitations of our model found in the experiments. Figure 8 shows visualizations of our MTSR image generation failure cases in TextZoom [12]. We can see that text restoration by our MTSR is problematic in cases where characters

are indistinguishable from each other due to their proximity to each other and in cases of lengthy text. It is also difficult to recover text using our model for cases where the images have complex backgrounds and text parts are cut off in the image and have occlusions. The cases where unique fonts are used also pose difficulties to the model, and this is because the font patterns are not in the dataset and can be improved by extending the dataset.

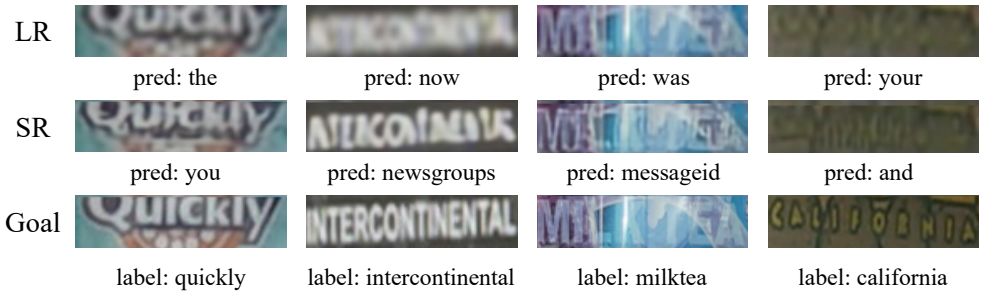


Figure 8. Examples of failures of images generated using MTSR. Several factors, such as complex backgrounds, unique fonts, and close spacing of characters, can limit text restoration using MTSR.

5. Conclusions and Discussions

In this paper, we proposed a Multi-task learning-based Text Super-resolution (MTSR) Network for scene text images. We adopt a multi-task architecture of SR and reconstruction, utilizing a transformer module called FST to recover text shape using complementary properties between the two tasks. Another transformer-based module, FET, was also used to capture sequential information for arbitrary directions by enhancing the feature map. Then, the feature maps output from each of these transformer modules was fused to improve the visual quality of text images of various shapes. Experimental results show that the proposed model achieves performance competitive with state-of-the-art methods on TextZoom. Furthermore, we proved the effectiveness of our model as a preprocessor not only for the STISR task but also for the STR benchmark.

However, our model does not achieve superior accuracy depending on the recognition method and test set, as shown in Table 4. One possible reason is that our learning methods, such as loss functions and label settings, are not sufficiently specialized for text images. It is also possible that our model does not sufficiently benefit from MTL due to the target image of the reconstruction branch being very close to the target image of the SR branch. Therefore, to build more text-specific models, our future work includes re-definition loss functions and the target image for the reconstruction branch.

Author Contributions: Methodology, K.H., M.K. and H.F.; Software, K.H.; Data curation, K.H.; Funding acquisition, H.F. and M.K.; Writing—original draft, K.H.; Writing—review and editing, M.K., H.F. and A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This study is funded by JSPS/JAPAN KAKENHI (Grants-in-Aid for Scientific Research) #JP20K11955.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SR	Super-resolution
STR	Scene Text Recognition
LR	Low Resolution
SISR	Single Image Super-Resolution
HR	High Resolution
STISR	Scene Text Image Super-Resolution

BLSTM	Bidirectional LSTM
MTSR	Multi-task learning based Text SR
MTL	Multi Task Learning
FST	Feature Sharing Transformer
MHA	Multi-Head Attention
FET	Feature Enhancement Transformer
APE	Absolute Positional Encoding
RPE	Relative Positional Encoding
CNNs	Convolutional Neural Networks
RNN	Recurrent Neural Network
STN	Spatial Transformer Network
RecBlock	Reconstruction Block

Appendix A. Architectural Variations of the SRBlock

To investigate how to send the feature maps extracted from CNNs, considering the calculation of the two transformer modules FET and FST, we present three architectures of SR Block. Three architectural variations are shown in Figure A1: (a) CNNs and feature maps via BLSTM are input to FET and FST in parallel, (b) clean CNN feature maps without BLSTM are input to FET, and (c) FST and FET in series. Comparative results are shown in Table A1. The results show that architecture (b) improves accuracy by far. This result may be because the FET is designed to capture sequential information for arbitrary directions, and the sequential information for the horizontal and vertical directions of the BLSTM is noise to the FET.

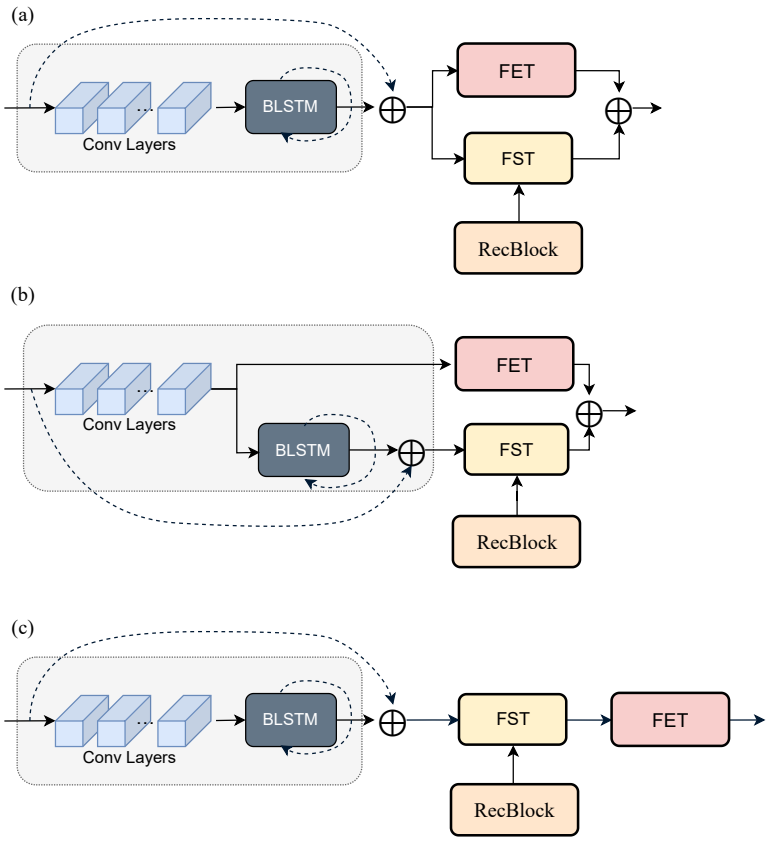


Figure A1. Three architectural of the SR Block: (**a**) CNNs and feature maps via BLSTM are input to FET and FST in parallel, (**b**) clean CNN feature maps without BLSTM are input to FET, and (**c**) FST and FET in series.

Table A1. Comparison results of the three SR Block architectures.

Architecture	Accuracy of ASTER [21]		
	Easy	**Medium**	**Hard**
(a)	72.1%	55.8%	40.2%
(b)	**75.6%**	**59.6%**	**43.4%**
(c)	73.9%	54.5%	40.9%

Appendix B. Descriptions on 2D Relative Position Calculation

This section introduces how to calculate relative positions rw_{ij} for 2D RPE in Section 3.3.2 of the main text. Both the Euclidean method and the Cross method are computed following [33].

Euclidean method. The relative positions $(\widetilde{x}_i - \widetilde{x}_j, \widetilde{y}_i - \widetilde{y}_j)$ on a 2D plane are defined as 2D coordinates, and the Euclidean distance between the two points is calculated. This distance is mapped to the corresponding encoding. This approach does not consider the orientation between elements. Equations (A1) and (A2) shows the definition:

$$rw_{ij} = P_{I(i,j)}, \tag{A1}$$

$$I(i,j) = g\Big(\sqrt{(\widetilde{x}_i - \widetilde{x}_j)^2 + (\widetilde{y}_i - \widetilde{y}_j)^2}\Big), \tag{A2}$$

where $P_{I(i,j)}$ is a learnable vector that stores the relative position weights. $I(i,j)$ is the 2D image of the target. The function $g(x)$ is a piecewise function based on [41] and indexed according to the relative position.

Cross method. This approach considers the position direction of the elements. Horizontal and vertical encodings are calculated separately, and they are fused by summarization. The method is given as Equations (A3)–(A5):

$$rw_{ij} = P^{\widetilde{x}}_{I^{\widetilde{x}}(i,j)} + P^{\widetilde{y}}_{I^{\widetilde{y}}(i,j)}, \tag{A3}$$

$$I^{\widetilde{x}}(i,j) = g(\widetilde{x}_i - \widetilde{x}_j), \tag{A4}$$

$$I^{\widetilde{y}}(i,j) = g(\widetilde{y}_i - \widetilde{y}_j), \tag{A5}$$

where $P^{\widetilde{x}}_{I^{\widetilde{x}}(i,j)}$ and $+P^{\widetilde{y}}_{I^{\widetilde{y}}(i,j)}$ are learnable vectors.

References

1. Diouf, M.; Lin, Z.; Harling, M.; Toussaint K.C. Demonstration of speckle resistance using space–time light sheets. *Sci. Rep.* **2022**, *12*, 14064. [CrossRef] [PubMed]
2. Diouf, M.; Lin, Z.; Krishna, K.; Toussaint, K.C. Multiphoton imaging using a quantitative CMOS camera. In *Proceedings of the SPIE 11965, Multiphoton Microscopy in the Biomedical Sciences XXII*; SPIE: San Francisco, CA, USA, 2022; Volume 11965, p. 119650D. [CrossRef]
3. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a Deep Convolutional Network for Image Super-Resolution. In *Computer Vision—ECCV 2014*; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2014; Volume 8692. [CrossRef]
4. Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
5. Du, X.; Liu, C.; Yang, X. Region Attention Network For Single Image Super-resolution. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021; pp. 1–6. [CrossRef]
6. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Change Loy, C. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In *Computer Vision—ECCV 2018 Workshops*; Leal-Taixé, L., Roth, S., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2018; Volume 11133. [CrossRef]

7. Qadri, T.M.; Asif, M. Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition. In Proceedings of the International Conference on Education Technology and Computer, Singapore, 17–20 April 2009; pp. 335–338. [CrossRef]
8. Yi, X.; Gao, L.; Liao, Y.; Zhang, X.; Liu, R.; Jiang, Z. CNN Based Page Object Detection in Document Images. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; pp. 230–235. [CrossRef]
9. Silva, M.S.; Jung, R.C. Realtime license plate detection and recognition using deep convolutional neural networks. *J. Vis. Commun. Image Represent.* **2020**, *71*, 102773. [CrossRef]
10. Dong, C.; Zhu, X.; Deng, Y.; Loy, C.C.; Qia, Y. Boosting optical character recognition: A super-resolution approach. *arXiv* **2015**, arXiv:1506.02211.
11. Tran, T.M.H.; Ho-Phuoc, T. Deep Laplacian Pyramid Network for Text Images Super-Resolution. In Proceedings of the 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF), Danang, Vietnam, 20–22 March 2019; pp. 1–6. [CrossRef]
12. Wang, W.; Xie, E.; Liu, X.; Wang, W.; Liang, D.; Shen, C.; Bai, X. Scene text image super-resolution in the wild. In *Computer Vision—ECCV 2020*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12355, pp. 650–666. [CrossRef]
13. Ma, J.; Guo, S.; Zhang, L. Text prior guided scene text image super-resolution. *arXiv* **2021**, arXiv:2106.15368.
14. Fang, C.; Zhu, Y.; Liao, L.; Ling, X. TSRGAN: Real-world text image super-resolution based on adversarial learning and triplet attention. *Neurocomputing* **2021**, *455*, 88–96. [CrossRef]
15. Chen, J.; Li, B.; Xue, X. Scene Text Telescope: Text-Focused Scene Image Super-Resolution. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 12021–12030. [CrossRef]
16. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [CrossRef] [PubMed]
17. Risnumawan, A.; Shivakumara, P.; Chan, S.C.; Tan, L.C. A robust arbitrary text detection system for natural scene images. *Expert Syst. Appl.* **2014**, *41*, 8027–8048. [CrossRef]
18. Mishra, A.; Alahari, K.; Jawahar, C.V. Scene text recognition using higher order language priors. In *BMVC-British Machine Vision Conference*; BMVA Press: Dundee, UK, 2012; pp. 127.1–127.11. [CrossRef]
19. Phan, Q.T.; Shivakumara, P.; Tian, S.; Tan, L.C. Recognizing Text with Perspective Distortion in Natural Scenes. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 2–8 December 2013; pp. 569–576. [CrossRef]
20. Wang, K.; Babenko, B.; Belongie, S. End-to-end scene text recognition. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 1457–1464. [CrossRef]
21. Shi, B.; Yang, M.; Wang, X.; Lyu, P.; Yao, C.; Bai, X. ASTER: An Attentional Scene Text Recognizer with Flexible Rectification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 2035–2048. [CrossRef] [PubMed]
22. Yang, F.; Yang, H.; Fu, J.; Lu, H.; Guo, B. Learning texture transformer network for image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5791–5800.
23. He, P.; Huang, W.; Qiao, Y.; Loy, C.C.; Tang, X. Reading scene text in deep convolutional sequences. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; AAAI Press: Phoenix, AZ, USA, 2016; pp. 3501–3508.
24. Jaderberg, M.; Vedaldi, A.; Zisserman, A. Deep Features for Text Spotting. In *Computer Vision—ECCV 2014*; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2014; Volume 8692. [CrossRef]
25. Shi, B.; Bai, X.; Yao, C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2298–2304. [CrossRef] [PubMed]
26. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial transformer networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 2017–2025.
27. Liao, M.; Zhang, J.; Wan, Z.; Xie, F.; Liang, J.; Lyu, P.; Yao, C.; Bai, X. Scene text recognition from two-dimensional perspective. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 8714–8721. [CrossRef]
28. Lee, J.; Park, S.; Baek, J.; Oh, J.S.; Kim, S.; Lee, H. On Recognizing Texts of Arbitrary Shapes with 2D Self-Attention. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 2326–2335. [CrossRef]
29. Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K.M. Enhanced deep residual networks for single image super-resolution. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 1132–1140.
30. Feng, C.M.; Yan, Y.; Fu, H.; Chen, L.; Xu, Y. Task Transformer Network for Joint MRI Reconstruction and Super-Resolution. In *Medical Image Computing and Computer Assisted Intervention—MICCAI 2021*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2021; Volume 12906. [CrossRef]

31. Honda, K.; Fujita, H.; Kurematsu, M. Improvement of Text Image Super-Resolution Benefiting Multi-task Learning. In *Advances and Trends in Artificial Intelligence. Theory and Practices in Artificial Intelligence*; Fujita, H., Fournier-Viger, P., Ali, M., Wang, Y., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2022; Volume 13343. [CrossRef]
32. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 $\times$ 16 words: Transformers for image recognition at scale. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 3–7 May 2021. Available online: https://openreview.net/forum?id=YicbFdNTTy (accessed on 8 November 2022).
33. Wu, K.; Peng, H.; Chen, M.; Fu, J.; Chao, H. Rethinking and Improving Relative Position Encoding for Vision Transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 10033–10041.
34. Srinivas, A.; Lin, T.Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck Transformers for Visual Recognition. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 16514–16524. [CrossRef]
35. Sun, J.; Xu, Z.; Shum, H.Y. Gradient Profile Prior and Its Applications in Image Super-Resolution and Enhancement. *IEEE Trans. Image Process.* **2011**, *20*, 1529–1542. [CrossRef] [PubMed]
36. Cai, J.; Zeng, H.; Yong, H.; Cao, Z.; Zhang, L. Toward real-world single image super-resolution: A new benchmark and a new mode. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Long Beach, CA, USA, 15–20 June 2019; pp. 3086–3095.
37. Zhang, X.; Chen, Q.; Ng, R.; Koltun, V. Zoom to learn, learn to zoom. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3762–3770.
38. Luo, C.; Jin, L.; Sun, Z. Moran: A multi-object rectified attention network for scene text recognition. *Pattern Recognit.* **2019**, *90*, 109–118. [CrossRef]
39. Honda, K.; Fujita, H.; Kurematsu, M. Multi-Task Learning-Based Attentional Feature Fusion Network for Scene Text Image Super-Resolution. *New Trends Intell. Softw. Methodol. Tools Tech.* **2022**, *355*, 334–345. [CrossRef]
40. Lai, W.S.; Huang, J.B.; Ahuja, N.; Yang, M.H. Deep Laplacian Pyramid Networks for Fast and Accurate Super-Resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5835–5843. [CrossRef]
41. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, J.P. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67. Available online: http://jmlr.org/papers/v21/20-074.html (accessed on 8 November 2022).

electronics

MDPI

Article

Research on Retinex Algorithm Combining with Attention Mechanism for Image Enhancement

Mingzhu Liu *, Junyu Chen and Xiaofei Han

The Higher Educational Key Laboratory for Measuring & Control Technology and Instrumentation of Heilongjiang Province, Harbin University of Science and Technology, Harbin 150080, China
* Correspondence: lmz@hrbust.edu.cn

Abstract: Considering the high noise and chromatic aberration in the Retinex-Net image enhancement results, this paper put forward a modified Retinex-Net algorithm for weak illumination image enhancement based on the Decom-Net and Enhance-Net structures of Retinex-Net. The improved structure proposed in this paper adds the attention mechanism ECA-Net into the Decom-Net and Enhance-Net convolution layer of the original Retinex-Net structure, which can effectively reduce the problem of irrelevant background and local brightness imbalance, activate sensitive features, and improve the image's details and brightness processing ability. Additionally, deep connected attention networks are embedded between the introduced attention modules, so that all of the attention modules can be trained jointly to improve the learning ability. Furthermore, the improved method also introduces a noise reduction loss function and a color loss function to suppress noise and to reduce image color distortion. The test results of the proposed method indicate that the image's overall brightness can be balanced, the local areas cannot be overexposed, and more image details and color information can be retained than with other enhancement algorithms.

Keywords: image enhancement; Retinex algorithm; ECA net module; connected attention; low brightness background

1. Introduction

Weak illumination image enhancement technology has recently become an important research topic. When the shooting environment is dark, the image's brightness and contrast will be affected, and the color will be distorted, which will affect intuitive visual effects and subsequent detections of image enhancement [1–3]. To enhance the low-light image, in that early stage, people performed a lot of research on histogram equalization [4,5]. The method in literature [6] is inspired by the dark channel prior defogging theory [7], and it is found that the visual effect and histogram of low illumination images after inversion are highly similar to those of fogged images. Based on this observation, literature [6] applied the defogging algorithm to the inverted weak illumination image and flipped it again to obtain the enhanced weak illumination image. Land put forward the theory of the retina cortex [8–10] (Retinal Cortex Theory, Retinex). With the development of research, machine learning methods can be applied. The combination of Retinex theory and CNNs further improves the effect of image enhancement [11–13]. However, after the Decom-Net extracts the reflection parts, the noise level will be affected by the light intensity, which makes the dark area's noise too high, causing the noise and color deviation of the low illumination enhanced result of Retinex-Net. Therefore, this paper put forward a modified algorithm based on Retinex-Net to ameliorate the effect of the enhanced image.

2. Retinex Image Enhancement Algorithm

At first, the gray value of the low illumination image's dark area is changed by histogram equalization and gamma correction to increase the image brightness [14]. However,

these methods only consider the image's global information, ignoring the details. The Retinex theory is based on color consistency, which can decompose the color image into a reflection part and an illumination part. As shown in Formula (1):

$$S = R \circ I \tag{1}$$

where $\circ$ denotes a pixel-by-pixel multiplication operation. S denotes a color image, in which the illumination of each region is different. R denotes a reflection part, which is an inherent physical property of an object, and the true color of the object is independent of the luminance. I denotes the illumination part, which reflects the degree of exposure of the photographic object.

2.1. Surround Retinex Method

The single-scale Retinex (SSR) uses the convolution of the surround function and the input image to obtain the illumination part. The SSR algorithm expression is presented in Formula (2):

$$R_i(x,y) = \lg \frac{I_i(x,y)}{L_i(x,y)} = \lg I_i(x,y) - \lg[F(x,y) * I_i(x,y)] \tag{2}$$

$$F(x,y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{(x^2+y^2)}{\sigma^2}} \tag{3}$$

where i denotes different RGB channels, $I_i(x,y)$ denotes the input image, and $L_i(x,y)$ denotes the incident part, defined by the convolution yields of $F(x,y)$ and $I_i(x,y)$. $R_i(x,y)$ denotes the reflection part. $F(x,y)$ denotes the wrap function, as shown in Equation (3), where σ is the scale parameter; the size of σ determines the effect of image enhancement [15].

The multiscale Retinex (MSR) is a weighted sum of SSR outputs at multiple scales [16]. The MSR algorithm expression is presented in Formula (4):

$$R_{MSR_i}(x,y) = \sum_{1}^{N} \omega_n \cdot R_{ni}(x,y) \tag{4}$$

where ω_n is the weighted value, and N is the number of scales. $R_{ni}(x,y)$ denotes the reflection image at the n-th scale. The surrounding function expression of $R_{ni}(x,y)$ is shown in Formula (5):

$$F_n(x,y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{(x^2+y^2)}{\sigma^2}} \tag{5}$$

The MSRCR algorithm adds a color restoration factor C_i used to improve the color deviation problem caused by the MSR algorithm [17], and the MSRCR expression is shown in Formula (6):

$$R_{MSRCRi}(x,y) = C_i(x,y) R_{MSR_i}(x,y) \tag{6}$$

where $C_i(x,y)$ denotes the color recovery factor of the i-th channel in R, G, and B, and its expression is shown in Equation (7):

$$C_i(x,y) = \beta \ln \left[\alpha \cdot \frac{I_i(x,y)}{\sum_{i=1}^{S} I_i(x,y)} \right] \tag{7}$$

where β denotes the gain coefficient and α denotes the nonlinear controlled intensity factor. According to [18], $\beta = 46$ and $\alpha = 125$. S presents the spectral channels, $S = 1$ presents a grayscale image, and $S = 3$ presents an RGB color image.

2.2. The Retinex-Net Model

Most of the existing image enhancement methods based on Retinex rely on the careful design of constraints and parameters by hand. Therefore, when these methods are applied in different scenarios, they may be trapped by the model's capacity. Retinex-Net can

automatically learn the input weak illumination image's features, which solves the problem that the traditional algorithm relies on manually setting parameters. The Retinex-Net model structure is presented in Figure 1. It consists of three parts: the decomposition model, adjustment model, and reconstruction. Namely, Decom-Net and Enhance-Net form the primary network of the Retinex-Net.

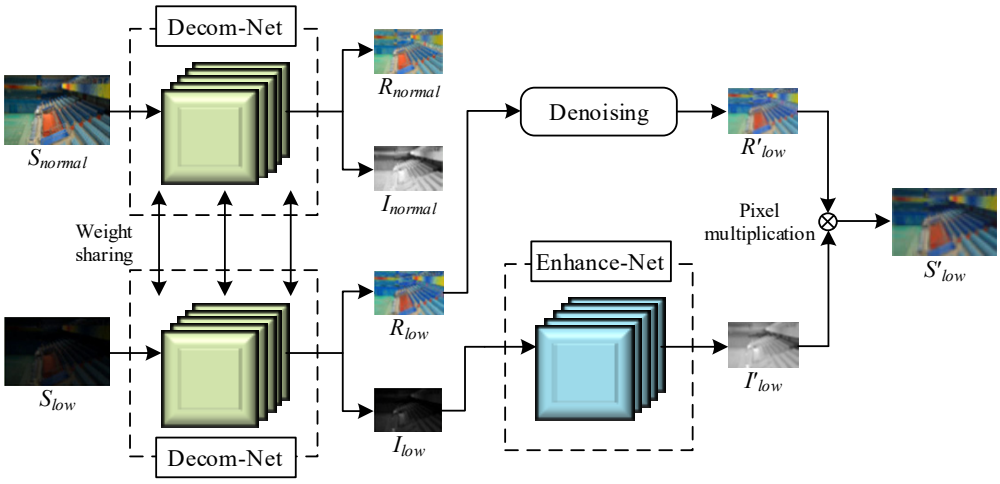


Figure 1. Retinex-Net network structure diagram.

The Decom-Net uses a 3×3 convolution layer to generate the characteristic of the input image, and uses the linear rectification function ReLU as the activation function. After five 3×3 convolution layers with ReLU, the RGB image is divided into reflection and illumination images. The low-light image S_{low} and the matched normal-light image S_{normal} are respectively input into the Decom-Net model, and the network parameters are shared by the two images, decomposed into the corresponding reflection parts (R_{low}, R_{normal}) and illumination parts (I_{low}, I_{normal}). Comparing the output images of the decomposition model in Figure 1 shows that the reflection parts R_{low} and R_{normal} under normal illumination and low illumination are relatively similar. In contrast, the large difference in the illumination part between I_{low} and I_{normal} indicates that the reflection part is the inherent attribute of the image, which is not easily affected by the original shooting illumination intensity. The illumination part determines the visual perception of the human eye when observing the image.

In the denoising process of the reflection image, the three-dimensional block matching algorithm (BM3D) [19] is mainly used to reduce the noise of the reflection part to obtain a denoised reflection image, R'_{low}. The main idea of the BM3D is to find a reference block with a step of s in the image, search in the vicinity of the reference block, and find some blocks with small differences from the reference block. Then, these blocks will be integrated into a three-dimensional matrix for filtering, and finally, the results are fused into a two-dimensional image using inverse transformation to obtain a denoised image.

The Enhance-Net adopts the encoder–decoder structure and inputs the illumination part I_{low} obtained by Decom-Net. The feature map with a smaller size is obtained through multiple down-sampling layers so that the light intensity is redistributed with larger scale angles in the Enhance-Net, and it can adjust the brightness. The Enhance-Net reconstructs the light intensity of the image through up-sampling, allocates lower brightness to a locally brighter place, and allocates higher brightness to a locally darker place. Furthermore, the output of the up-sampling layer is cascaded by the number of channels so that different local illuminations are adjusted, and the global illuminations of the image are kept consistent. At the same time, the element-by-element summation is used to guide skip connections from one down-sampling layer to its corresponding mirrored up-sampling layer so that the network learns the residual. The low illumination part I_{low} is subjected to the Enhance-Net to generate an enhanced illumination part, I'_{low}. Finally, the enhanced illumination part

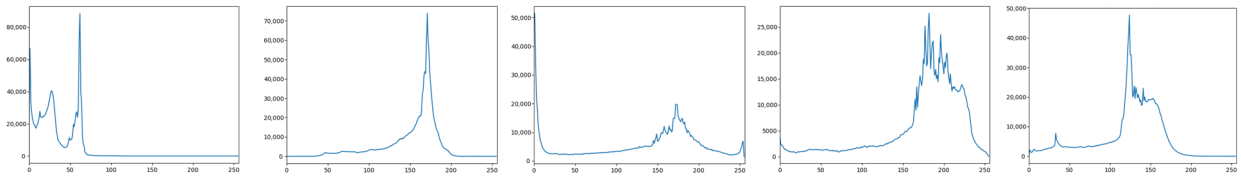
I'_{low} is recombined by multiplying the two images' pixels and the denoised reflection part R'_{low}, forming the enhanced image S'_{low} as the network output.

2.3. Image Enhancement Test and Analysis

In this paper, the images under low illumination are selected, and the enhancement effects of the SSR, MSR, MSRCR, and Retinex-Net algorithms are evaluated. Figure 2a is a dark image to be processed with low brightness and contrast. Figure 2b–e are effect pictures based on four enhancement algorithms of SSR, MSR, MSRCR, and Retinex-Net. Figure 3a–e are gray histograms of the images.

(**a**) Original image. (**b**) SSR. (**c**) MSR. (**d**) MSRCR. (**e**) Retinex-Net.

Figure 2. Renderings of different enhancement algorithms.



(**a**) Original image. (**b**) SSR. (**c**) MSR. (**d**) MSRCR. (**e**) Retinex-Net.

Figure 3. Gray histograms corresponding to different enhancement algorithms.

From the gray histograms of Figure 3a–e, it is found that the basic structure of the enhanced image is quite different from the original image, indicating that the image's brightness is significantly improved. Combined with the enhancement effect and the gray histogram, it is shown that SSR only retains the low-frequency part, which improves the overall brightness while losing the important high-frequency part of the input image. MSR is less sensitive to the highlighted area and loses the image's color characteristics. Although MSRCR recovers the color, the color retention ability is not good enough, and the image is overexposed. Although Retinex-Net has the fastest processing speed and can obtain a better enhancement effect without manually setting parameters, it has the phenomena of large noise and color distortion. Therefore, this paper introduces the attention mechanism and adds the denoising loss and the color loss to ameliorate the traditional Retinex-Net, aiming at the shortcomings of Retinex Net's inability to retain image texture details and color features.

3. The Improved Retinex-Net Model

In the convolution process, each channel in the default feature map has the same importance, while facing the actual problem, and the importances of different channels are different. The attention mechanism can reallocate the input weight during convolution. The introduction of the attention mechanism in Retinex-Net can reduce the attention of the network to the irrelevant features of the image and only focus on the crucial parts of the image, such as color, texture, and shape, to improve the processing ability of image brightness and details.

3.1. Introduction of Attention Mechanism and Its Deep Connection

ECA-Net (Efficient Channel Attention Network) is an efficient channel attention mechanism [20], which avoids feature dimension reduction in the previous attention mechanism and enables the information between channels to interact effectively. After

channel-level global average pooling (GAP) without dimension reduction, it uses the one-dimensional sparse convolution operation to capture the current channel and its k domain channel information. The interaction of domain channel information dramatically reduces the number of parameters and improves the performance. The ECA module structure is shown in Figure 4 below. Embedding the ECA module in the Decom-Net can learn the importance of each feature channel adaptively. Each noise point is normalized and weighted during the denoising process, which automatically removes noise points, improves the clarity of decomposed image, and reduces the loss of key features.

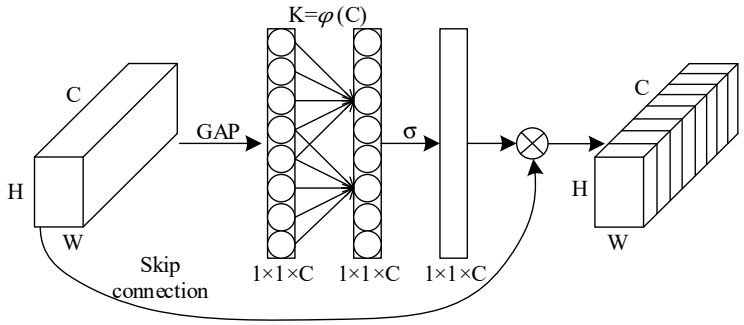


Figure 4. ECA-Net module structure diagram.

Assume that input from the previous convolution layer is a $H \times W \times C$ feature map of C channels in total, and then map each two-dimensional feature channel to a real Z using Formula (8) through global average pooling. Output a $1 \times 1 \times C$ global description feature, and the mapping relationship is shown in Formula (8):

$$Z = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_C(i, j) \tag{8}$$

where $X_C(i, j)$ denotes the C-th two-dimensional matrix in the input feature map, which indicates the numerical distribution of the C feature maps in this layer, and Z denotes the obtained global information. Then, a one-dimensional convolution with a convolution kernel size of k captures the local cross-channel interaction information, obtains the normalized weights through the Sigmoid activation function, and acts on the previous feature map. To make the model more lightweight and compact, the ECA model simplifies the dense connection mode in the SE module of the previous attention mechanism, avoids complex channel dependence or combines additional spatial attention, and only considers the information interaction in adjacent channels. The weight calculation formula is as shown in Formula (9):

$$\omega_i = \sigma(\sum_{j=1}^{k} \omega_i^j y_i^j), y_i^j \in \Omega_i^k \tag{9}$$

where σ denotes the Sigmoid function and y_i denotes the channel. ω_i is the weight of the channel y_i, ω_i^k is the set of the k-th neighboring channel of y_i, and k is the convolution kernel size, which denotes the coverage of local cross-channel interaction, which signifies how many adjacent channels close to the channel participate in the attention prediction of the channel. Meanwhile, to avoid manually tuning the value of k, the ECA module will adaptively determine k. One-dimensional convolution with the kernel size of k can realize the effective channel attention module, as shown in Formula (10):

$$\omega = \sigma(C1D_k(y)) \tag{10}$$

where $C1D_k$ denotes the one-dimensional convolution operation with convolution kernel k, and y denotes the channel. The coverage of cross-channel information interactions (i.e., the kernel size k of the one-dimensional convolution) is proportional to the channel dimension

C. There is a mapping relationship φ between k and C. Because the linear mapping relationship has limitations for some related features, the mapping relationship φ adopts a nonlinear mapping. Taking into account that the channel dimension is typically an exponential multiple of 2, the linear map relation is introduced into the nonlinear mapping relation. The expression of the mapping relationship with the channel dimension C and the convolution kernel size k is shown in Equation (11), where η and b are constants representing the linear mapping relationship, and $\eta = 2$, $b = 1$ is taken [18].

$$k = \varphi(C) = \left| \frac{\log_2(C)}{\eta} + \frac{b}{\eta} \right|_{odd} \tag{11}$$

Although the attention mechanism can play a good role in many image processing tasks, it is limited to the current features each time, and does not fully use the attention mechanism. DCA-Net (Deep Connected Attention Network) can establish skip connections between adjacent ECA attention modules so that information interacts between each attention module, and all attention modules are jointly trained [21]. This improves the ability of the attention mechanism to learn important features of images, and optimizes the performance of Retinex-Net to extract features of reflection images and restore illumination images and colors. The DCA module is plug-and-play, improving attention performance without changing the internal structure of the Retinex-Net model.

The DCA uses a general attention framework consisting of three parts: Context Extraction, Transform, and Fusion, and the network structure is shown in Figure 5. Firstly, the ECA module extracts the feature G from the input feature map X through the extractor g with a global average pooling, where $X \in R^{H \times W \times C}$, and ω_g is the relevant parameter of the extraction operation. The extractor g determines the relevant parameter. When g is a no-parameter operation, such as a pooling operation, ω_g is no longer needed. Then, the extracted feature blocks are transformed into a new nonlinear attention space T by using a one-dimensional convolution kernel and an excitation function, and the output T can be indicated as $T = t(G, \omega_t)$. t is defined as the feature transformation operation, and ω_t is the parameter used in the transformation operation. Finally, the attention map is fused with the original convolution block feature and the output is $X' = T \otimes X$. $\otimes$ denotes the feature fusion mode, and the fusion function performs element multiplication when it is designed to be scaled, and performs element addition otherwise. To sum up, the model of a general attention module can be expressed as Equation (12):

$$X' = t(g(X, \omega_g), \omega_t) \otimes X \tag{12}$$

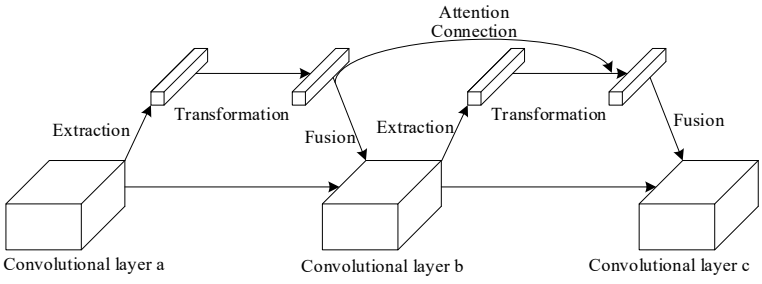


Figure 5. Deep Connected Attention (DCA) network structure.

The DCA module will introduce a connection channel between each attention block to fuse the feature output extracted by the previous attention module with the output of the current attention module. This design ensures that the present attention module can learn the extracted features and the former information. From this, the attention module can be indicated as Formula (13):

$$X' = t(f(\alpha G, \beta T), \omega_t) \otimes X \tag{13}$$

where f is a link function, a and β are learnable parameters, and $\widetilde{T}$ denotes the attention mapping feature map generated by the former attention module. Since the improvement of network performance is more from the connection among attention blocks [19] rather than the form of the connection function, it is not sensitive to the connection mode. Therefore, the connection mode between attention blocks uses direct connection by default, and the connection function f is as shown in Formula (14):

$$f(\alpha G, \beta\widetilde{T}) = \alpha G + \beta\widetilde{T} \tag{14}$$

3.2. Improvement of Subnetworks

Aiming at the problem that the original Decom-Net is not conducive to preserving image details and suppressing noise, this paper first improves and optimizes the structure of the Decom-Net in the original Retinex-Net model. The improved Decom-Net model is shown in Figure 6. First, the Decom-Net uses a 3×3 convolution layer to extract the input image S_{low}. Then, it uses five convolution layers with ReLU activation functions to change the size of the feature map and to learn the characteristic of the reflection part R_{low} and the illumination part I_{low}. The attention mechanism ECANet is added after the convolutions of the second and fourth layers, respectively, and the deep connection attention DCA is established between the two attention blocks. Finally, using the convolution layer and Sigmoid function, the learned image features are mapped into a reflection image R_{low} and an illumination image I_{low}, and then output.

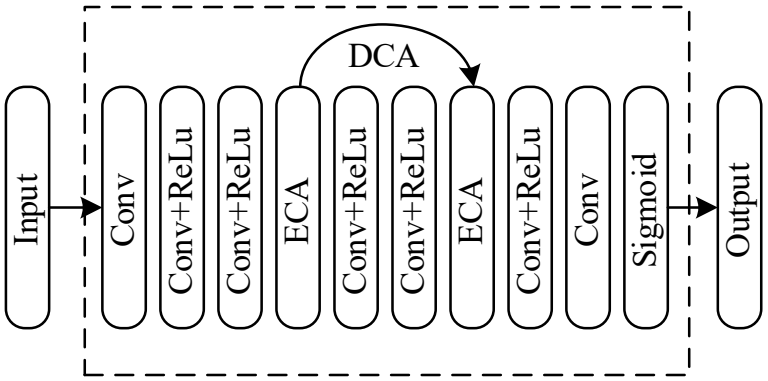


Figure 6. Improved Decom-Net network.

The original Retinex-Net has the problem of image color distortion. This paper ameliorates the structure of the Enhance-Net based on the original Retinex-Net to improve the color distortion. Firstly, the Enhance-Net uses a 3×3 convolution layer to extract the features of the low-light input obtained by Decom-Net. Then, the encoder–decoder architecture obtains the context information in a large area, and the local illumination distribution is reconstructed. The skip layer is connected between the down-sampling layer and the corresponding mirror-image up-sampling layer. The attention module ECA is added to the connection between the upper sampling layer and the jump layer to reduce the response of the network to image-independent features, so that the next upper sampling layer can carry more illumination information. Additionally, deep connections are established between the attention modules to strengthen the attention mechanism's attention to the image's brightness, and further improve the network's learning ability of brightness characteristics. Finally, these feature maps are adjusted to the final scale at different scales using the nearest neighbor interpolation algorithm. By using a 1×1 convolution layer and a 3×3 convolution layer to reduce the number of channels, the illumination image is reconstructed. The improved augmented network model is shown in Figure 7:

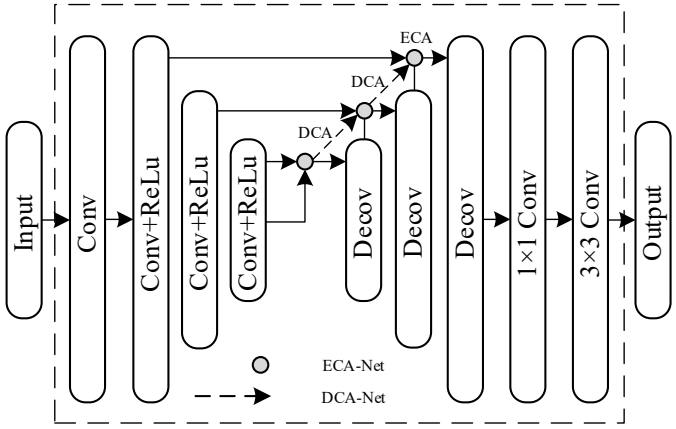


Figure 7. Improved Enhance-Net network.

3.3. Improvement of Loss Functions

As for the loss function of the Decom-Net, the proposed method retains the reconstruction loss $\mathcal{L}_{recon}$, invariable reflectance loss $\mathcal{L}_{ir}$, and illumination smoothness loss $\mathcal{L}_{is}$ of the original Decom-Net. In addition, a denoising loss $\mathcal{L}_{dn}$ is added for further denoising. Thus the total loss $\mathcal{L}_{de}$ is shown in Formula (15):

$$\mathcal{L}_{de} = \mathcal{L}_{recon} + \lambda_{ir}\mathcal{L}_{ir} + \lambda_{is}\mathcal{L}_{is} + \lambda_{dn}\mathcal{L}_{dn} \tag{15}$$

where λ_{ir}, λ_{is}, and λ_{dn} are the coefficients of each type of loss used to balance each loss part.

Based on the output result R_{low} and R_{normal} in the Decom-Net, and under the assumption that the images can be reconstructed using the respective illumination maps I_{low} and I_{normal}, the reconstruction loss is described by Equation (16):

$$\mathcal{L}_{recon} = \sum_{i=low,normal} \sum_{j=low,normal} \lambda_{ij} \| R_i \circ I_j - S_j \|_1 \tag{16}$$

where $\circ$ denotes the pixel multiplication. According to reference [10], when i takes low or j takes normal, the weight coefficient $\lambda_{ij} = 1$, otherwise, $\lambda_{ij} = 0.001$. The Decom-Net uses larger weight coefficients to better learn the paired images' characteristics with low and normal illumination.

The invariable reflectance loss constrains the network from learning the reflectivity of the low-illumination and normal-illumination images. According to the color consistency of Retinex theory, the color reflected by the object is irrelevant to the light intensity and has consistency, as shown in Formula (17):

$$\mathcal{L}_{ir} = \| R_{low} - R_{normal} \|_1 \tag{17}$$

A good light map should be smooth in detail while preserving the overall structural boundaries. Illumination smoothness loss avoids the problem that the image detail and the boundary gradient are uniformly reduced when the total variation (TV) is directly used as the loss function so that the illumination is blurred and black edges exist on the reflection map. Weighting the original TV function and the reflectance gradient, ∇R_i makes the brightness of the illumination map smoother, and image characteristics are favorably retained. The final $\mathcal{L}_{is}$ is shown in Formula (18):

$$\mathcal{L}_{is} = \sum_{i=low,normal} \| \nabla I_i \circ \exp\left(-\lambda_g \nabla R_i\right) \|_1 \tag{18}$$

where λ_g presents the balance coefficient of the strength of structure awareness. Then, $\exp\left(-\lambda_g \nabla R_i\right)$ relaxes the smooth constraint at the steeper reflection gradient, where the image structure is more complex, and the lighting is discontinuous.

When SSIM is used as the loss function, although it can better constrain the structural features of the network learning image, it is less sensitive to the smooth region of the image, and it needs a specific configuration to perform well. Although $MS-SSIM$ with multiscale structure similarity can keep high-frequency information well for images with different resolutions, it is easy to change the brightness and produce color deviation. In comparison, the $L1$ loss function can maintain brightness and color more effectively. Hence, for that, the denoising loss $\mathcal{L}_{dn}$ is optimized using a hybrid loss function of $MS-SSIM$ and $L1$ loss, where the expression of $MS-SSIM$ is as shown in Equation (19); then the expression is shown in Formula (20):

$$\mathcal{L}_{MS-SSIM}(P) = 1 - MS-SSIM(\widetilde{p}) \tag{19}$$

$$\mathcal{L}_{dn} = \alpha \cdot \mathcal{L}_{MS-SSIM} + (1-\alpha) \cdot G_{\sigma_G^M} \cdot \mathcal{L}_{L1} \tag{20}$$

where $\widetilde{p}$ is the center point pixel, α is the coefficient, $\mathcal{L}_{L1}$ is the $L1$ loss function, and α is taken as 0.84 according to [22]. $G_{\sigma_G^M}$ is the Gaussian distribution coefficient of the pixel point, and the approximate value is used instead of the pyramid structure to reduce the calculation cost [22].

When denoising loss, $\mathcal{L}_{dn}$ is $L1$, $SSIM$, $MS-SSIM$, and $MS-SSIM$ are mixed with $L1$, and the denoising evaluation results from the literature [22] are shown in Table 1, where the mixing loss is recorded as Mix.

Table 1. Different denoising loss results.

Evaluation Index	$L1$	SSIM	$MS-SSIM$	Mix
PSNR(dB)	34.42	33.15	33.29	34.61
SSIM	0.9535	0.9500	0.9536	0.9564
FSIM	0.9775	0.9764	0.9782	0.9795

As for the loss function of the Enhance-Net, the proposed method retains the reconstruction loss and illumination smoothness loss of the original Enhance-Net. In addition, color loss $\mathcal{L}_{color}$ is also added to enhance the saturation of the color and to improve the color deviation problem. Therefore, the total loss constraint $\mathcal{L}_{en}$ of the network is shown in Equation (21), where μ is the balance coefficient of color loss function $\mathcal{L}_{color}$.

$$\mathcal{L}_{en} = \mathcal{L}_{recon} + \mathcal{L}_{is} + \mu\mathcal{L}_{color} \tag{21}$$

The reconstruction loss of Enhance-Net is defined as the distance between the normal image and the corresponding enhanced illumination image, as shown in Equation (22):

$$\mathcal{L}_{recon} = \left\| R_{low} \circ I' - S_{normal} \right\|_1 \tag{22}$$

The illumination smoothness loss resembles that of the Decom-Net. The only difference is that I'_{low} in the Enhance-Net takes the gradient of R_{low} as the weight coefficient, and its expression is shown in Equation (23):

$$\mathcal{L}_{is} = \left\| \nabla I'_{low} \exp\left(-\lambda_g \nabla R_{low}\right) \right\|_1 \tag{23}$$

The color loss function $\mathcal{L}_{color}$ uses the Huber loss function to enhance the saturation of the color; i.e., $\mathcal{L}_{color} = \mathcal{L}_h$. The Huber loss is a robust estimator, and has been shown to solve the average coloring [23,24], so it helps to increase the color saturation of images in Enhance-Net. The expression of the Huber loss function is shown in Equation (24):

$$\mathcal{L}_h = \begin{cases} \frac{1}{2}\left(I_r - \widetilde{I}\right)^2 & I_r - \widetilde{I} < \delta \\ \delta\left|I_r - \widetilde{I}\right| - \frac{1}{2}\delta^2 & I_r - \widetilde{I} \geq \delta \end{cases} \tag{24}$$

where I_r is the true value, $\widetilde{I}$ is the estimated value, and δ is a hyperparameter. When δ trends to zero, the Huber loss tends to the mean absolute error (MAE). When δ trends to ∞, the Huber loss trends to the mean square error (MSE). According to the E.L Lehmann point estimation theory [25], it is more appropriate to take 1.5 as the hyperparameter δ. The color loss function $\mathcal{L}_{color}$ evaluates at each pixel, measures the image color more accurately, learns the color deviations that occur locally, and then adds them together to evaluate the total loss constraint of the image.

4. Experiment and Results Analysis

4.1. Experimental Environment

A Low-Light (LOL) data set is used to train the network. The test set selects the validation set of the LOL data set. The network reads weak light and normal light images in turn for training. The batch size is set to 16, the patch size is set to 48×48, the training epoch is set to 100, and the model is evaluated and saved every 20 training times. The initial learning rate is set to 0.001. The loss balance coefficients for the decomposed network are set to be: $\lambda_{ir} = 0.001$, $\lambda_{is} = 0.1$, and $\lambda_{dn} = 0.001$; the coefficient of denoising loss parameter $\alpha = 0.84$, and the color loss balance coefficient of the Enhance-Net are $\mu = 0.01$, $\lambda_g = 10$, respectively.

All network model training and improvement test experiments are based on the Tensorflow framework. In order to verify the performance and effect of the proposed method, a variety of methods are used for comparison, including SRIE, NPE, GLADNet, Enlighten-GAN, and Retinex-Net. Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM), natural quality image evaluator (NIQE) [26], and lightness order error (LOE) [27] were used as objective evaluation indexes. PSRN is used to measure the noise level, and SSIM is used to evaluate the comprehensive image quality from brightness, contrast, and structure. The higher the SSIM and PSNR, the closer the enhanced image is to the normal image. NIQE is used to measure the difference among enhanced images in natural image features, and LOE is used to evaluate the difference in local brightness order. The higher the SSIM and PSNR, the lower NIQE and LOE, indicating the higher the image enhancement quality.

4.2. Ablation Experiment

Based on the Retinex-Net framework, ablation experiments are designed to evaluate the actual effect of the improvements of the proposed method on network performance. The objective evaluation index of the ablation experiment is PSNR, to measure the noise of the image, and SSIM, to evaluate the overall quality of the image from the perspective of brightness and contrast. The experimental results are shown in Table 2. ECA denotes the efficient attention mechanism module, and DCA denotes the attention deep connection module. $\mathcal{L}_{dn}$ denotes the denoising loss function in the Decom-Net, and $\mathcal{L}_{color}$ denotes the color loss function in the Enhance-Net.

Table 2. Ablation experiment results.

Serial No	Basic Framework	Improvement of the Method	PSNR (dB)	SSIM
1	–	Retinex-Net	16.7622	0.5465
2	Retinex-Net	add ECA, do not add DCA	17.5834	0.6945
3		add ECA, add DCA	18.4488	0.7249
4	Retinex-Net+ECA+DCA	add, do not add $\mathcal{L}_{dn}$	18.8743	0.749
5		add $\mathcal{L}_{dn}$, do not add	18.9616	0.7443
6		add $\mathcal{L}_{color}$, add $\mathcal{L}_{dn}$	19.4536	0.7581

In the ablation experiment, No.1 tested the PSNR and SSIM values of the network without any improvement, based on Retinex-Net. Based on this result, the improvement effects of the PSNR and SSIM values under different improvement methods were observed.

The algorithm in this paper first introduces the attention mechanism ECA module in both the Decom-Net and the Enhance-Net; see No.2. It shows that the values of PSNR and SSIM are markedly improved, indicating that the attention mechanism can suppress the

noise caused by image decomposition, reduce the response of the network to irrelevant features, and make the network more focused on learning to improve image brightness. On the basis of introducing the ECA attention module, No.3 adds the DCA module, aiming at strengthening the interoperability between the attention modules, and allowing all the attention modules to be trained jointly. The experimental results show that the combination of ECA attention modules and DCA modules can further enhance the Retinex-Net's ability to extract the features of reflection images and restore the illumination images and colors.

Experiments No.4, No.5, and No.6 are based on experimental results No.3. ECA modules and DCA modules are added to further study the effect on the enhancement of the denoising loss $\mathcal{L}_{dn}$ and color loss $\mathcal{L}_{color}$. When only color loss has been added, the Enhance-Net can estimate the chromatic aberration more accurately, enhance the color saturation, and improve the values of PSNR and SSNR. Additionally, when only the denoising loss is added, the effect is better than when the color loss is added. It is shown that the loss of the mixed $L1$ of the multiscale structure similarity $MS - SSIM$ can effectively suppress the image noise and better retain the image's high-frequency information, including the image's key features and edge information. The experiment of No.6 shows that the denoising loss and color loss are added at the same time, and the proposed method obtains the best result.

4.3. Comparative Experiment

In this paper, we choose three low-illumination images and use different methods to enhance them. Through subjective comparison, we verify the enhancement effect of our algorithm from the evaluation set of the LOL data set combined with objective evaluation indicators. The enhanced visual effect of different methods is shown in Figure 8.

(**a**) Input (**b**) SRIE (**c**) NPE (**d**) GLADNet (**e**) EnlightenGAN (**f**) Retinex-Net (**g**) Proposed method

Figure 8. Comparison of enhancement results of different methods.

Figure 8 shows that the brightness enhancement degree of SRIE is limited, which is not enough to meet the visual needs of human eyes. The enhancement results of NPE are also darker, and there are more serious distortions in the brighter local areas. The enhancement results of GLADNet and EnlightenGAN have the problem of low saturation. The enhanced image of Retinex-Net has noise and artifacts, and the problem of image color distortion is more serious. The image enhanced by the proposed method has a balanced overall brightness and retains the image's details and the original color without local overexposure or artifacts.

To further prove the reliability of the proposed algorithm, we carried out experiments on the validation set of the LOL data set. Table 3 shows a quantitative comparison between the algorithm in this paper and other low illumination enhancement algorithms. It shows that the proposed method is the best in all indicators, and it is significantly improved compared with Retinex-Net, and performs better than the original algorithm. It has certain advantages in reducing noise, enhancing image brightness, and preserving image color.

Table 3. The results of different evaluation indicators.

	SRIE	NPE	GLADNet	EnlightenGAN	Retinex-Net	Proposed Method
SSIM	0.4977	0.5842	0.7115	0.6260	0.5594	0.7581
PSNR (dB)	11.8550	16.8034	19.3821	19.2303	16.7739	19.4536
NIQE	7.2873	8.2562	6.1744	4.528	9.7303	4.2874
LOE	575	439	493	445	1106	417

5. Conclusions

This paper proposed a modified Retinex-Net algorithm. By using an efficient attention mechanism method and constructing information interaction between the attention modules, the performance of attention mechanism can be maximized to a certain extent to ameliorate the network's learning ability.The denoising loss can effectively suppress noise and retain the essential structure of the image. The color loss restores the image to a more natural color, making the result more robust.The final enhancement result of the proposed algorithm is moderate in brightness, and is not too dark or too bright. The color is natural and not distorted, avoiding artifacts and excessive noise. The objective evaluation indexes such as PSNR and SSIM have also achieved better results. The primary research goal in the future is to improve the running time of the algorithm and to extend it to the task of object detection or object recognition under low light.

Author Contributions: Conceptualization, M.L.; methodology, M.L. and J.C.; software, J.C.; validation, J.C. and X.H.; formal analysis, M.L.; investigation, M.L. and J.C.; resources, M.L. and J.C.; data curation, J.C.; writing—original draft preparation, M.L. and J.C.; writing—review and editing, M.L., J.C.; visualization, J.C. and X.H.; supervision, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: This study did not involve humans and chose to exclude this statement.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hu, X.; Ma, P.; Mai, Z.; Peng, S.; Yang, Z.; Wang, L. Face hall-ucination from low quality images using definition-scalable inference. *Pattern Recognit.* **2019**, *94*, 110–121. [CrossRef]
2. Pan, R.; Zeng, L.; Wum, S.; Wang, R. Feature detection method for low illumination image. *Sens. Microsyst.* **2021**, *40*, 110–113+117.
3. Liu, M.; Su, T.; Wang, Y. Companding multiscale Research on Retinex image enhancement algorithm. *J. Harbin Univ. Sci. Technol.* **2020**, *25*, 93–99.
4. Jiang, J.L.; Liu, G.M.; Zhu, Z.; Huang, Z.; Zheng, J.Y. Dynamic Multi-Histogram Equalization Based on Fast Fuzzy Clustering. *Acta Electron. Sin.* **2022**, *50*, 167–176.
5. Wang, L.; Chang, X.; Ren, W. Color Image Enhancement Simulation Based on Weighted Histogram Equalization. *Comput. Simul.* **2021**, *38*, 126–131.
6. Xuan, D.; Guan, W.; Yi, P.; Wen, J. Fast Efficient Algorithm for Enhancement of Low Lighting Video. In Proceedings of the 2011 IEEE International Conference on Multimedia and Expo, Barcelona, Spain, 11–15 July 2011; IEEE: Toulouse, France; pp. 1–6.
7. He, K.; Sun, J.; Tang, X. Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2341–2353. [PubMed]
8. Land, E.H. The Retinex Theory of Color Vision. *Sci. Am.* **1978**, *237*, 108–128. [CrossRef] [PubMed]
9. Yang, J.; Xu, Y.; Yue, H.; Jiang, Z.; Li, K. Low-light image enhancement based on Retinex decomposition and adaptive gamma correction. *IET Image Process.* **2020**, *15*, 211–220. [CrossRef]
10. Liu, S.; Long, W.; He, L.; Li, Y.; Ding, W. Retinex-Based Fast Algorithm for Low-Light Image Enhancement. *Entropy* **2021**, *23*, 746. [CrossRef] [PubMed]

11. Lore, K.G.; Akintayo, A.; Sarkar, S. LLNet: A Deep Autoencoder Approach to Natural Low-light Image Enhancement. *Pattern Recognit.* **2017**, *61*, 650–662. [CrossRef]
12. Wei, C.; Wang, W.; Yang, W.; Liu, J. Deep Retinex Decomposition for Low-Light Enhancement. *arXiv*. Available online: https://arxiv.org/pdf/1808.04560.pdf (accessed on 11 May 2020).
13. Ignatov, A.; Kobyshev, N.; Timofte, R.; Vanhoey, K.; Van Gool, L. DSLR-Quality Photos on Mobile Devices with Deep Convolutional Networks. In Proceedings of the IEEE Computer Society, Bochum, Germany, 3–5 July 2017; pp. 3297–3305.
14. Li, J.; Wang, J.; Wan, G.; Li, Z.; Xu, D.; Cao, H.; Zhang, G. A combination of histogram equalization and A new image enhancement algorithm based on MSRCR. *J. Xidian Univ.* **2014**, *41*, 103–109.
15. Choi, D.H.; Jang, I.H.; Kim, M.H.; Kim, N.C. Color Image Enhancement Based on Single-Scale Retinex With a JND-Based Nonlinear Filter. In Proceedings of the International Symposium on Circuits and Systems (ISCAS 2007), New Orleans, LA, USA, 27 May 2007; IEEE: New York, NY, USA; pp. 1242–1254.
16. Lin, H.; Shi, Z. Multiscale retinex improvement for nighttime image enhancement. *Opt.-Int. J. Light Electron Opt.* **2014**, *125*, 7143–7148. [CrossRef]
17. Rahman, Z.U.; Jobson, D.J.; Woodell, G.A. Retinex processing for automatic image enhancement. *Proc. SPIE-Int. Soc. Opt. Eng.* **2004**, *13*, 100–110.
18. Jobson, D.J.; Rahman, Z.; Woodell, G.A. A multiscale retinex for bridging the gap between color images and the human observation of scenes. *IEEE Trans. Image Process.* **2002**, *6*, 965–976. [CrossRef] [PubMed]
19. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising with block-matching and 3D filtering. *Proc. SPIE-Int. Soc. Opt. Eng.* **2006**, *6064*, 354–365.
20. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
21. Ma, X.; Guo, J.; Tang, S.; Qiao, Z.; Chen, Q.; Yang, Q.; Fu, S. DCANet: Learningconnected attentions for convolutional neural networks. In Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 5–9 July 2021; pp. 1–6.
22. Zhao, H.; Gallo, O.; Frosio, I.; Kautz, J. Loss Functions for Image Restoration With Neural Networks. *IEEE Trans. Comput. Imaging* **2017**, *3*, 633–640. [CrossRef]
23. Feifan, L.; Bo, L.; Feng, L. Fast Enhancement for Non-Uniform Illumination Images using Light-weight CNNs. In Proceedings of the 28th ACM International Conference on Multimedia, Online, 12 October 2020; Association for Computing Machinery: New York, NY, USA; pp. 1450–1458.
24. Zhang, R.; Zhu, J.Y.; Isola, P.; Geng, X.; Lin, A.S.; Yu, T.; Efros, A.A. Real-time user-guided image colorization with learned deeppriors. *ACM Trans. Graph. (TOG)* **2017**, *36*, 121–131. [CrossRef]
25. Lehmann, E.L.; Casella, G. *Point Estimation Theory*; Beijing China Statistics Press: Beijing, China, 2004; p. 11.
26. Mittal, A.; Moorthy, A.K.; Bovik, A.C. Blind/Referenceless Image Spatial Quality Evaluator. In Proceedings of the 2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Pacific Grove, CA, USA, 6–9 November 2011; IEEE: New York, NY, USA; pp. 784–792.
27. Guo, X.; Li, Y.; Ling, H. LIME: Low-light Image Enhancement via Illumination Map Estimation. *IEEE Trans. Image Process.* **2016**, *26*, 982–993. [CrossRef] [PubMed]

electronics

Article

A Coverage Hole Patching Algorithm for Heterogeneous Wireless Sensor Networks

Xinmiao Lu [1,*], Yuhan Wei [1], Qiong Wu [2], Cunfang Yang [1], Dongyuan Li [1], Liyu Zhang [1] and Ying Zhou [2]

[1] School of Measurement-Control Technology and Communications Engineering, Harbin University of Science and Technology, Harbin 150080, China
[2] Heilongjiang Network Space Research Center, Harbin 150090, China
* Correspondence: lvxinmiao0611@hrbust.edu.cn; Tel.: +86-187-0462-8841

Abstract: The improvement of coverage is a critical issue in the coverage hole patching of sensors. Traditionally, VOPR and VORCP algorithms improve the coverage of the detection area by improving the original VOR algorithm, but coverage hole patching algorithms only target homogeneous networks. In the real world, however, the nodes in the wireless sensor network (WSN) are often heterogeneous, i.e., the sensors have different sensing radii. The VORPH algorithm uses the VOR in a hybrid heterogeneous network and improves the original algorithm. The patched nodes are better utilized, and the detection range is enlarged. However, the utilization rate of the patched nodes is not optimized, making it impossible to patch the coverage holes to the maximum degree. In the environment of hybrid heterogeneous WSN, we propose a coverage hole patching algorithm with a priority mechanism. The algorithm determines the patching priority based on the size of the coverage holes, thereby improving network coverage, reducing node redundancy, and balancing resource allocation. The proposed algorithm was compared under the same environment by simulation and analysis. The results show that our algorithm is superior to the traditional coverage hole patching algorithms in coverage rate, and can reduce node redundancy.

Keywords: coverage hole; hybrid heterogeneous wireless sensor network (WSN); priority mechanism; patching

1. Introduction

Classic harmful gas detection mainly uses wired fixed devices and portable instruments and other detection methods with poor flexibility, low real-time, inaccurate leakage point positioning and other shortcomings. A wireless sensor network with low cost, high real-time, good cooperation and other characteristics to achieve remote detection of harmful gases is a new idea. However, the wireless sensor network is a resource-limited network. How to use limited sensor nodes to improve network coverage performance in the wireless sensor network in the application of harmful gas detection has technical difficulties.

The deployment of network nodes determines the coverage performance of the network [1]. In the natural environment, it is impossible to predetermine the specific situation of the WSN in many monitoring applications due to the inaccessibility of personnel and the complexity and randomness of the environment [2]. Therefore, the sensors are often deployed randomly, without pre-determining the location of nodes in the monitoring area. Random deployment may greatly reduce the WSN coverage of the monitoring area, leaving coverage holes [3]. What is worse, wireless sensor nodes have a limited lifetime, and when energy is exhausted, the node fails, creating a coverage hole. Wireless sensors may fail due to various reasons, such as high temperature, malicious attacks, harsh environments, overuse, and lack of energy. Coverage holes appear right after the failure [4].

Once generated, the coverage holes affect the service quality of the entire network, making the network unable to capture the data of the object accurately, or to realize the intended

purpose [5,6]. Network coverage, an important indicator of the performance of WSN, depicts the quality of network service and network perception ability. The appearance of coverage holes significantly squeezes the coverage of WSN, undermining the coverage performance [7].

Yu et al. [8] improved the Voronoi-based optimization algorithm (VOR) into two coverage hole patching algorithms, namely, VORP and VORCP. The improvement enhances the coverage of the test area. Zhou et al. [9] proposed an improved coverage hole patching enhancement algorithm based on virtual force relocation. The improved algorithm adopts the Delaunay triangle algorithm to detect the local coverage holes between sensors, and then adopts the virtual force algorithm to patch the holes. The algorithm speeds up convergence, while elevating the network coverage. Cao et al. [10] devised the FISS algorithm for selecting patching nodes based on the fuzzy inference system (FIS). First, the coverage holes were identified by the adaptive sensing radius adjustment strategy, before patching the coverage holes. The FISS algorithm not only effectively improves the coverage rate, but also controls the energy consumption. Hao et al. [11] developed a three-dimensional (3D) dynamic detection and patching algorithm for coverage holes. The algorithm improves the node utilization rate and lowers the network coverage cost. It can meet the overall requirement of network coverage with fewer nodes, and consumes less mobile energy. The above algorithms work well in homogeneous networks, but the WSN nodes in the actual environment are often heterogeneous. In other words, the sensors in a heterogeneous network differ in sensing range, which results in coverage holes. Thus, the above algorithms are not effective in patching coverage holes in heterogeneous networks. Wu et al. [12] presented a coverage hole patching algorithm called VORPH, which uses the VOR in a hybrid heterogeneous network and improves the original algorithm. The patched nodes are better utilized, and the detection range is enlarged. However, the utilization rate of the patched nodes is not optimized, making it impossible to patch the coverage holes to the maximum degree. Each algorithm has limitations [13–15].

To solve the defects of the above algorithms in heterogeneous networks, we propose a coverage hole patching algorithm with priority mechanism in the environment of hybrid heterogeneous WSNs. Using the hybrid heterogeneous WSN, the we fixed static sensors to the area to be tested randomly, creating coverage holes for various reasons. Next, the sensors were moved by the priority mechanism to patch the coverage holes of the hybrid heterogeneous network.

2. Network Model and Coverage Hole Positioning Algorithm

2.1. Network Model

Suppose the rectangular detection area A is divided into m × n grids. A total of N static sensors $S = \{S_1, S_2 \cdots S_N\}$ are randomly deployed in the area. The static nodes fall into two categories: the nodes with a large sensing radius R_{S1}, and those with a small sensing radius R_{S2}. The radius of the former nodes is 1.5 times that of the latter nodes. The latter nodes are selected as the movable sensors that patch the coverage holes. The communication radii of the nodes with a large sensing radius and those with a small sensing radius are denoted by R_{C1} and R_{C2}, respectively, and $R_{C1} = 2R_{S1}$, $R_{C2} = 2R_{S2}$, and $R_{S2} = 1.5R_{S1}$. The communication radius length R_C between sensor nodes is twice the sensing radius R_S, ensuring that sensors can communicate with each other. The communication graphs between the sensors are connected. Each sensor can obtain the information (e.g., position) of other sensors.

In the real environment, the sensors with a large sensing radius will not surpass those with a small sensing radius by 0.5 times in terms of the sensing radius. The reason is that the overall cost of the system will hike, if sensors have an excessively large sensing radius. Let the center $S_i = (x_i, y_i)$ of a circle be the coordinates of a node. The area within the circle is the coverage of the WSN. The mobile nodes belong to M categories. The number of categories is the same with that of the sensors with a sensing radius of R_{S1}. In WSN research, network coverage and redundancy are two important metrics, which can be calculated by the following formulas.

1. Perceptibility

The perceptibility of node S_i for target t can be calculated by:

$$p(s_i, t_j) = \begin{cases} 1, & d(s_i, t_j) \leq R_s \\ 0, & d(s_i, t_j) > R_s \end{cases} \quad (1)$$

where $d(s_i, t_j)$ is the distance from the circle center of the sensor to the target. If target t is covered, the perceptibility of node S_i for target t equals 1; otherwise, the perceptibility is 0.

2. Network coverage

Coverage, the most crucial indicator of WSN performance, can be calculated by:

$$R_{cov}(s) = \frac{A_{cov}(s)}{A_{total}} = \frac{\sum_{x=1}^{m} \sum_{y=1}^{n} p(x, y, s)}{m \times n} \quad (2)$$

where A_{cov} is the coverage area, A_{total} is the area of the detection area, m is the length of the detection area and n is the width of the detection area.

3. Network redundancy

Redundancy indicates the degree of overlap between sensors. The greater the redundancy, the less even the distribution of the sensors. The network redundancy can be calculated by:

$$R_r(s) = \frac{A_{\mathrm{r}}}{A_{cov}} = \frac{\sum_{x=1}^{m} \sum_{y=1}^{n} p_r(x, y, s)}{\sum_{x=1}^{m} \sum_{y=1}^{n} p(x, y, s)} \quad (3)$$

where A_{cov} is the area of the sensor coverage, A_r is the overlapping area between sensors, and $p_r(x, y, s)$ indicates that grid node (x, y) can be perceived by sensor S_i and sensor S_j.

2.2. Covering Hole Positioning Based on Circumscribed Circles

In Figure 1, circles A, B, and C are the initially deployed static nodes, among which a hole is generated. Make a circle O that circumscribes circles A, B, and C at the same time. The larger the area of circle O, the wider the coverage hole, and the greater the coverage propriety. When the radius of circle O is less than 1/4 of the radius of the sensors with small sensing range, it is not recorded as a coverage hole, and the center O of circle O is the location where the mobile nodes will move to.

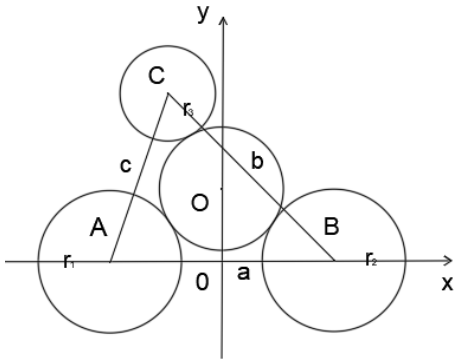


Figure 1. Determining the specific location of the node to be patched.

In Figure 1, the radii of the three circles A, B, and C on the known plane are denoted by r_1, r_2, and r_3, respectively, with $r_1 = r_2 = 1.5r_3$; AB $= a$, BC $= b$, and CA $= c$. Taking the straight line of segment AB as the horizontal axis, and the midpoint of AB as the origin, a rectangular coordinate system is established, with point C falling on the upper half plane of the rectangular coordinate system. Suppose A$(-a/2, 0)$, B$(a/2, 0)$, and C(x_3, y_3). Let

$O(x, y)$ and r be the center and radius of the circle O to be solved, respectively. Then, C can be determined by the following equation set (note that $y_3 > 0$):

$$\begin{cases} (x_3 - \frac{a}{2})^2 + {y_3}^2 = b^2 \\ (x_3 + \frac{a}{2})^2 + {y_3}^2 = c^2 \end{cases} \Rightarrow \begin{cases} x_3 = \frac{c^2 - b^2}{2a} \\ y_3 = \frac{2S}{a} \end{cases} \tag{4}$$

where $p = \frac{a+b+c}{2}$ is the half perimeter of Δ_{ABC}. The area of the circle can be described by Heron's formula as $S = \sqrt{p(p-a)(p-b)(p-c)}$. Then, the center $O(x, y)$ of circle O can be determined by the following equation set:

$$\begin{cases} (x + \frac{a}{2})^2 + y^2 = (r + r_1)^2 \\ (x - \frac{a}{2})^2 + y^2 = (r + r_2)^2 \\ (x + x_3)^2 + (y - y_3)^2 = (r + r_3)^2 \end{cases} \tag{5}$$

Thus, we have:

$$\begin{cases} x = \frac{(r_1 - r_2)(2r + r_1 + r_2)}{2a_3} \\ y = \frac{\sqrt{-[a^2 - (r_1 - r_2)^2][a^2 - (2r + r_1 + r_2)^2]}}{2a} \end{cases} \tag{6}$$

Substituting $O(x, y)$ into Formula (5), the same solution $r = \frac{P \pm 4S \cdot Q}{R}$ can be obtained, with P, Q and R being:

$$P = \sum_{cyc} b^2[-b^2 r_1 + c^2(r_1 + r_2) - (r_1 - r_2)(r_1 - r_3)(2r_1 + r_2 + r_3)] \tag{7}$$

$$Q = \prod_{cyc} \sqrt{b^2 - (r_2 - r_3)^2} \tag{8}$$

$$R = \sum_{cyc} b^2[b^2 - 2c^2 + 4(r_1 - r_2)(r_1 - r_3)] \tag{9}$$

In this way, the location of the circle O to be solved can be obtained, that is, where the mobile nodes to be moved to.

3. Improved Coverage Hole Patching Algorithm with Priority Mechanism

We propose a coverage hole patching algorithm for hybrid heterogeneous WSN with priority mechanism based on circumscribed circles. The priority mechanism assigns a priority to each hole to be patched. First, the holes are compared in size, and the larger hole is given a higher priority. Then, a limited number of mobile nodes is used to patch the holes by the priority, thereby optimizing the coverage.

3.1. Situations of Coverage Holes

In Figure 2, N is the center of the circle. The nodes A, B and C are far apart, resulting in very large coverage holes. A mobile node is set at the center of the common circumscribed circle of the three nodes. Although it does not overlap nodes A, B and C, the mobile node patches the coverage holes to the maximum possible extent. In addition, since the distance between nodes A, B and C exceeds the communication distance, the mobile node increases the communication coverage of nodes A, B and C, and enhances the connectivity of communication.

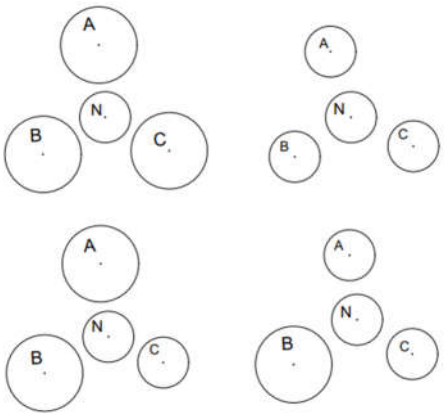


Figure 2. Nodes A, B and C are far from each other.

In Figure 3, the distance between nodes A, B and C is relatively short, resulting in a large coverage hole. A mobile node is set at the center of the common circumscribed circle of the three nodes to cover nodes A, B and C, thereby making up the hole.

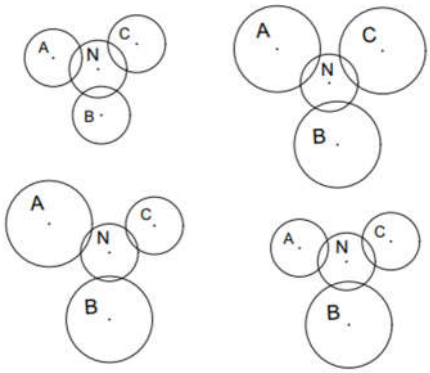


Figure 3. Nodes A, B and C are close to each other.

In Figure 4, two nodes A and B intersect each other, and the other node C has no overlap with A or B. The sensing ranges of nodes A and B cover each other, resulting in a large coverage hole. A mobile node is set at the center of the common circumscribed circle of the three nodes so that the position of the mobile node can maximize the patching of the hole.

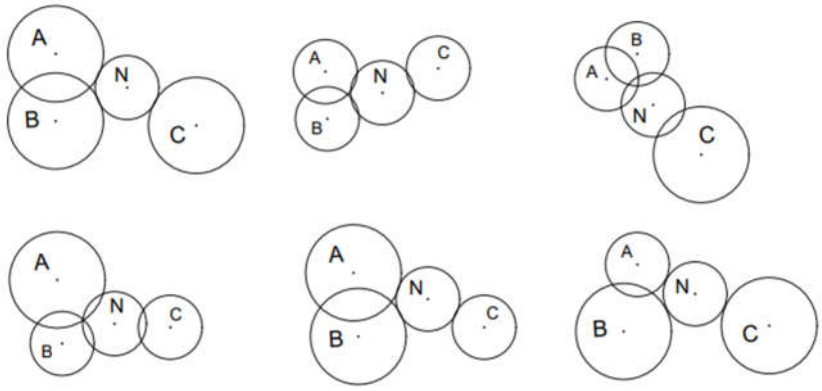


Figure 4. Nodes A and B intersect each other.

In Figure 5, nodes A and B intersect, and nodes B and C intersect, but A and C do not intersect. The sensing ranges of nodes A and B cover each other, and the sensing ranges of nodes B and C cover each other. In the common circumscribed circle of the three nodes, a mobile node is set at the center of the circle to cover nodes A, B and C to repair the hole. The setting of the position of the mobile node maximizes the patching of the hole.

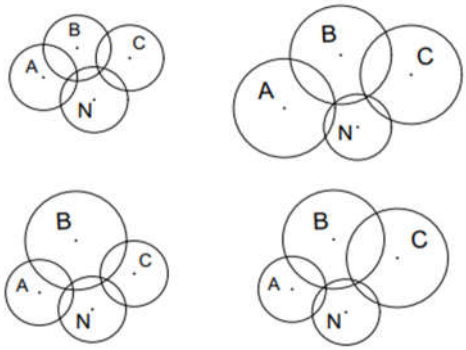


Figure 5. Multiple nodes intersect each other.

As shown in Figures 2–6, the hole in Figure 2 is the largest, and its patching priority is the highest. For Figures 3–5, the patching priority increases with the size of the circumscribed circle. Figure 6 needs not to be covered, for no hole is generated.

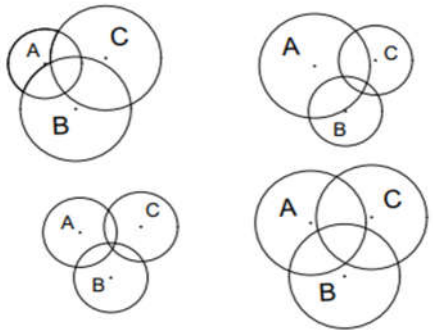


Figure 6. Nodes A, B and C intersect each other in pairs.

3.2. Steps

The specific hole patching steps are as follows:

1. After randomly arranging static nodes, obtain the specific location of each node and store it in the list of locations v.
2. Create a new list of the patching locations List and a temporary information list temp.
3. Divide the adjacent three nodes into a group, make a common circumscribed circle of each group of three nodes, calculate the area of the circumscribed circle $G_1 \cdots G_N$, and store it into the List.
4. Initialize the temporary information list temp.
5. Carry out a loop, traverse the list, directly update the first G_1 in the list and its related location information into temp. Continue to traverse the list, and take G_2 to compare it with G_1 in temp. If $G_2 > G_1$, add G_2 and its related location information into temp, and remove G_1 and its related location information from the list. Otherwise, retain G_1 in temp. When all items in the list are traversed, send the marked locations in the List to the mobile node to guide its movement.
6. Repeat Steps 3–5 until the mobile nodes are all used, and terminate the loop.

4. Simulation Experiments

To demonstrate the effectiveness of our algorithm, network coverage and redundancy were analyzed, using the software simulation platforms of Win10 and Matlab 2016b.

The detection area is of the size 125 m $\times$ 100 m. A total of 25 static nodes were deployed randomly inside the area, $S = \{S_1, S_2 \cdots S_{25}\}$. The nodes fall into two categories. The sensing radii of the two types of nodes are R_{S1} and R_{S2}, respectively; the communication radii of the two types of nodes are R_{C1} and R_{C2}, respectively. These radii satisfy $R_{C1} = 2R_{S1}$, $R_{C2} = 2R_{S2}$, and $R_{S2} = 1.5R_{S1}$. The sensing radii of the nodes are $R_{S1} = 8$ m and

$R_{S2} = 12$ m. The communication radii assure that the nodes within the one-hop range can exchange information. The center $S_i = (x_i, y_i)$ of each circle represents the coordinates of a node. The area enclosed in the circle indicates the coverage of the WSN. The types and sensing radii of the mobile nodes are the same as those of the sensors with a sensing radius of R_{S1}. There are a total of 15 sensors. Simulation experiments were carried out to verify the proposed method. Figure 7 shows the coverage of the static nodes. The coverage was calculated to be 62.46%. There are 27 potential coverage holes in the detection area, which is larger than the number of mobile nodes that can be used.

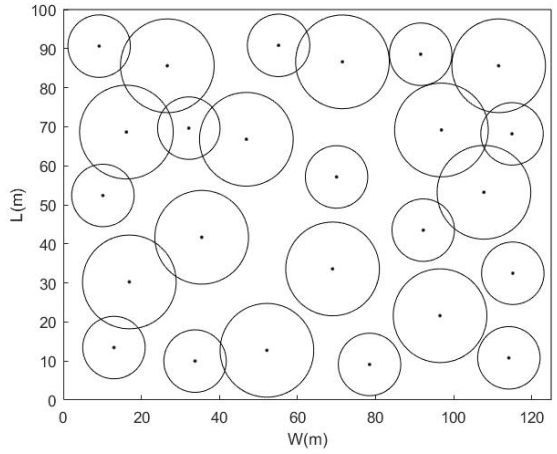


Figure 7. Randomly assigned static nodes.

The sizes of the coverage holes of the static nodes in Figure 7 are different. Our algorithm was adopted to simulate the patching of the coverage holes. The simulation results are shown in Figure 8. The calculated coverage rate was 79.13%, which is 16.67% higher than the coverage rate before patching. The redundancy was 8.24%. This means the coverage holes of the WSN have been well repaired, reflecting the effectiveness of our algorithm.

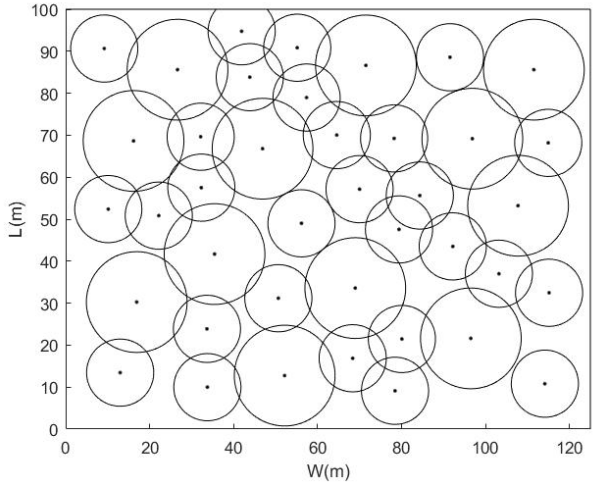


Figure 8. Nodes after patching by our algorithm.

Using the same parameters, our algorithm was compared with VORCP [1] and VORPH [2] in terms of coverage and redundancy, with the goal of revealing the superiority of our algorithm intuitively.

Figure 9 shows the simulation results on patching the coverage holes in the scene of Figure 7, using VORCP. The coverage (75.04%) was 12.58% greater than that before patching. The redundancy was 10.54%. Figure 10 shows the simulation results on patching the coverage holes in the scene of Figure 7, using VORPH. The coverage (76.78%) was 14.32% greater than that before patching. The redundancy was 11.13%. Table 1 shows

that our algorithm performed the best in hole patching and coverage enhancement, under the same environment. In addition, the redundancy after patching of our algorithm was lower than that of the other two algorithms. This fully demonstrates the superiority of our algorithm. Table 1 shows the comparison result of the algorithms.

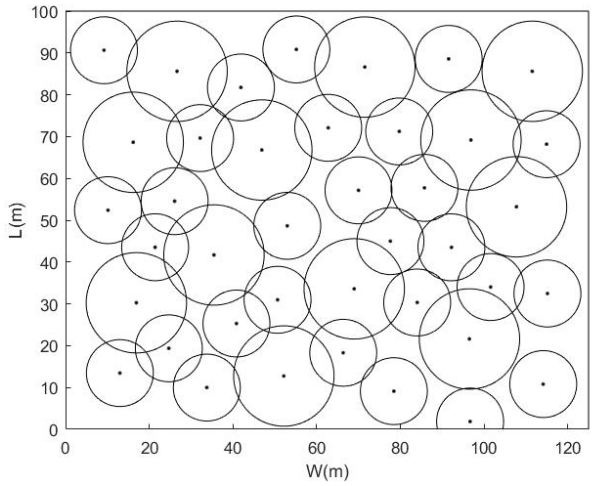


Figure 9. Nodes after VORCP patching.

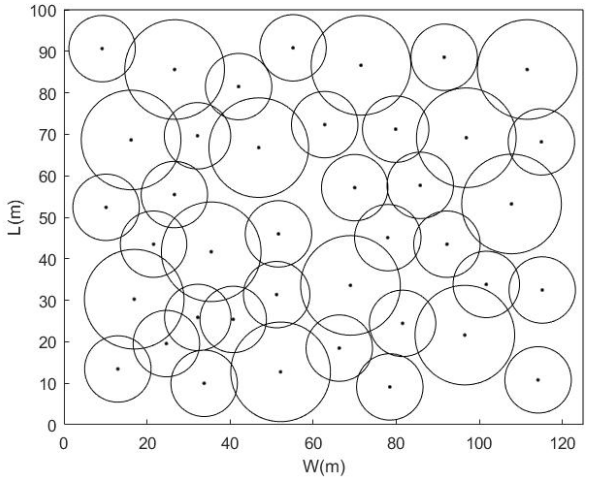


Figure 10. Nodes after VORPH patching.

Table 1. Algorithm comparison.

	Coverage	Increased Coverage	Redundancy
VORCP	75.04%	12.58%	10.54%
VORPH	76.78%	14.32%	11.13%
Our Algorithm	79.13%	16.67%	8.24%

5. Conclusions

To solve the defects of traditional coverage hole patching algorithms, we proposed a coverage hole patching algorithm with priority mechanism in hybrid heterogeneous WSN environment. Specifically, we analyzed the network model and the hole positioning algorithm, and studied the coverage and priority of the coverage holes. Then, our algorithm was subjected to simulation and analysis, and compared with VORCP and VORPH. The results show that, in the hybrid heterogeneous WSN, our algorithm outperformed the traditional coverage hole patching algorithms in both coverage and redundancy.

Author Contributions: Conceptualization, X.L.; methodology, X.L.; software, Y.W.; validation, X.L., Q.W. and D.L.; writing—original draft preparation, C.Y., L.Z. and Y.Z.; writing—review and editing, Y.W.; project administration, X.L.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Heilongjiang Provincial Natural Science Foundation of China, grant number YQ2022F014.

Acknowledgments: The authors acknowledge Heilongjiang Provincial Natural Science Foundation of China (grant number YQ2022F014).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yue, Y.G.; He, P. A comprehensive survey on the reliability of mobile wireless sensor networks: Taxonomy, challenges, and future directions. *Inf. Fusion* **2018**, *44*, 188–204. [CrossRef]
2. Joshitha, K.L.; Jayashri, S. A novel redundant hole identification and healing algorithm for a homogeneous distributed Wireless Sensor Network. *Wirel. Pers. Commun.* **2019**, *4*, 1261–1282. [CrossRef]
3. Etancelin, J.M.; Fabbri, A.; Guinand, F. DACYCLEM: A decentralized algorithm for maximizing coverage and lifetime in a mobile wireless sensor network. *Ad Hoc Netw.* **2019**, *87*, 174–187. [CrossRef]
4. Xu, Y.; Ding, O.; Qu, R. Hybrid multiobjective evolutionary algorithms based on decomposition for wireless sensor network coverage optimization. *Appl. Soft Comput.* **2018**, *68*, 268–282. [CrossRef]
5. Fang, W.; Song, X.; Wu, X. Novel efficient deployment schemes for sensor coverage in mobile wireless sensor networks. *Inf. Fusion* **2018**, *41*, 25–36. [CrossRef]
6. SO-IN, C.; Nguyen, T.G.; Nguyen, N.G. An efficient coverage hole-healing algorithm for area-coverage improvements in mobile sensor networks. *Peer-to-Peer Netw. Appl.* **2019**, *3*, 541–552. [CrossRef]
7. Zhai, S.; Tang, Z.; Wang, D. Enhancing received signal strength-based localization through coverage hole detection and recovery. *Sensors* **2018**, *18*, 2075. [CrossRef] [PubMed]
8. Yu, Q.; Wang, W.D.; Li, L.J.; Qian, T. Improved Recovery Algorithms of Coverage Holes in Hybrid WSN. *J. Univ. Electron. Sci. Technol. China* **2017**, *46*, 534–539.
9. Zhou, F.; Guo, H.T.; Yang, Y. An Improved Virtual Force Relocation Coverage Enhancement Algorithm. *J. Electron. Inf. Technol.* **2020**, *42*, 2194–2200.
10. Cao, Y.; Wang, Z.L. A repair algorithm for coverage hole in mobile WSNs. *J. Terahertz Sci. Electron. Inf. Technol.* **2021**, *19*, 145–149.
11. Hao, Z.J.; Xu, H.W.; Dang, X.C.; Duan, Y. A Dynamic Detection and Repair Algorithm for Three-Dimensional Coverage Holes in WSN. *Comput. Eng.* **2020**, *46*, 178–186.
12. Wu, Q. Research on Gap Fixing and Data Transmission Technology of Wireless Sensor Network for Harmful Gas Detection. Ph.D. Thesis, Harbin University of Science and Technology, Harbin, China, 2019.
13. Sirajuddin, M.; Rupa, C. Techniques for Enhancing the Performance of TCP in Wireless Networks. *Artif. Intell. Evol. Algorithms Eng. Syst.* **2015**, *324*, 159–167.
14. Sirajuddin, M.; Rupa, C.; Bhatia, S. Hybrid Cryptographic Scheme for Secure Communication in Mobile Ad Hoc Network-Based E-Healthcare System. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 9134036. [CrossRef]
15. Bharath, G.; Rupa, C.; Karthik, M. Revelation of Geospatial Information using Augmented Reality. In Proceedings of the 2021 Sixth International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 25–27 March 2021.

electronics

MDPI

Article

Research on Small Acceptance Domain Text Detection Algorithm Based on Attention Mechanism and Hybrid Feature Pyramid

Mingzhu Liu *, Ben Li and Wei Zhang

The Higher Educational Key Laboratory for Measuring & Control Technology and Instrumentation of Heilongjiang Province, Harbin University of Science and Technology, Harbin 150080, China
* Correspondence: lmz@hrbust.edu.cn

Abstract: In the traditional text detection process, the text area of the small receptive field in the video image is easily ignored, the features that can be extracted are few, and the calculation is large. These problems are not conducive to the recognition of text information. In this paper, a lightweight network structure on the basis of the EAST algorithm, the Convolution Block Attention Module (CBAM), is proposed. It is suitable for the spatial and channel hybrid attention module of text feature extraction of the natural scene video images. The improved structure proposed in this paper can obtain deep network features of text and reduce the computation of text feature extraction. Additionally, a hybrid feature pyramid + BLSTM network is designed to improve the attention to the small acceptance domain text regions and the text sequence features of the region. The test results on the ICDAR2015 demonstrate that the improved construction can effectively boost the attention of small acceptance domain text regions and improve the sequence feature detection accuracy of small acceptance domain of long text regions without significantly increasing computation. At the same time, the proposed network constructions are superior to the traditional EAST algorithm and other improved algorithms in accuracy rate P, recall rate R, and F-value.

Keywords: lightweight network structure; hybrid feature pyramid; BLSTM network; acceptance domain; attention mechanism

1. Introduction

In the last several years, the evolution of cultural exchanges and network technology has become faster and faster, and considerable video resources, such as movies, documentaries, and various shows, have sprung up on the Internet and have begun to affect people's lives. Due to cultural differences and personal understanding differences, there are still many text messages that the audience has difficulty noticing and understanding, such as street signs, trademarks, and commodities in videos, even though there are subtitles in these videos. If this information can be extracted, recognized, and translated, it will be helpful for the audience to understand the video information. Therefore, text detection in natural scenes is of great research significance.

Directly applying target detection technology to natural scene text detection is the most direct way in early natural scene text detection. Among them, the representative ways are Faster-RCNN [1] and SSD [2]. However, Faster-RCNN and SSD still have low efficiency for long text and slanted text detection due to the complexity of natural scene text detection, such as different types, unfixed position and arrangement direction, and the diversity of size and shape [3]. Thus, researchers consider improving the construction of the original target detection algorithm, making it more suitable for natural scene text detection.

Methods of traditional text detection and some text detection methods based on deep learning mostly use multi-stage constructions. Each level needs to be adjusted

and optimized during training. This inevitably affects the final text location effect and is very time-consuming. To solve the above problems, Kuangshi Technology Company proposed an Efficient and Accurate Scene Text Detector algorithm (EAST) in 2017 [4]. The EAST algorithm comes up with an end-to-end text detection method, which can directly forecast the text line region and eliminate multiple stages in the middle, such as candidate region aggregation, text segmentation, and post-processing. The construction of the EAST model is simple. Since it adopts FCN full convolution neural network construction and non-maximum suppression (NMS) merging process, the accuracy of text detection on ICDAR2015 can reach 80.57% [5]. Furthermore, the EAST algorithm is no longer limited to text size and text proportion and can detect any size and scale of text because it abandons the idea of anchor in Faster-RCNN and directly forecasts text lines [6]. Additionally, the EAST algorithm uses locality-aware NMS to filter the generated geometry, contributing to reducing the probability of text boxes being deleted falsely [7]. The EAST algorithm can forecast two different forms of the text box, namely slanted text and irregular text, which cannot be detected by CTPN and other algorithms [8]. However, the EAST algorithm does not pay enough attention to the small acceptance domain text area and does not deal with the small acceptance domain feature image, which will increase the difficulty of follow-up processing [9]. To solve the problem of insufficient attention to the small acceptance domain, based on the EAST algorithm, combined with the characteristics of text in natural scenes, this paper improves the feature extraction network, optimizes the feature pyramid construction, and completes the text detection.

2. Preliminaries

The EAST algorithm is divided into two stages: full convolution network (FCN)for forecasting text boxes and local perception non-maximum suppression (LNMS). The network construction of the FCN (see Figure A1 in Appendix A), where f_i (i is 1, 2, 3, 4), represents the output feature map of different size feature pyramids, and h_i denotes the output feature map of the fusion of different acceptance domains. The network construction can be decomposed into three parts: Feature extractor, Feature merge, and Output [10]. In the first stage, the image is inputted into the FCN network construction, then the single-channel pixel-level text fractional feature map and multi-channel geometric feature map are generated. Generally, the text area is marked by two geometric shapes: a rotating box and a horizontal box [11]. The EAST algorithm also sets the corresponding loss function for each geometric shape and puts a threshold into each prediction region, and the geometry whose score exceeds the threshold is considered effective. Afterward, the result processed by LNMS is taken as the final detection result of the output [12].

The original EAST algorithm uses VGG16 as the feature extraction network [13]. Although VGG16 is quite mature, it has only 16 tiers, otherwise, a mass of parameters is needed for the calculation during the training. Therefore, Google launched the latest lightweight network MobileNetV3 in May 2019 [14]. The new version of Mobile Net adopts more new features, such as the introduction of 5×5 depth convolution into MobileNetV3 to replace the original 3×3 depth convolution, and the introduction of a squeeze-and-excitation (SE) [15] module and h-swish (HS) [16] activation function to improve the model accuracy. As revealed from the comparison of VGG16 construction between MobileNetV3 and EAST original algorithm, the parameter of VGG16 is 138 million, while the parameter of MobileNetV3 is only 0.054 million, which accounts for only 4% of the VGG16 network parameter. The computation of VGG16 is 15.3 billion, while that of MobileNetV3 is only 219 million, which accounts for only 1.4% of the VGG16 network. Simultaneously, the accuracy of MobileNetV3 is 75.2%, which is higher than that of VGG16 (71.5%). Mobile Net network has great merits in lightweight neural networks, such as being smaller in size, a smaller computation, and higher accuracy. In this paper, based on VGG, deep separable convolution is used to take the place of the ordinary convolution of the traditional VGG network, and the attention mechanism is integrated into the VGG network to obtain a more efficient feature extraction effect.

In MobileNetV3 construction, deep Separable Convolution (Depth-wise Convolution) is used [17]. It is constituted of a tier of Depth-wise convolution and a tier of Point-wise convolution. Each tier of convolution is followed by batch normalization and ReLU activation function. The difference between deeply separable convolution and standard convolution is that the parameters and the computation are significantly reduced when the accuracy remains unchanged. For the contrast between depth-separable convolution and traditional convolution, see Figure A2 in Appendix A.

3. Methods

3.1. Lightweight Network Structure with a Mixed Attention Mechanism

SE module is introduced in MobileNetV3, and it is a kind of attention mechanism [18]. SE block is a sub-construction that can be embedded in other classification or detection models rather than a complete network construction.

3.1.1. Channel Domain Attention Mechanism

This attention mechanism is divided into three sections: Squeeze, Excitation, and Attention [19]. SE module construction see Figure A3 in Appendix A.

The squeeze function in the squeeze construction as in Equation (1) [19]:

$$z_C = F_{\mathrm{sq}}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \tag{1}$$

where $u_c(i,j)$ represents the feature value of the i-th row and j-th column of the c-th channel in the feature map u; H and W denotes the length and width of the feature map severally; z_C indicates the output of the c-th channel. The squeeze function shown in Equation (1) completes a global average operation, namely global average pooling.

The excitation function in the excitation construction as in Equation (2) [19]:

$$s_{\mathrm{c}} = F_{ex}(z_{\mathrm{c}}, W) = \sigma(g(z_c, W)) = \sigma(W_2 \frac{1}{1 + e^{-W_1 z_c}}) \tag{2}$$

where zc is the squeezing function shown in Equation (1). σ is the ReLU function; the dimensions of W_1 and W_2 are $C/r \cdot C$ and $C \cdot C/r$, where r is a scaling parameter. The purpose of introducing parameter r is to decrease the number of accesses, thus decreasing the total computation; r is set to 16 in this paper. By training and learning the weight values of W_1 and W_2, a one-dimensional weight sc is acquired to activate each tier channels which is shown in Equation (2). The final output $\widetilde{X}_C$ of the block is obtained by the scale function as in Equation (3) [19]:

$$X_c = F_{scale}(u_c, s_c) = s_c \cdot u_c \tag{3}$$

where s_c is the output of the c-th feature map of the excitation function, and u_c is the output feature value of the c-th channel of the feature map. As shown in Equation (3), by multiplying the eigenvalues of disparate channels with disparate weights, the key access weight is large and the non-key channel weight is small, which can enhance the attention to the key channel domain.

3.1.2. Spatial Domain Attention Mechanism

Another kind of attention mechanism is the attention mechanism of the space domain, and its construction (see Figure A4 in Appendix A). According to the feature graph obtained from the former processing, the maximum value is globally pooled on the channel, and the average value is globally pooled on the dimension. Then, the feature information of different locations in the feature graph is compared and extracted, and the weight of each location is acquired. The output $O(i,j)$ of the aftermath of the maximum global pooling as in Equation (4) [18]:

$$O(i,j) = \max(I_m(i,j)), m \in (1, C) \tag{4}$$

where $I_m(i,j)$ indicates the feature value of the feature map at the (i,j) position on the m-th channel, C is the total channel number of the input feature map.

The average global pooling handle as in Equation (5) [18]:

$$P(i,j) = \frac{1}{d}\sum_{n=1}^{d} T_n(i,j) \tag{5}$$

where $T_n(i,j)$ represents the feature value at the location (i,j) of the n-th access, and d is the number of the amount of accesses in the input feature map. The two feature maps are linked in the access dimension utilizing the access link, then a 1×1 convolution kernel is used for convolution operation. Finally, a sigmoid function is used to process an output. This attention mechanism can effectively draw the important message of each location of the feature map and the main features by assigning weights and suppressing the background information.

3.1.3. Hybrid Attention Mechanism Module

In the two attention mechanisms, the attention in the spatial domain neglects the message in the access domain, and the image features in each access are treated equally. This will make the spatial domain operation limited to the original image feature extraction stage, resulting in the application of the neural network in each tier of the interpretability to be not strong. On the other hand, the attention of the channel domain is to pool the overall situation, which means to obtain information directly and to ignore the local news in every access. Therefore, a lightweight hybrid-domain attention mechanism model is constructed by combining the characteristics of the two attention mechanisms and can perform attention in access and spatial dimensions [20]. The construction of the Convolution Block Attention Module (CBAM) is illustrated in Figure 1.

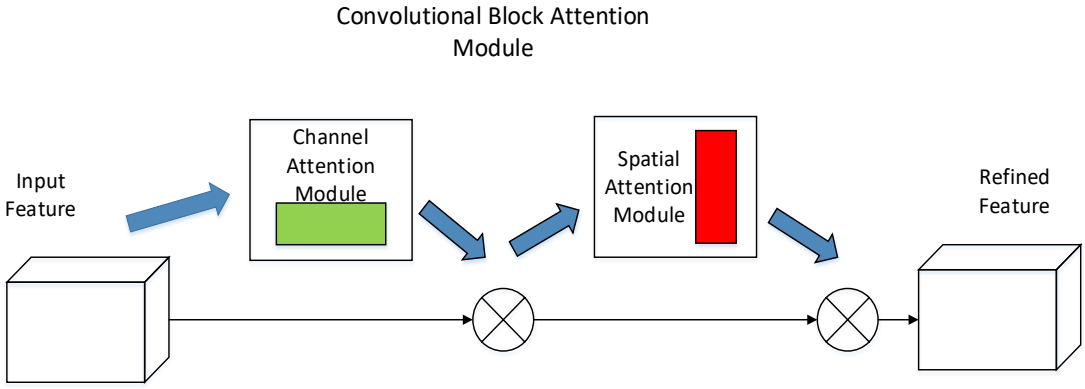


Figure 1. CBAM construction diagram with the hybrid attention mechanism module.

As mentioned above, CBAM is made up of two sub-modules; one is the Channel Attention Module (CAM), and the other is the Spatial Attention Module (SAM). CBAM is to multiply the input feature map with the CAM to get a new feature map, then multiplies the new feature map with SAM to get the resulting output. Thus, the channel domain and the spatial domain are combined. Because SE and CBAM belong to the access domain attention module and hybrid domain attention module, respectively, CBAM has a better attention concentration effect than the SE module. Therefore, this paper uses the attention mechanism module CBAM to replace the SE module in the lightweight network MobileNetV3. At the same time, combined with ResNet-50, this paper deepens the number of network tiers to 50 to achieve the purpose of enriching high-level semantic information [21].

3.2. Optimization Way of Small Acceptance Domain Based on a Hybrid Feature Pyramid

In the FCN network construction, the EAST algorithm adopts a four-tier feature pyramid construction and merges in the feature fusion part. Through deconvolution operation, the small-scale feature map is expanded to the same size as the large-scale feature map, and then the access dimensions are connected by adding in the channel domain. From the FCN construction (see Figure A1 in Appendix A), it can be seen that in the feature merging process, the small acceptance domain feature map at the bottom of the feature extraction network is merged first. This construction will lead to the small acceptance domain feature map only [22] through a little convolution calculation, and when the number of accesses is small, the proportion of fused information is less [22]. It will lead to the problem that the EAST algorithm does not pay enough attention to the small acceptance domain text, and the small acceptance domain text feature map is less concerned and operated, which will not be conducive to its feature extraction. For the sake of solving the problem of the small acceptance domain, a hybrid feature pyramid construction is applied to optimize feature collection and feature merging [23].

As shown in Figure 2, the hybrid feature pyramid before optimization contains top-down and bottom-up paths, and then the two paths are combined. A path with the same direction as the original EAST algorithm and a path with the opposite direction are fused as the final output. In this method, the latter small acceptance domain feature map is located at the bottom of the feature merging. Thus, the problem of insufficient attention to the small acceptance domain existing in the feature pyramid construction of the original EAST algorithm is solved through the two-path merging. On the other hand, in the original feature pyramid construction, too much attention is paid to the large acceptance domain, which leads to too much feature extraction of the large receptive feature map. Through the test, it is found that when the two-way path is used to merge the large acceptance domain regions, the detection accuracy is not significantly improved, but it will bring the burden of calculation. Therefore, the hybrid feature pyramid construction is redesigned in this paper. The new pyramid construction proposed in this paper only merges the first two tiers with a smaller acceptance domain of the original feature pyramid, then merges with the path in the other direction. The specific optimization construction is shown in Figure 3. The dotted box in Figure 3 shows the construction of the acceptance domain optimization part in the improved feature pyramid. The feature map of the new feature pyramid construction still comes from the output of the feature extraction network. The two feature extraction paths of the new construction and the original construction are almost the same, as shown in Figure 2, except that the path direction of the feature merging part is different, so it not only solves the problem that the EAST algorithm does not pay enough attention to the text in the acceptance domain but also increases the computation that can be ignored.

However, in the EAST algorithm, enough attention has been paid to the large acceptance domain so that the features of the large receptive feature graph are rich enough. Therefore, the feature detection accuracy of the large receptive feature graph is not significantly improved by the bi-directional path combination but will bring computational burden. Therefore, the mixed-feature pyramid construction is redesigned in this paper. Only the first two tiers of the feature pyramid are merged (the two tiers with smaller acceptance domains) on the merge path in the opposite direction, and then the fusion is carried out with the path in the other direction. The concrete construction is shown in Figure 3 (the dotted box is the optimized part). Compared with the original construction, the improved construction retains the characteristics of the original construction, but the improved construction is more concise and lightweight. Due to the features of the merger figure coming from the output of the network of feature extraction, two feature extraction paths are the same (according to the Figure 2, feature extraction is conducted only once), only in part are features combined in a different direction. Therefore, the improved construction will only bring a slight increase in the computation during the feature combination of the first two tiers. Taking the 50-tier feature extraction network as an example, at least 50 convolution operations are required during the extraction and 5 times during the feature combination.

Moreover, the number of channels in the feature map of the small acceptance domain are less, and the computation during the combination is lower than that of the large acceptance domain combination. Therefore, the calculation amount of the final parameter number is less than 1/50 of the original algorithm, while the original mixed feature pyramid will bring about 1/10 of the computation. So, the optimized mixing characteristics of the pyramid sense retains the original construction with an upgrade to small wild features. It solves the sense that there is too little wild text, and there are less acceptance domain characteristics graph processing in feature extraction because the characteristics of the pyramid before two tiers merge will not bring a burden for the network of the computation.

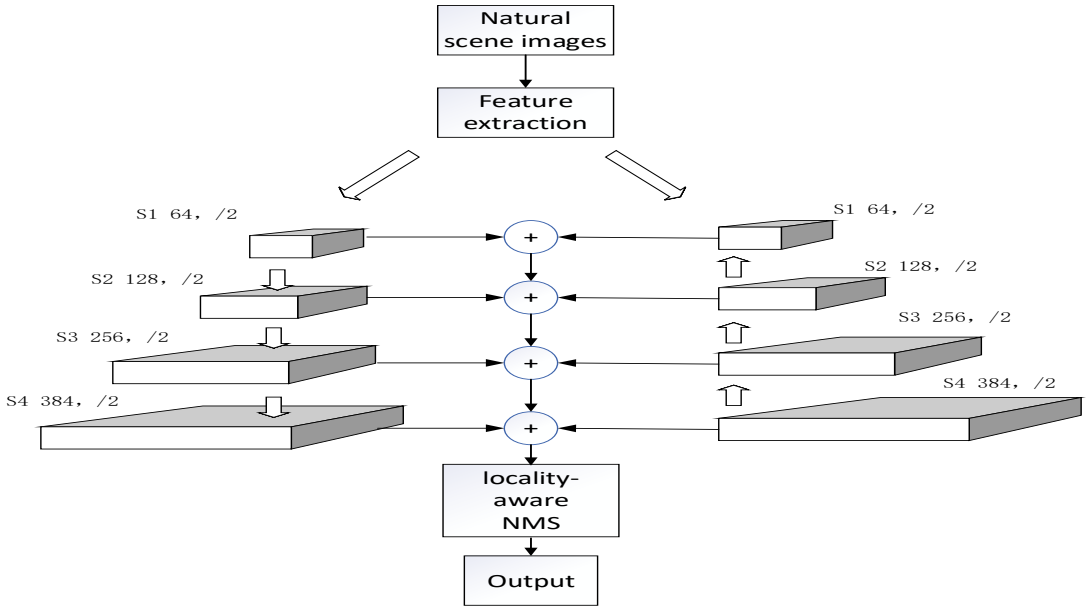


Figure 2. Schematic diagram of the original hybrid feature pyramid construction.

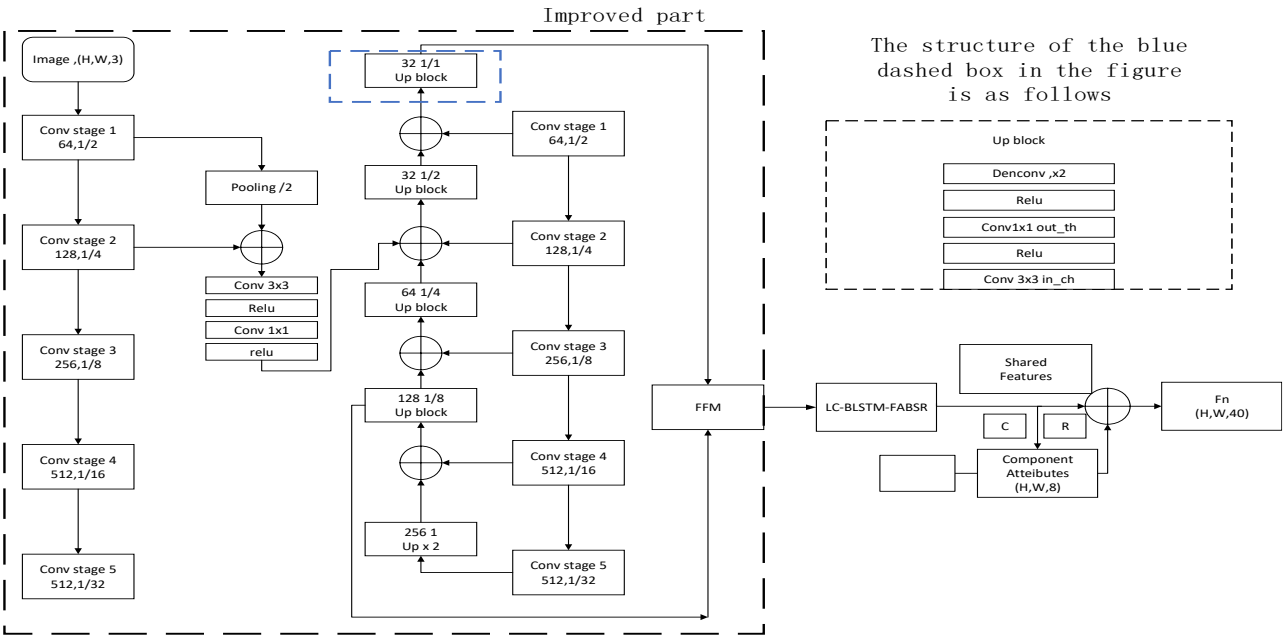


Figure 3. Construction block diagram of the improved algorithm.

At present, to solve the problem of the unbalanced distribution of targets, most methods adopt balanced sampling and hard negative mining. Although this will improve the

network performance, this technology inevitably incorporates a stage and more parameters to adjust the pipeline. This is contrary to the design of the EAST algorithm. To simplify the training process, equilibrium cross-entropy (the same as solving class imbalance, β is the number of counterexamples/total number of samples) is used in this paper, and the formula as in Equation (6):

$$L_{\mathrm{s}} = balanced - xent(\hat{\mathrm{Y}}, \mathrm{Y}^*) = -\beta \mathrm{Y}^* \log \hat{\mathrm{Y}} - (1-\beta)(1-\mathrm{Y}^*) \log(1-\hat{\mathrm{Y}}) \quad (6)$$

$\hat{\mathrm{Y}}$ is the prediction of the fractional graph and Y^* is the labeled value. β is the balance factor between positive and negative samples as in Equation (7):

$$\beta = 1 - \frac{\sum y^* \in \mathrm{Y}^* * y^*}{|\mathrm{Y}^*|} \quad (7)$$

L_g geometry loss is divided into two parts, one part is IoU loss, and one part is rotation angle loss as in Equations (8) and (9):

$$L_{AABB} = -InIoU(\hat{R} \cap \mathrm{R}^*) = -\ln \frac{\hat{R} \cap \mathrm{R}^*}{\hat{R} \cup \mathrm{R}^*} \quad (8)$$

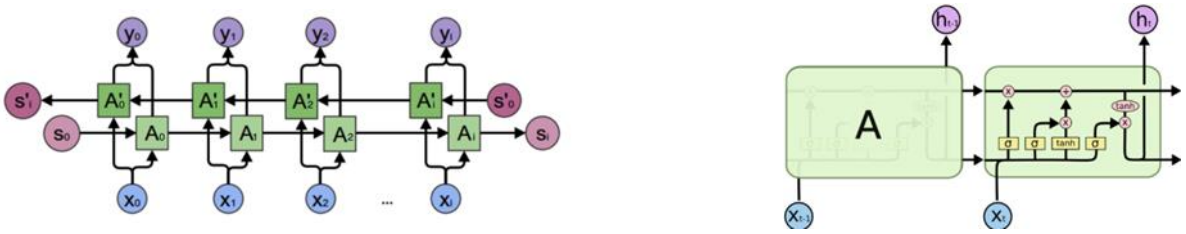
$$L_\theta(\hat{\theta}, \theta^*) = 1 - \cos(\hat{\theta} - \theta^*) \quad (9)$$

Among them, θ is the prediction of the rotation Angle, and $\overset{*}{\theta}$ Indicates the annotation value. Finally, the overall geometric loss is the sum of the $AABB$ loss and the angular loss, the formula is as in Equation (10):

$$L_g = L_{AABB} + \lambda_\theta L_\theta \quad (10)$$

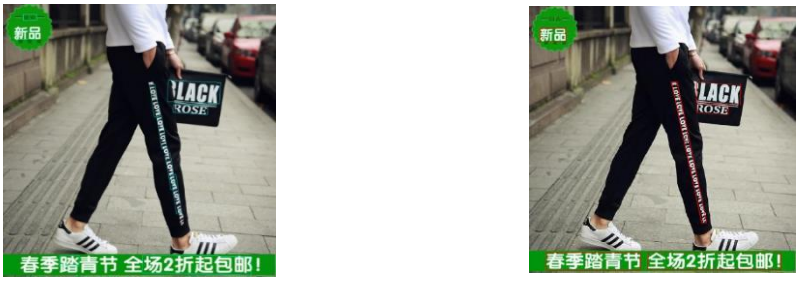
Feature Extraction of Text Sequence Based on BLSTM Network

Attention mechanism includes not only spatial–domain and channel–domain attention mechanisms mentioned above, but also the time–domain attention mechanism. This attention mechanism focuses on a special feature: a sequence feature usually called context information, which is the unique feature information of the text. Because the traditional CNN network construction cannot effectively integrate the sequence features, this paper adds the BLSTM network to the conventional feature extraction CNN algorithm [24]. The BLSTM network construction is shown in Figure 4a, where x_n, y_n and s_n represent the input sequence, output sequence, and sequence characteristics at different times, respectively. The construction of memory cell A in BLSTM is shown in Figure 4b, which is similar to the LSTM network. It is a kind of special construction with three gates: the forgetting gate, the input gate, and the output gate. With these three gates, LSTM can more effectively determine which information is forgotten or retained. LSTM has the ability of forwarding propagation and it can easily extract and output long-distance text features, so the LSTM network can effectively solve the learning of long sequence information. The BLSTM network construction is a combination of two LSTM network constructions in different directions. Context information can be integrated well by combining the two different propagation directions. The improved CTPN text detection algorithm proposed in this paper adopts the improved CNN-BLSTM network construction and combines the characteristics of a text sequence, which makes the detection results more robust to the long-term text. However, due to the existence of feature pyramid construction in the EAST algorithm, the BLSTM network cannot be directly added to the feature extraction CNN network construction. Therefore, this paper researches the specific way of adding a BLSTM network to feature fusion and output tier. It can improve the precision of the network and also obtain preferable test results. The comparison effect of text detection with or without BLSTM network construction is shown in Figure 5.



(**a**) (**b**)

Figure 4. BLSTM network construction. (**a**) Schematic diagram of BLSTM network construction. (**b**) Construction diagram of Memory cell A in the BLSTM.



(**a**) (**b**)

Figure 5. Comparison of text detection effect with or without adding BLSTM network. (**a**) Test result without BLSTM network; (**b**) Test result with BLSTM network.

The results shown in Figure 5 show that for the traditional algorithm without adding BLSTM network construction, due to the lack of sequence features, the algorithm cannot detect whether the text belongs to the same sentence. Therefore, when detecting long text, the long text will be detected into multiple short texts by mistake, and some texts will not even be detected, as shown in Figure 5a. After the addition of the BLSTM network, long texts are detected well owing to the integration of sequence features, thus contributing to the improvement of the detection accuracy, as shown in Figure 5b.

4. Experimental Results

4.1. Test Description

For the sake of testing the effectiveness of the new construction of a hybrid attention mechanism, hybrid feature pyramid, and BLSTM network for text detection in natural scenes, this paper not only tests but also analyses the hybrid feature pyramid performance and ablation performance of the new algorithm, as well as the text detection performance of the algorithm combined with a hybrid attention mechanism.

The framework used in the test is TensorFlow1.12 in the pre-training model. The test uses the improved lightweight network MobileNet-50 and uses the ADAM optimizer to drill the network from end to end. For the sake of expediting the learning, 512 $\times$ 512 frame rate images are evenly sampled from the images. After rotation and translation, the initial batch size is set to start training from 16. The learning rate of the Adam optimizer starts from 10^{-3}, and decays to one-tenth of the original after every 1000 training, until the end of training when it reaches the optimal level.

Among many data sets, the ICDAR2015 data set which includes one thousand training images and five hundred test images is especially suitable for text training and detection in natural scenes. In this paper, the ICDAR2015 data set is selected as the training set and the test set, and the test results will be compared. For the sake of verifying the effectiveness of the new construction of BLSTM and CBAM, the ablation experiment was carried out on the ICDAR2015 data set, and the precision rate P, the recall rate R, and F-value were used as assessment indexes.

4.2. Comprehensive Comparison of Several Natural Scene Text Detection Algorithms

The proposed algorithm is compared with other algorithms such as Seglink, TextSnke, and RRPN on the ICDAR2015 dataset. Meanwhile, this paper selects the video frame image with natural scene text in the video stream to test the algorithm and the test results are shown in Figures 6–8. In these figures, it can be observed that the result of the algorithm based on the optimization of the hybrid feature pyramid has a better performance than that of the traditional EAST arithmetic, because the former can attract more attention to the small acceptance domain so it has a better detection effect on the text area of the small acceptance domain. However, whether it is slanted text, slanted perspective text, long sequences text, or complex background text, the improved algorithm in this paper can achieve a better text detection effect than the other two algorithms. Especially for small acceptance domain text, the text detection effect of the proposed algorithm is particularly outstanding.

(**a**) (**b**) (**c**) (**d**)

Figure 6. Comparison of the detection effect of slanted text. (**a**) Slanted original text; (**b**) EAST algorithm; (**c**) Hybrid feature pyramid algorithm; (**d**) The algorithm of this paper.

(**a**) (**b**) (**c**) (**d**)

Figure 7. Comparison of the detection effect of long sequences text. (**a**) Original long text; (**b**) EAST algorithm; (**c**) Hybrid feature pyramid algorithm; (**d**) The algorithm of this paper.

(**a**) (**b**) (**c**) (**d**)

Figure 8. Comparison of the detection effect on complex natural background text. (**a**) Original complex background text; (**b**) EAST algorithm; (**c**) Hybrid feature pyramid algorithm; (**d**) The algorithm of this paper.

Table 1 lists the accuracy P, the recall rate R, and the F-value performance of the CTPN algorithm, the Seglink algorithm, several improved East algorithms proposed in other literature, and the improved algorithm proposed in this paper in text detection. The results show that the F-value of the proposed algorithm can reach 83%, and the recall rate R can reach 86%, which is higher than other text detection algorithms such as Seglink, TextSnke, and RRPN. The accuracy rate P of the improved algorithm proposed in this paper reaches 80%, which is higher than most text detection algorithms, but slightly lower than TextSnke and RRPN algorithms. As shown in Table 1, the improved algorithm proposed in this text

has obvious advantages in text detection. Compared with the original EAST algorithm, the recall rate R and the accuracy rate P of the improved construction are improved by 8% and 6%, respectively, and the F-value is improved by 7%. Therefore, the proposed optimization construction has obvious performance advantages compared with other improved algorithms and constructions.

Table 1. Comparison results of R, P, and F-value evaluation indexes of several algorithms.

Algorithm Name	R (%)	P (%)	F (%)
EAST + VGG16	72.8	80.5	75.4
Seglink [25]	76.8	80.5	75
CTPN + VGG16 [26]	51.6	74.2	60.9
EAST + resnet50 + BLSTM [27]	78.07	85.10	81.64
EAST + PVANET2 × MS [28]	77.23	84.64	80.77
TextSnke [29]	83.20	73.90	78.72
RRPN [30]	86.00	70.00	77.00
The algorithm in this paper	80.02	86.04	82.93

5. Conclusions

An improved EAST algorithm for text detection is proposed in this paper. The complications of text localization and inaccurate detection and high calculation load in the traditional network construction can be handled by adopting a hybrid attention mechanism method and introducing a lightweight feature extraction neural network construction. The method in this paper can reduce the computation, deepen the number of tiers of the network, and improve the accuracy of the network. Meanwhile, the improved hybrid attention mechanism is combined with the hybrid pyramid model to solve the problem of deficiency attention to small acceptance domain text and insufficient feature extraction in the EAST algorithm. Furthermore, the BLSTM network construction is also added after the feature fusion to effectively integrate the sequence features of the text region and make the text detection more robust. The test results demonstrate that the improved EAST algorithm presented in this paper has better performance than other existing ways of detecting text regions in natural scenes and complex scenes, and significantly reduces the number of parameters and calculations.

Therefore, the improved algorithm proposed in this paper is indeed a good algorithm. However, this study has three limitations. First, in the process of testing the algorithm, this study has not found a better training method for the improved algorithm, and the training method adopted in this paper is still based on the traditional network model, although this training method can also have a certain effect on the accuracy of text recognition under complex backgrounds. However, if we can find a training model that is more compatible with the lightweight network, it will help to further improve the recognition accuracy and improve the problems of insufficient attention to the text in small receptive fields and insufficient feature extraction. Second, through a large number of experimental tests, it is concluded that the EAST algorithm proposed in this paper, compared with other algorithms in terms of recognition accuracy, improved and can reduce the amount of calculation. However, the improved algorithm computational complexity of the theory of quantitative analysis model is constructed. In the future, we will further build the calculation theory of quantitative analysis model and verify its effectiveness. Finally, in the experimental test of this paper, a special data set for text detection and recognition (ICDAR2015) is used. Although this data set is widely applicable to the training and detection of conventional text, it also has a shortage of a single background scene. Therefore, because of the research field of complex background text recognition in this paper, it is necessary to collect and sort out more complex scene text data sets, to provide more reliable and richer model training resources for text detection and recognition in complex scenes.

Author Contributions: Conceptualization, M.L.; methodology, M.L. and B.L.; software, B.L.; validation, B.L. and W.Z.; formal analysis, M.L.; investigation, M.L. and B.L.; resources, M.L. and B.L.; data curation, B.L.; writing—original draft preparation, M.L. and B.L.; writing-review and editing, M.L. and B.L.; visualization, B.L. and W.Z.; supervision, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Short Name	**Full Name**
CBAM	Convolution Block Attention Module
EAST	Efficient and Accurate Scene Text
NMS	non-maximum suppression
FCN	full convolution network
LNMS	local perception non-maximum suppression
SE	squeeze-and-excitation
HS	h-swish
CAM	Channel Attention Module
SAM	Spatial Attention Module
BLSTM	bidirectional long short-term memory
ICDAR	International Conference on Document Analysis and Recognition
Faster-RCNN	Faster-Region-Convolutional Neural Network
CNN	Convolutional Neural Network
SSD	Single Shot MultiBoxDetector
CTPN	Connectionist Text Proposal Network
VGG	Visual Geometry Group
ReLU	Rectified Linear Unit
ResNet	Deep residual network
IoU	Intersection over Union
LSTM	Long Short Term Memory
ADAM	Adaptive Moment
RRPN	Rotation Region Proposal Network

Appendix A

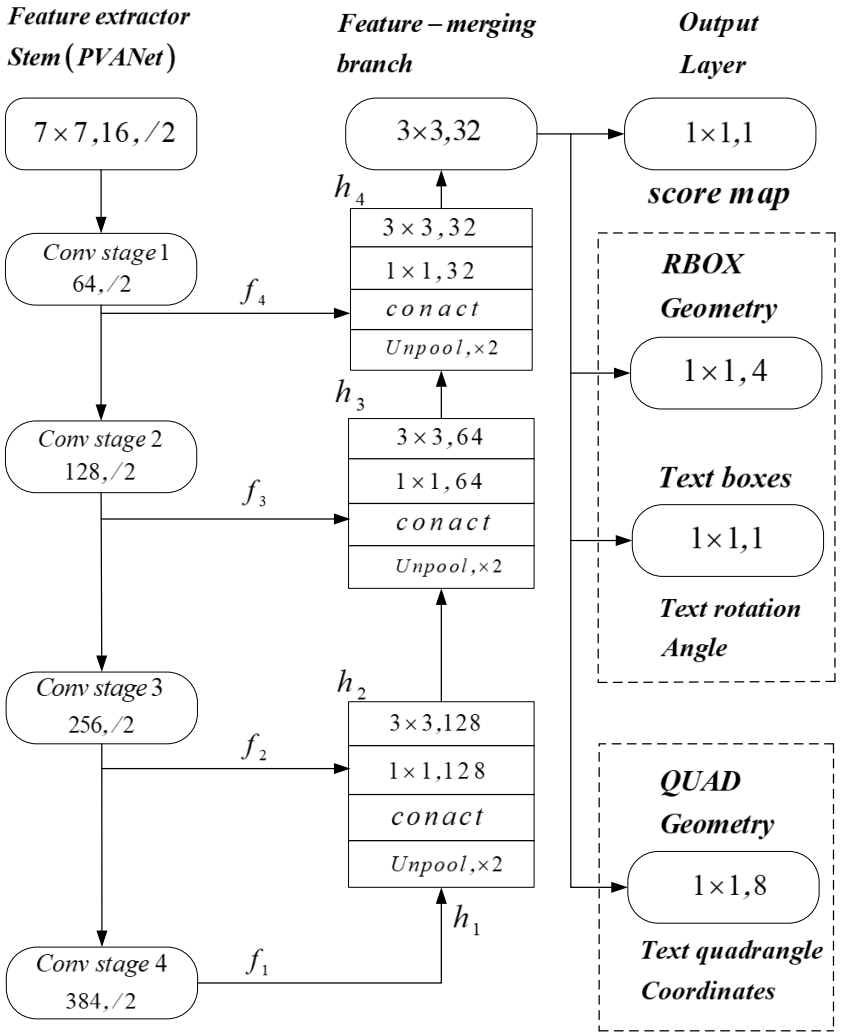


Figure A1. School diagram of FCN network construction.

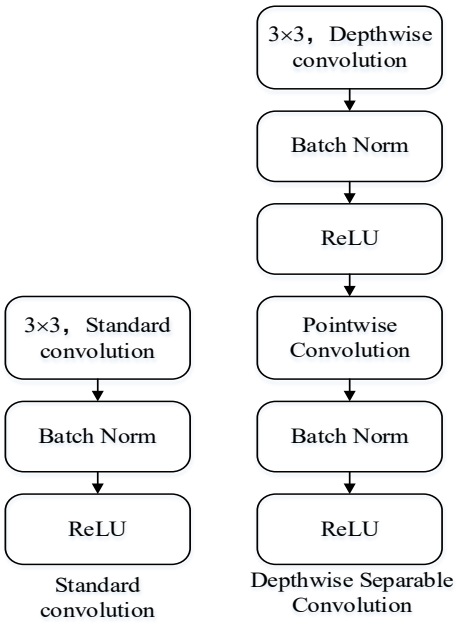


Figure A2. Standard separable convolution and Depth wise separable convolution.

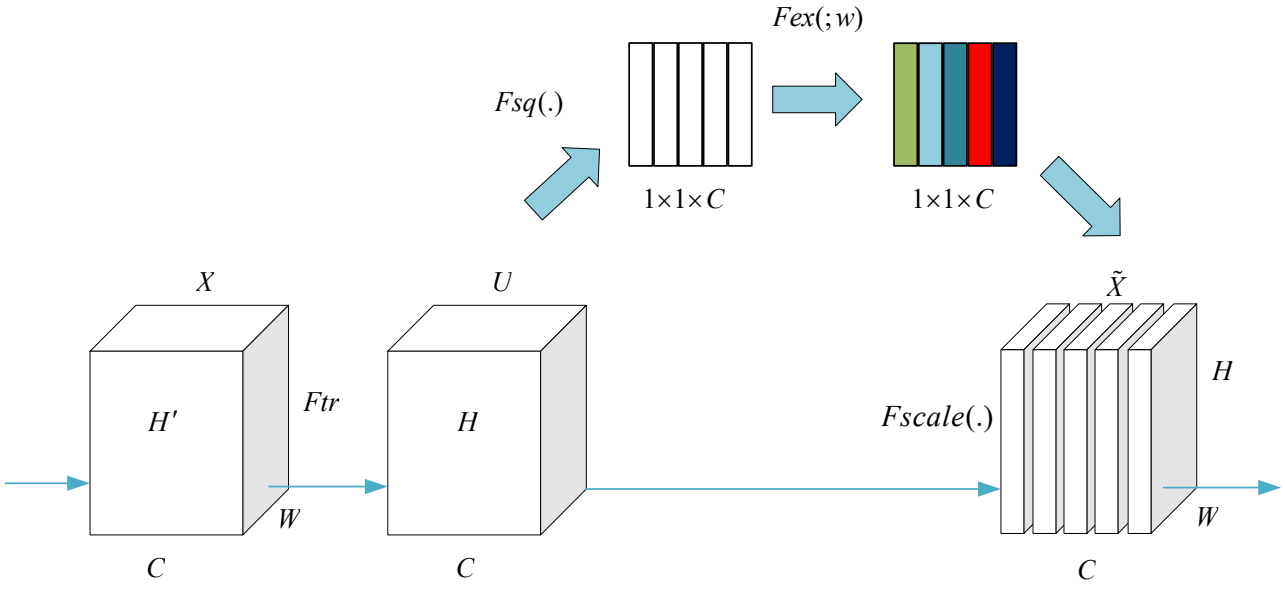


Figure A3. Schematic diagram of SE module construction.

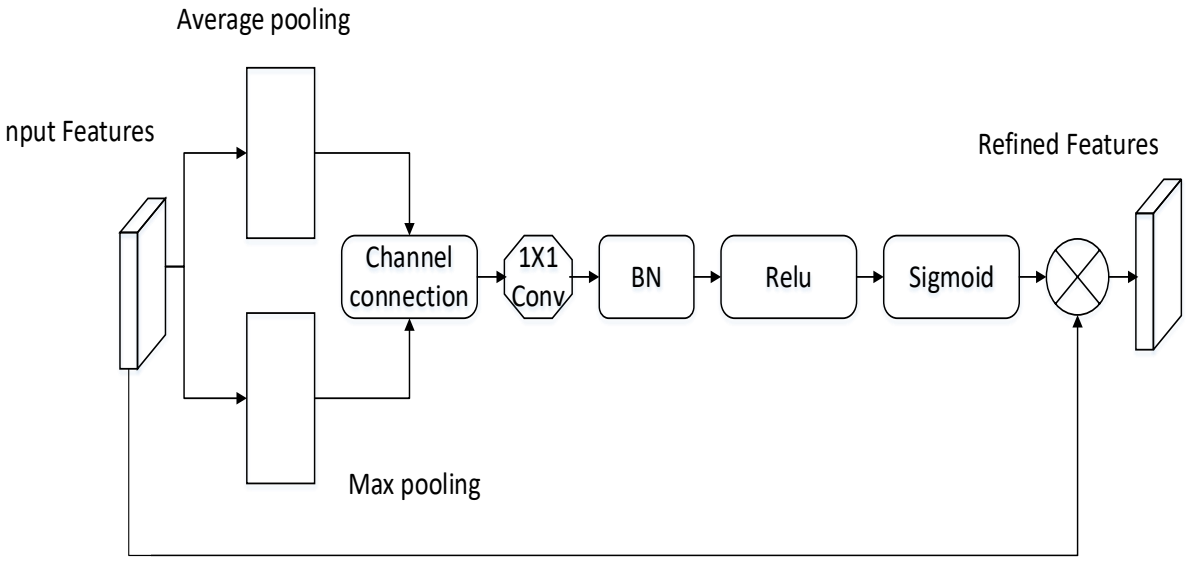


Figure A4. Schematic diagram of spatial attention mechanism.

References

1. Ren, S.; He, K.; Girshick, R.; Jian, S. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
2. Hwang, Y.J.; Lee, J.G.; Moon, U.C.; Park, H.H. SSD-TSEFFM: New SSD Using Trident Feature and Squeeze and Extraction Feature Fusion. *Sensors* **2020**, *20*, 3630. [CrossRef] [PubMed]
3. Li, X.B.; Chen, L. Face detection in natural scenes based on improved Faster-RCNN. *J. Comput. Eng.* **2021**, *47*, 210–216.
4. Yun, N.; Lee, S. Analysis of effectiveness of tsunami evacuation principles in the 2011 Great East Japan tsunami by using text mining. *Multimed. Tools Appl.* **2016**, *75*, 12955–12966. [CrossRef]
5. Huang, X. Automatic video scene text detection based on saliency edge map. *Multimed. Tools Appl.* **2019**, *78*, 34819–34838. [CrossRef]
6. Zhou, B.; Liu, Y.Q.; Min, Z.; Jin, Y.J.; Ma, S.P. Retrieval effectiveness analysis for anchor texts. *J. Softw.* **2011**, *22*, 1714–1724. [CrossRef]
7. Sun, J.; He, X.; Wu, M.; Wu, X.; Lu, B. Detection of tomato organs based on convolutional neural network under the overlap and occlusion backgrounds. *Mach. Vis. Appl.* **2020**, *31*, 15–22. [CrossRef]
8. Chang, L.; Zhang, S.; Du, H.; You, Z.; Wang, S. Position-aware lightweight object detectors with depthwise separable convolutions. *J. Real-Time Image Processing* **2020**, *34*, 857–871. [CrossRef]
9. Pang, Y.; Zhang, Y.J.; Lin, L.Z.; Cai, Y.Q. Improved scene text detection in any direction based on EAST. *J. Chongqing Univ. Posts Telecommun. Nat. Sci. Ed.* **2021**, *33*, 868–876.
10. Yang, B.; Du, X.Y. Natural scene text localization algorithm based on improved EAST. *J. Comput. Eng. Appl.* **2019**, *55*, 161–165.
11. Qiao, T.; Su, H.S.; Liu, G.H.; Wang, M. Object detection algorithm based on improved feature extraction network. *J. Adv. Lasers Optoelectron.* **2019**, *56*, 134–139.
12. Saha, S.; Chakraborty, N.; Kundu, S.; Paul, S.; Mollah, A.F.; Basu, S.; Sarkar, R. Multi-lingual scene text detection and language identification. *Pattern Recognit. Lett.* **2020**, *138*, 16–22. [CrossRef]
13. Wang, X.; Xue, F.; Wang, W.; Liu, A. A network model of speaker identification with new feature extraction ways and asymmetric BLSTM. *Neurocomputing* **2020**, *403*, 167–181. [CrossRef]
14. Li, X.Q.; Liu, J.; Zhang, S.W.; Zhang, G.X.; Zheng, Y. Single shot multi-oriented text detection based on local and non-local features. *Int. J. Doc. Anal. Recognit.* **2020**, *23*, 241–252. [CrossRef]
15. Naiemi, F.; Ghods, V.; Khalesi, H. Hassan Khalesi Scene text detection using enhanced Extremal region and convolutional neural network. *Multimed. Tools Appl. Int. J.* **2020**, *79*, 27137–27159. [CrossRef]
16. Arunkumar, P.; Shantharajah, S.P.; Geetha, M. Improved canny detection construction for processing and segmenting text from the images. *Clust. Comput.* **2018**, *22*, 7015–7021. [CrossRef]
17. Wang, D.F.; Chen, C.B.; Ma, T.L.; Li, C.H.; Miao, C.Y. YOLOv3 pedestrian detection algorithm based on depth-separable convolution. *J. Comput. Appl. Softw.* **2020**, *37*, 218–223.
18. Liu, K.; Pan, G.Y.; Zheng, D.G.; Gu, J.J.; Meng, C.Y. An aviation forensics target detection algorithm based on RetinaNet and SE fusion. *J. Mod. Def. Technol.* **2022**, *50*, 25–32.
19. Zhao, L.L.; Wang, X.Y.; Zang, Y.; Zhang, M.Y. Research on vehicle target detection technology based on YOLOv5s and SENet. *J. J. Graphics.* **2022**, *4038*, 1–8.

20. Jlab, C.; Qza, B.; Yuan, Y.D.; Hai, S.E.; Bo, D. SemiText: Scene text detection with semi-supervised learning. *Neurocomputing* **2020**, *407*, 343–353.
21. Basavaraju, H.T.; Aradhya, V.; Pavithra, M.S.; Guru, D.S.; Bhateja, V. Arbitrary oriented multilingual text detection and segmentation using level set and Gaussian mixture model. *Evol. Intell.* **2022**, *14*, 881–894. [CrossRef]
22. Guo, Q.F.; Liu, L.; Zhang, C.; Xu, W.J.; Jing, W.F. Multi-scale feature fusion network based on feature pyramid. *J. Eng. Math.* **2020**, *37*, 521–530.
23. Xu, C.Q.; Hong, X.H. Feature pyramid target detection network based on function preservation. *J. Pattern Recognit. Artif. Intell.* **2020**, *33*, 507–517.
24. Guo, C.; Qiu, X.H. Improved EAST text detection algorithm based on BLSTM network. *J. Comput. Technol. Dev.* **2020**, *30*, 21–24.
25. Liu, J.; Feng, K.; Pan, J.Z.; Deng, J.; Wang, L. MESD: Multi-Modal Web Rumor Detection Way. *J. Comput. Res. Dev.* **2020**, *57*, 2328–2336.
26. Chakraborty, N.; Chatterjee, A.; Singh, P.K.; Mollah, A.F.; Sarkar, R. Application of daisy descriptor for language identification in the wild. *Multimed. Tools Appl.* **2020**, *80*, 323–344. [CrossRef]
27. Bai, Z.C.; Li, Q.; Chen, P.; Guo, L.Q. Text detection in natural scenes: A literature review. *Chin. J. Eng.* **2020**, *42*, 1433–1448.
28. Lu, L.; Wu, D.; Wu, T.; Huang, F.; Yi, Y. Anchor-free multi-orientation text detection in natural scene images. *Appl. Intell.* **2020**, *50*, 3623–3637. [CrossRef]
29. Shiravale, S.S.; Jayadevan, R.; Sannakki, S.S. Devanagari Text Detection from Natural Scene Images. *Int. J. Comput. Vis. Image Process. IJCVIP* **2020**, *10*, 44–59. [CrossRef]
30. Qin, X.; Chu, X.; Yuan, C.; Zhao, Q. Necklace: A novel long text detection model. *J. Eng.* **2020**, *2020*, 416. [CrossRef]

electronics

MDPI

Article

Modulation Recognition of Digital Signal Using Graph Feature and Improved K-Means

Guodong Li [1], Xvan Qin [1], He Liu [2], Kaiyuan Jiang [1,*] and Aili Wang [1,*]

[1] Heilongjiang Province Key Laboratory of Laser Spectroscopy Technology and Application, Harbin University of Science and Technology, Harbin 150080, China
[2] Comprehensive Data Processing Center, Heilongjiang Electric Power Company Limited, Harbin 150010, China
* Correspondence: jiangkaiyuan@hrbust.edu.cn (K.J.); aili925@hrbust.edu.cn (A.W.)

Abstract: Automatic modulation recognition (AMR) has been wildly used in both military and civilian fields. Since the recognition of digital signal at low signal-to-noise (SNR) ratio is difficult and complex, in this paper, a clustering analysis algorithm is proposed for its recognition. Firstly, the digital signal constellation is extracted from the received waveform (digital signal + noise) by using the orthogonal decomposition and then, it is denoised by using an algorithm referred to as auto density-based spatial clustering technique in noise (ADBSCAN). The combination of density peak clustering (DPC) algorithm and improved K-means clustering is used to extract the constellation's graph features, the eigenvalues are input into cascade support vector machine (SVM) multi-classifiers, and the signal modulation mode is obtained. BPSK, QPSK, 8PSK, 16QAM and 32QAM five kinds of digital signals are trained and classified by our proposed method. Compared with the classical machine learning algorithm, the proposed algorithm has higher recognition accuracy at low SNR (less than 4dB), which confirmed that the proposed modulation recognition method is effective in noncooperation communication systems.

Keywords: modulation recognition; modulation classification; support vector machine; K-means

1. Introduction

During signal transmission, according to the relationship between the sender and the receiver, the communication process can be divided into two modes: cooperative and non-cooperative. In the cooperation mode, the relevant parameters of the signal are known to both the receiving and sending terminals; in the noncooperative mode, the parameters related to the signal transmission at the receiving end are only partially known or completely unknown. The modulation recognition technology mainly works in the non-cooperative mode to identify the modulation mode and its parameters of the signal to be detected, such as carrier frequency, symbol rate, etc.

In the field of military communications, information warfare has become a key factor in the victory of a war. Mastering more information in a war often means taking a greater initiative. Modulation recognition technology, originated from military communication, is the early foundation and necessary condition for electronic reconnaissance, target detection, communication warning, monitoring and other activities. In military information warfare, modulation recognition technology can be used to intercept unknown wireless signals, decode them, extract information, and obtain the communication content of the target; The jamming parameters can also be set according to the identified modulation information to generate corresponding jamming signals, so as to realize the jamming of enemy wireless signals. Therefore, an efficient, fast and accurate modulation recognition system is of great significance for occupying the information commanding heights in modern information warfare.

Signal modulation recognition technology mainly refers to the technology of distinguishing the modulation mode used by the modulation signal and estimating the modulation related parameters in the absence of relevant prior information [1]. Modulation recognition technology has always been an important basic technology in military and civil fields such as electronic reconnaissance and radio monitoring. The traditional modulation recognition technology mainly uses the signal time domain waveform and its transform domain characteristics to identify and judge. The recognition process depends on the subjective experience of algorithm designers, which has low work efficiency, high labor cost and poor universality of the algorithm. Nowadays, with the improvement of hardware computing power and the wide application of artificial intelligence algorithm theory, artificial intelligence related algorithms are applied to the field of signal modulation recognition with high recognition rate, high speed and good applicability, which will have a very broad development prospect and application space in military and civil fields.

Modulation recognition is the process of determining the modulation mode of digital signal by means of signal correlation feature extraction. It is not only a part of modulation recognition, but also the basis of subsequent operations such as signal demodulation, interpretation and information extraction. Modulation recognition is mainly divided into the traditional methods (such as statistical mode, decision theory, neural network, wavelet transform and so on) and deep learning methods based on convolutional natural network [2]. Liu et al. combines the traditional method with the feature of power spectrum sidelobe attenuation to recognize the 5G signal [3]. Izhou Gep et al. proposed a method which combines the method of cyclic spectrum with high-order cumulant to perform the modulation recognition of UWA communication to improve rate recognition of underwater signals at low SNR [4].

Most traditional method based on artificial designed features to recognition, the ability of this kind of algorithm is limited. The modulation recognition algorithm based on deep learning uses the ability of network model to extract the characteristics of graphic data, recognizes the digital signal through autonomous learning, and can also be recognized under the condition of less a priori information. At the same time, it can reduce the dependence on the subjective experience of algorithm designers. Ya Liu et al. first tried to use graph convolution neural network to solve the problem of high algorithm complexity, which can effectively reduce the algorithm complexity of deep learning network [5]. Dechun Sun et al. proposed a modulation recognition algorithm based on VGG convolutional neural network, aiming at the difficulty of feature extraction of traditional algorithms under low SNR. This algorithm makes up for the shortage of manual feature extraction, converts the sampled data of communication signal into gray image, and trains the model and classifies it through VGG network [6]. In view of the characteristics that traditional recognition algorithms mostly rely on manual design, which leads to the application limitations of the algorithm, Zhang Tingting et al. proposed a modulation recognition algorithm based on the Gate Recurrent Unit (GRU), which uses a multi-layer bidirectional GRU network structure to improve the feature representation ability [7]. Ke Bu et al. introduced Adversarial Transfer Learning Architecture (ATLA) into deep learning, combined adversarial training with knowledge transfer, and reduced the sample size of training set [8]. In order to solve the problem of low recognition rate under low SNR, Fugang Liu et al. optimized on the basis of Long Short-Term Memory (LSTM) networks [9]. This method first extracts the characteristics of the signal, such as advanced cumulants, and then sends them to the parallel network composed of convolutional neural network and GRU for recognition. Zhang et al. proposed a framework called Attention Relation Network (ARN) which introduced channel and spatial attention, respectively to learn a more effective feature representation of support samples [10]. Wu et al. pointed out that most existing automatic modulation recognition (AMR) algorithm highly depended on the feature of data, so they introduced the LSTM as the classifier into neural network to improve the robustness [11]. Yang et al. proposed an algorithm based on AlexNet to increase the rate of recognition [12]. High-order Quadrature Amplitude Modulation (QAM) has problem that the performance

severely degraded in multipath fading channels, Xuefei et al. proposed an algorithm that combines a blind equalization module and ResNet [13].

In addition, machine learning has a good performance in modulation recognition. Because of the characters of the algorithm itself, deep learning also has some problems such as having a complex learning process in neural networks, learning results cannot be corrected, training is time-consuming, model correctness verification is complex and cumbersome. Chou et al. used K-means cluster algorithm to contain the feature of the signal constellation to reduce the complexity of the algorithm [14]. H. TAYAKOUT et al. used a simplified Distributed Space-Time Block Coding (D-STBC) scheme for a three-node cooperative network system incorporating, which addresses the problem of semi-blind automatic digital modulation recognition for wireless relaying network [15]. Liu Wang et al. proposed a clustering algorithm to automatically choose the center of the constellation according to the number of center points [16]. Punith et al. proposed a new algorithm to distinguish between 6 digital modulation schemes by using adaptive threshold methods [17].

Aiming at the low recognition accuracy of traditional modulation recognition methods under low SNR conditions, a modulation recognition algorithm model based on the characteristics of digital signal constellation is proposed. The algorithm model first converts the signal into the form of constellation diagram, uses density clustering algorithm for noise reduction, and inputs the features extracted by KADPC algorithm into multi-level SVM classifier to obtain modulation recognition results. This method reduces the complexity of feature extraction and the cost of classification. Compared with traditional modulation recognition methods, this method can improve the recognition performance of several common types of digital modulation signals. The rest of this paper is organized as follows. Section 2 briefly introduces the proposed methods relevant and improvements in this paper. The experimental results and analysis are given in Section 3. Section 4 summarizes the conclusions and future work.

2. Algorithm Design

The proposed algorithm including three parts: preprocessing, feature extraction, and modulation type recognition. The block diagram of the modulation recognition algorithm is shown in Figure 1. The preprocessing part transforms the signal to be detected into the form of constellation, uses ADBSCAN to reduce noise, and obtains the noise reduced constellation. In the feature extraction part, K-means is improved by using density peak clustering. Clustering is performed on the constellation after noise reduction and the eigenvalues are calculated to input into the SVM classifier for classification. In the classification part, multiple SVM cascades are used to form a multi-level classifier. The extracted eigenvalues are classified to obtain the recognition results of digital signal modulation.

2.1. Denoising Processing Based on ADBSCAN

In digital communication, constellation diagram is a more visual and direct method to represent the signal to be detected, which is generated by orthogonal modulation The received signal is divided into two channels after receiving the signal and multiplies by two carries with a phase difference $\pi/2$, then goes through a low-pass filter. Then, two separate components the in-phase component and the quadrature component are obtained. Constellation diagram consists of two coordinate axes, the horizontal X-axis is relevant to the in-phase component, the vertical Y-axis is relevant to the quadrature component, each point in the diagram contains four information. The projection in X-axis defines the peak amplitude of in-phase component, the projection in Y-axis defines the peak amplitude of quadrature component, the length of the line from the point to the origin is the peak amplitude of the signal element, the angle between the line and the X-axis is the phase of the signal element. Because of the noise interference, it is hard to recognize the modulation of the constellation diagram, so ADBSCAN clustering algorithm is used into reducing noise in preprocessing parts.

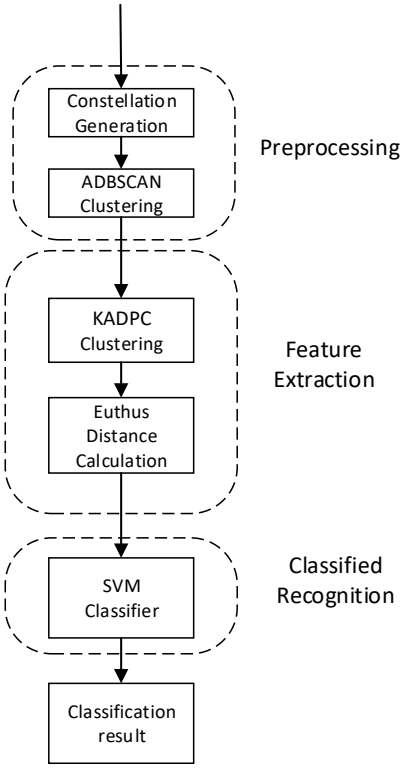


Figure 1. The block diagram of the proposed digital modulation recognition system.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) clustering algorithm, a kind of fast-clustering algorithm [17], has a fast and good affection in processing the noise [18]. The algorithm uses threshold density as the classify standard based on a fact that one clustering algorithm can be determined by any core object of the sample [19], meanwhile the number of the clusters can be determined by the tightness of the sample distribution. Through classifying connected samples into one cluster, a cluster category is obtained. The final results of all categories are obtained by dividing all closed connected samples into different categories.

Consuming the sample set $D = (x_1, x_2, \cdots, x_m)$, ε describes the neighborhood distance threshold of a sample, *Minpts* describes the threshold of the number of samples in the neighborhood where the distance of a sample is ε, the parameter definition of ADBSCAN clustering algorithm is defined as follows:

1. ε neighborhood: For $x_j \in D$, the neighborhood whose distance from x_j is not greater than ε is defined as $\left|N_\varepsilon(x_j)\right|$:

$$N_\varepsilon(x_j) = \left\{x_i \in D \middle| dist(x_i, x_j) \leq \varepsilon\right\} \tag{1}$$

2. Core object: For $x_j \in D$, if the $N_\varepsilon(x_j)$ corresponding to the neighborhood contains at least *Minpts* samples, then the core object is defined as follows:

$$N_\varepsilon(x_j)\big| \geq Minpts \tag{2}$$

3. Directly density-reachable: If x_i is located in the ε neighborhood of x_j, and x_j is the core object, then x_i is said to be direct by the density of x_j.

4. Density-reachable: For x_i and x_j, if there are sample sequences $p_1, p_2, \cdots, p_T$, satisfying $p_1 = x_i$, $p_T = x_j$, and p_{t+1} is directly reached by the density of p_t, then x_j is said to be reachable by the density of x_i. In other words, the density can be reached to meet the transitivity. At this time, the transmitted samples $p_1, p_2, \cdots, p_{T-1}$ in the sequence are all core objects, because only core objects can directly reach the density of other samples. Note

that the density is reachable and does not satisfy the symmetry, this can be derived from the direct asymmetry of the density.

5. Density-connected: For x_i and x_j, if there is a core constellation sample x_k, so that both x_i and x_j are reachable by the density of x_k, then x_i and x_j are called density connected. Note that the density connection relationship satisfies symmetry.

According to the principle of ADBSCAN clustering, the category or cluster of the largest connectable sample set is found from the density reachable relationship as the final result. If there is only one core object in an ADBSCAN cluster, all non-core object constellation points in the cluster are in the ε neighborhood of the constellation point; if there are multiple core objects, there is a density reachable relationship between any two core objects. Because ADBSCAN does not need to determine the number of clusters during clustering, it can denoise the signal constellation during digital signal preprocessing. The specific steps of ADBSCAN clustering algorithm are shown in Algorithm 1.

Algorithm 1 ADBSCAN clustering algorithm

INPUT: Constellation sample coordinate set D = $(x_1, x_2, \cdots, x_m)$
OUTPUT: Clustering results

Step1: ε and *Minpts* are automatically determined by calculating the distance between constellation sample points.

Step2: Initialize the core object collection Ω, cluster number K unvisited sample set Γ and cluster interval C.

Step3: For $j = 1, 2, \cdots m$, follow the steps below to find all core objects:

(a) Find the ε neighborhood sub-sample set $N_\varepsilon(x_j)$ of the sample x_j through the distance measurement method.

(b) If the number of samples in the sub-sample set satisfies $N_\varepsilon(x_j)| \geq Minpts$, add sample x_j to the core object sample set: $\Omega = \Omega \cup \{x_j\}$.

Step4: If the core object set $\Omega = \phi$, the algorithm ends, otherwise, go to step 5.

Step5: In the core object set Ω, randomly select a core object O, initialize the current cluster core object queue $\Omega_{cur} = \{o\}$, the category number $k = k + 1$ and the current cluster sample set $C_k = \{o\}$, update the unvisited samples set $\Gamma = \Gamma - \{o\}$.

Step6: If the current cluster core object queue $\Omega_{cur} = \phi$, the current cluster C_k has been generated, update the cluster division $C = \{C_1, C_2, \cdots, C_k\}$ and the core object set $\Omega = \Omega - C_k$ go to step 4. Otherwise, update the core object set $\Omega = \Omega - C_k$.

Step7: Take a core object o' from the current cluster core object queue Ω_{cur}, find all ε neighborhood subsample sets $N_\in(o')$ through the neighborhood distance threshold ε, let $\Delta = N_\in(o') \cap \Gamma$, update the current cluster sample set $C_k = C_k \cup \Delta$, the unvisited sample set $\Gamma = \Gamma - \Delta$, and $\Omega_{cur} = \Omega_{cur} \cup (\Delta \cap \Omega) - o'$, go to step 6.

The output result is Cluster division $C = \{C_1, C_2, \cdots, C_k\}$

2.2. Feature Extraction

In this paper, K-means combined with auto DPC (KADPC) algorithm is used for feature extraction. K-means clustering algorithm, a traditional clustering based on distance, according to the distance between the sample points and the cluster center to cluster data. As the most classical unsupervised clustering algorithm, the main purpose of K-means is to divide all the samples to K clusters, make the sample similarity between the same class is as high as possible, and the similarity between different categories is as small as possible. Points on the constellation map that are closely distributed around the cluster center can well be clustered.

DPC is a fast-clustering algorithm, based on density peaks, according to the density distribution of sample points to determine the clustering result. DPC algorithm is greatly affected by the local density, when a sample is classified incorrectly, the remaining points are greatly affected, the classification result is easy to cause errors. K-means algorithm selects the initial clustering center randomly when clustering the constellation diagram. Thus, the clustering result has greater randomness, and the number of clustering centers needs to be determined before clustering. In modulation recognition, the number of

cluster centers cannot be known in advance. Therefore, in this paper, the idea of DPC algorithm is used for improving the K-means algorithm. According to the distribution of points in the constellation diagram, find the points with the highest density as cluster centers, thereby reducing the number of iterations of K-means clustering and improving the clustering effect.

KADPC is an auto clustering algorithm combined with K-means and DPC clustering algorithm, whose parameters are defined as follows:

1. Cutoff distance d_c: let the cutoff distance gradually increase from zero, and find the cutoff distance that makes the entropy H reach the minimum

$$H = -\sum_{i=1}^{n}(\delta_i \sum_{j} e^{-(\frac{d_{ij}}{d_c})^2}) \log(\delta_i \sum_{j} e^{-(\frac{d_{ij}}{d_c})^2}) \tag{3}$$

$$d_{ij} = ||x_j - x_i|| \tag{4}$$

2. Local density p_j: Define the local density of point x_j in the constellation diagram with local density p_j as:

$$p_j = \sum_{j=1}^{k} \chi(d_{ij} - d_c) \tag{5}$$

$$\chi(x) = \begin{cases} 1 & x \leq 0 \\ 0 & x \geq 0 \end{cases} \tag{6}$$

3. High local density point distance θ_j:

$$\theta_j = \min_{j:p_j > p_i} (d_{ij}) \tag{7}$$

where d_{ij} is the distance between any two constellation point x_j and x_i, $\chi(x)$ is the decision function. The specific steps of KADPC clustering algorithm are shown in Algorithm 2.

Algorithm 2 KADPC clustering algorithm

INPUT: Sample set D = $(x_1, x_2, \cdots, x_m)$
OUTPUT: Eigenvalues T_c

Step1: Obtain the truncation distance d_c by increasing the truncation distance from zero to the distance that minimizes the entropy H.

Step2: According to the truncation distance d_c, the local density value of constellation points in the constellation diagram is calculated, and the distance between other constellation points and high local density constellation points is calculated.

Step3: Use the value of the calculated local density p_j and the distance of high local density point θ_j to make a two-dimensional decision diagram to select the cluster center of the data sample, that is, select the point with the larger p_j and θ_j as the cluster center:

Step4: Select K cluster centers and perform K-means clustering on the selected K cluster centers:

Step5: Calculate the sum of the Euclidean distances from the points in the K cluster to the centers of this cluster, find the average value of the sum of distances S_k:

$$S_k = \frac{s_1 + s_2 + s_3 + \cdots + s_k}{k}$$

Step6: Normalize S_k, and define unit Euclidean distance as the eigenvalue T_c.

In modulation recognition, the modulation method of the signal cannot be known, and the number of cluster centers cannot be determined. The number of cluster centers obtained by the existing modulation method is limited, and the number of cluster centers can be set multiple times, Classify and recognize the obtained set of feature values as the features of the signal, that is, the value of K can be set to 2, 4, 8, 16, 32, 64, and the corresponding feature values obtained are T2, T4, T8, T16, T32, T64.

2.3. Modulation Recognition Based on Multi-SVM

The basic SVM model is a linear classifier with the largest interval defined in the feature space. Since the use of the kernel function method overcomes the curse of dimensionality and the problem of non-linear separability, there is no increase in computational complexity when mapping to a high-dimensional space. The SVM classification algorithm was originally only used to solve two classification problem, and lacked the ability to deal with multi-classification problems. Due to the wide variety of modulation methods, in order to reduce the number of SVM classifiers, one-versus-rest (OVR) SVM is used in this paper. Take BPSK, 4PSK, 8PSK and 16PSK as classification targets to realize multi classification, three SVM classifiers are used to form a multi-classifier as shown in Figure 2, for each classifier is composed of the eigenvalues of the same signal under different SNR.

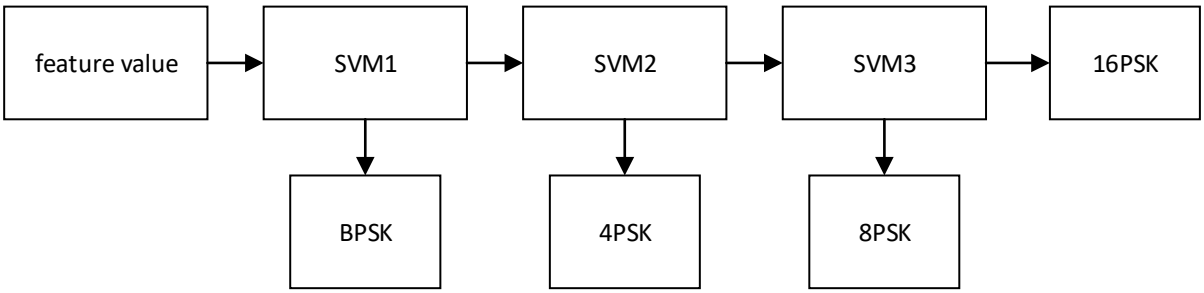


Figure 2. The block diagram of multi-classifier system using SVM.

In the data training process, the training of each classifier is composed of the eigenvalues of the same signal under different SNRs, thus reducing the amount of parameter training, as shown in Equation (8).

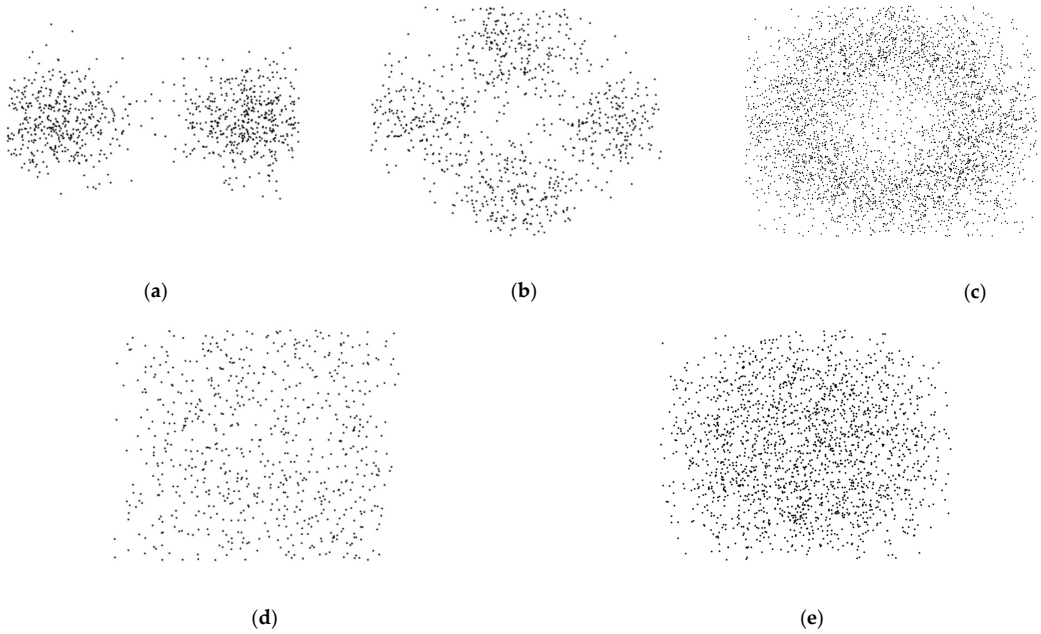
$$\left[T_{2(jdb)}, T_{4(jdb)}, T_{16(jdb)}, T_{32(jdb)}, T_{64(jdb)}\right]_{j=-2,0,2,5,10} \tag{8}$$

3. Experimental Results and Analysis

In order to verify the feasibility of the algorithm, MATLAB is used in this paper to simulate Phase Shift Keying (PSK) and Quadrature Amplitude Modulation (QAM) signal to obtain BPSK, QPSK, 8PSK, 16QAM, 32QAM at different SNRs. Figure 3 are the constellation diagrams of the five signals at SNR = 6 dB. The symbol rate of each signal is 1200 B/s, the number of symbols is 4000, the carrier frequency is 4800 Hz and the sampling frequency is 16 $\times$ 4800 Hz. Gaussian white noise is added to simulate the transmission of all signals on the Gaussian channel.

In this paper, ADBSCAN is used to reduce noise to increase the rate of recognition. Because there are too many constellations generated, in order to more intuitively observe the results of noise reduction, the number of symbols sent by simulation is set to 100. Figure 4 shows that the points in the constellation diagram are divided into two clusters, which are represented by red and blue colors, and the remaining points are labeled as noise points, which will be deleted in the subsequent processing.

This paper selects several classical algorithms of machine learning and deep learning for comparison, including backward propagation (BP) neural network, fuzzy K-means (FK), random forest (RF), K-nearest neighbor (KNN) and SVM algorithm. Under each SNR, there are 21,000 training samples and 9045 test samples. The recognition rate of BPSK modulated signal under different methods is shown in Figure 5a. It can be seen that the overall recognition rate curve of KADPC algorithm model is better than other comparison algorithms. The recognition rate curve is stable and close to 100% when SNR = 2 dB. KADPC algorithm adds noise reduction and preprocessing. At the same time, under ideal conditions, BPSK signal has only two clustering centers. When KADPC algorithm is used for feature extraction, the difference of eigenvalues is obvious. Therefore, KADPC algorithm is significantly improved compared with other algorithms.



(**a**) (**b**) (**c**)

(**d**) (**e**)

Figure 3. Signal Constellation Chart (SNR = 6 dB) (**a**) BPSK; (**b**) QPSK;(**c**) 8PSK; (**d**) 16QAM; (**e**) 32QAM.

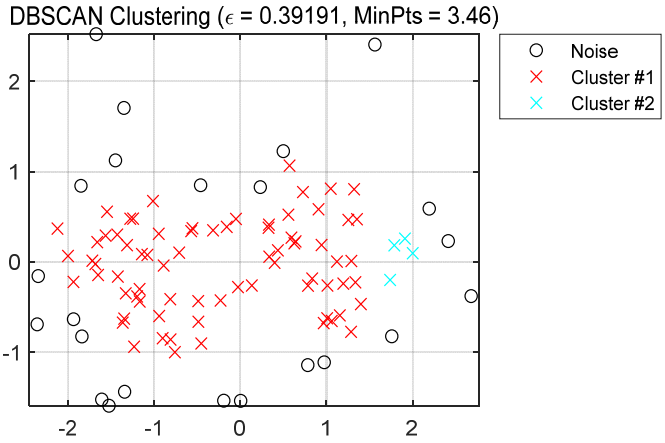
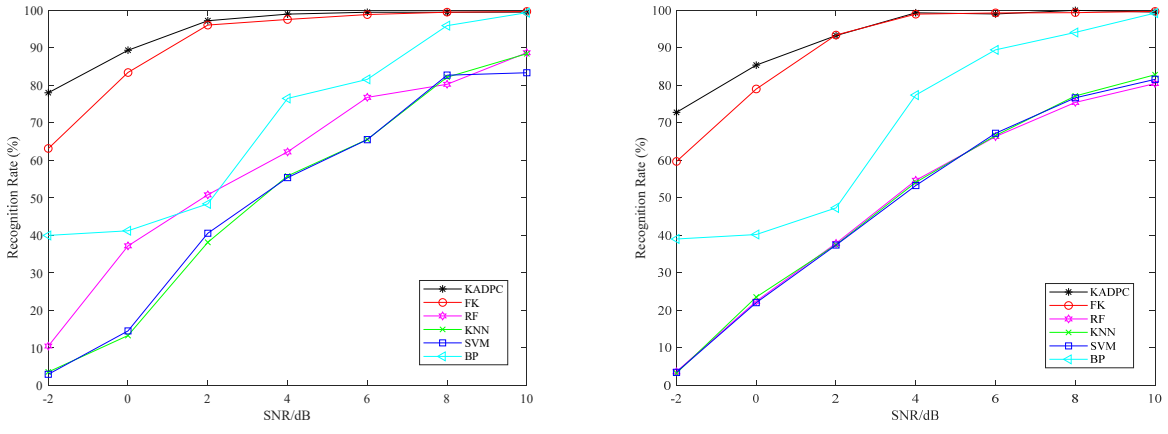


Figure 4. The noise reduction visualization result of BPSK at SNR = 0 dB.



(**a**) (**b**)

Figure 5. *Cont.*

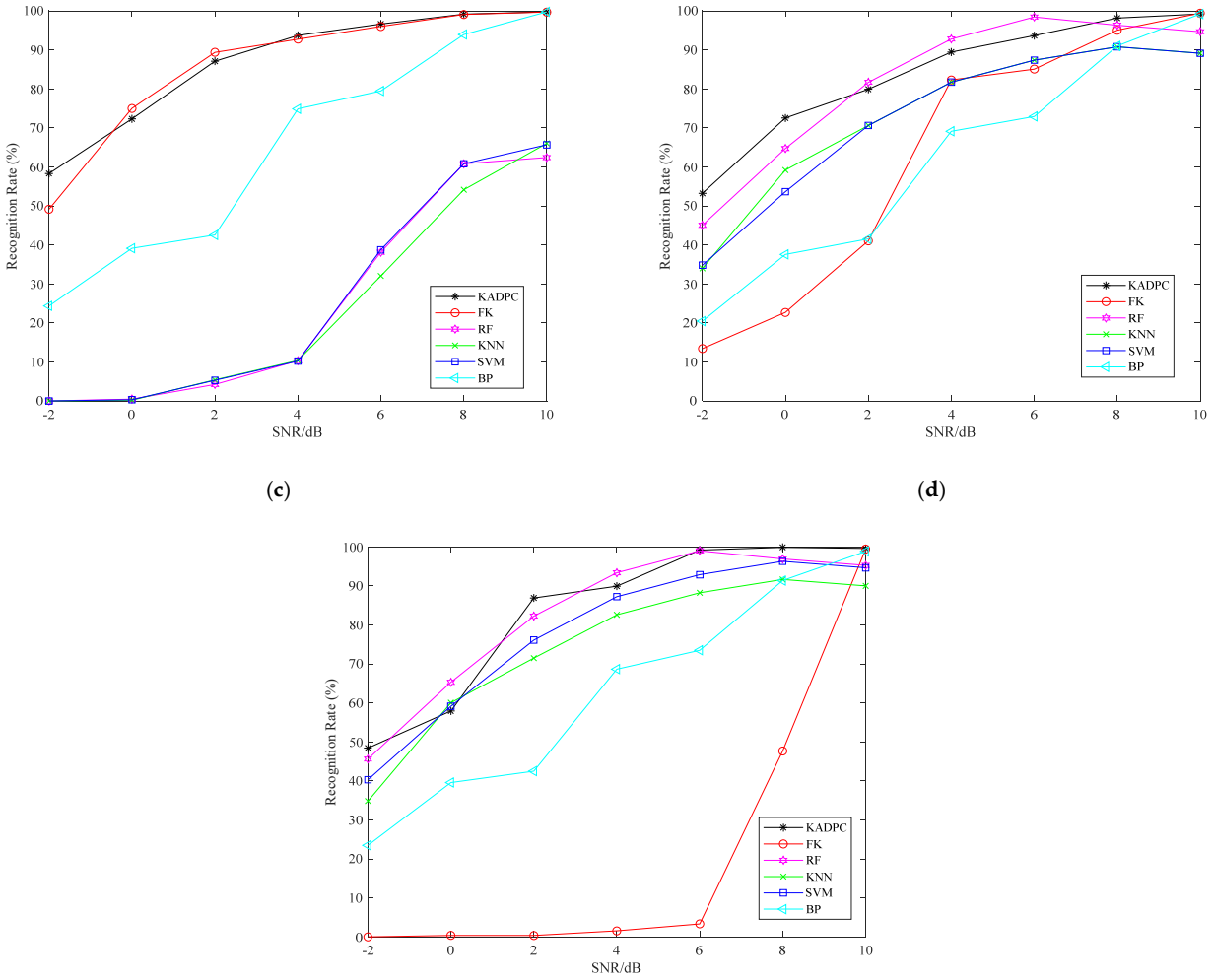


Figure 5. The Recognition rate of different signals. (**a**) BPSK; (**b**) QPSK; (**c**) 8PSK; (**d**) 16QAM; (**e**) 32QAM.

The simulation results of recognition accuracy of QPSK signal under different SNR are shown in Figure 5b. KADPC model is gradually stable and close to 100% when SNR = 4 dB, while the recognition rate of other algorithms is significantly lower than KADPC algorithm when SNR = 2 dB. Although it is close to that of FK when SNR = 4 dB, it is better than FK algorithm at low SNR. Under ideal conditions, the points in the constellation of QPSK signal are closely distributed around the four clustering centers. With the improvement of SNR, the number of clustering centers of FK is closer to the actual value, so the algorithm is improved significantly. However, the recognition rate of FK is lower than that of KADPC algorithm under low SNR, and the other comparison algorithms perform poorly on QPSK signal. The comprehensive results show that the recognition effect of the algorithm proposed in this chapter is better. The recognition accuracy of 32QAM signal at different SNR is shown in Figure 5e. It can be seen that the overall curve of KADPC algorithm has higher recognition rate than other algorithms under SNR = 2 dB, and the average recognition rate of the overall algorithm is higher. Because the FK algorithm does not specify the number of clustering centers and uses a fuzzy K value for clustering, the number of clustering centers is farther away from the real value with the increase of modulation order. Therefore, the recognition effect is very poor at low SNR.

The recognition accuracy of the five signals under different SNR is shown in Table 1. For BPSK signal, it can be seen from Table 1 that the highest recognition rate of KADPC

model is 78.01% at SNR = −2 dB, while the highest recognition rate of other comparison algorithms is only 63.15%. The recognition accuracy of RF, KNN and SVM reach 88.66%, 88.39% and 83.3% at SNR = 10 dB. The results show that the recognition effect of KADPC algorithm proposed in this paper on BPSK signal is better than other comparison algorithms. For 16QAM signal, it can be seen that KADPC model is ahead of the comparison algorithm at SNR = −2 dB, that is to say, KADPC algorithm model combined with noise reduction has better recognition effect at low SNR for 16QAM signal.

Table 1. Recognition accuracy of communication signals by different methods.

Signal	SNR (dB)	KADPC	FK	RF	KNN	SVM	BP
BPSK	−2	0.7801	0.6315	0.1050	0.0359	0.0298	0.3998
	0	0.8933	0.8339	0.3720	0.1332	0.1453	0.4123
	2	0.9718	0.9603	0.5080	0.3814	0.4057	0.4838
	4	0.9894	0.9753	0.6224	0.5583	0.5538	0.7650
	6	0.9944	0.9882	0.7678	0.6556	0.6550	0.8158
	8	0.9944	0.9946	0.8026	0.8214	0.8269	0.9588
	10	0.9950	0.9965	0.8866	0.8839	0.8333	0.9933
QPSK	−2	0.7278	0.5969	0.0364	0.0326	0.0348	0.3904
	0	0.8537	0.7901	0.2244	0.2354	0.2205	0.4023
	2	0.9327	0.9339	0.3786	0.3747	0.3742	0.4732
	4	0.9933	0.9897	0.5467	0.5411	0.5328	0.7739
	6	0.9903	0.9929	0.6633	0.6661	0.6721	0.8944
	8	0.9997	0.9937	0.7545	0.7716	0.7667	0.9408
	10	0.9953	0.9970	0.8048	0.8280	0.8159	0.9925
8PSK	−2	0.5833	0.4916	0.0000	0.0000	0.0000	0.2437
	0	0.7233	0.7501	0.0049	0.0022	0.0033	0.3916
	2	0.8712	0.8939	0.0425	0.0541	0.0530	0.4256
	4	0.9373	0.9278	0.1033	0.1039	0.1022	0.7490
	6	0.9666	0.9602	0.3814	0.3206	0.3875	0.7949
	8	0.9916	0.9909	0.6080	0.5417	0.6080	0.9397
	10	0.9976	0.9971	0.6241	0.6605	0.6567	0.9977
16QAM	−2	0.5323	0.1342	0.4505	0.3399	0.3482	0.2047
	0	0.7255	0.2270	0.6473	0.5920	0.5367	0.3758
	2	0.7993	0.4107	0.8170	0.7064	0.7064	0.4156
	4	0.8946	0.8231	0.9281	0.8175	0.8175	0.6916
	6	0.9368	0.8509	0.9845	0.8739	0.8739	0.7296
	8	0.9816	0.9506	0.9635	0.9082	0.9082	0.9104
	10	0.9922	0.9943	0.9469	0.8916	0.8916	0.9922
32QAM	−2	0.4840	0.0000	0.4560	0.3482	0.4029	0.2347
	0	0.5791	0.0036	0.6528	0.6003	0.5914	0.3958
	2	0.8690	0.0034	0.8225	0.7147	0.7611	0.4250
	4	0.8993	0.0153	0.9336	0.8258	0.8723	0.6865
	6	0.9918	0.0333	0.9900	0.8822	0.9286	0.7349
	8	0.9988	0.4766	0.9690	0.9165	0.9629	0.9137
	10	0.9958	0.9943	0.9524	0.8999	0.9463	0.9884

The simulation results show that the recognition accuracy of KADPC model for the modulation modes of five digital signals is shown in Figure 6. It can be seen that the overall curve of recognition rate increases with the increase of SNR, and the increase of recognition rate decreases with the increase of modulation signal scale. Due to the large noise interference under low SNR, it is difficult for the algorithm to extract the features of the constellation. At the same time, the increase of modulation order leads to the indistinct degree of eigenvalue discrimination. However, generally speaking, the average modulation recognition rate of KADPC algorithm is high and the recognition effect is good.

The two indexes on the far right are the recognition rate of each signal and the accuracy of the signal from top to bottom. The accuracy is the percentage of the correct

modulation mode classification result and the wrong modulation mode classification result in all samples of the modulation mode. The cells on the diagonal represent the correct classification, the percentage is the recall rate. The remaining cells represent the classification error under this SNR. As can be seen from Figure 7, the KADPC model has great difficulty in feature extraction with the increase of hexadecimal number M. The recall rate of 16QAM and 32QAM modulation types is lower than that of other algorithms. In terms of accuracy and recognition rate, KADPC algorithm has high accuracy and recognition rate when SNR = 4 dB, with an average of more than 90%.

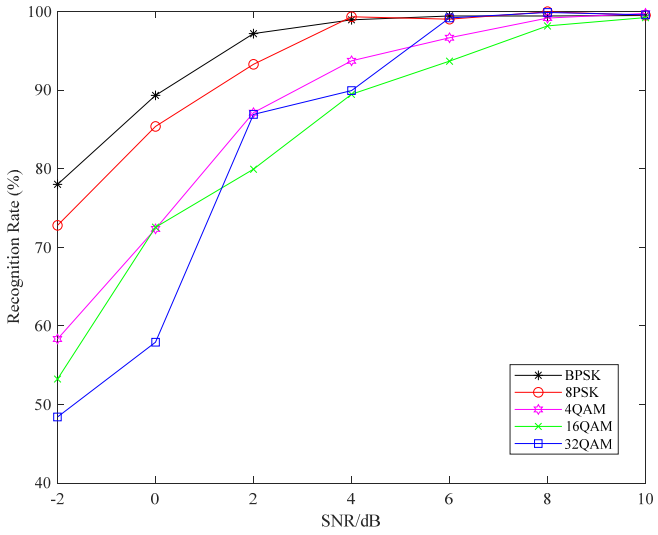


Figure 6. Recognition accuracy of KADPC algorithm for various digital signals.

The confusion matrix calculated by KADPC algorithm when SNR = 4 dB is shown in Figure 7.

	BPSK	QPSK	8PSK	16QAM	32QAM	
BPSK	1789 19%	18	9	3	3	98.7% 1.3%
QPSK	2	1797 19%	8	69	58	92.9% 7.1%
8PSK	39	31	1695 19%	23	18	93.8% 6.2%
16QAM	54	65	17	1618 18%	35	90.4% 9.6%
32QAM	25	44	14	15	1626 18%	94.3% 5.7%

Figure 7. Confusion matrix of KADPC.

The simulation experimental results show that the recognition rate of the proposed KADPC algorithm in this paper is higher than that of the comparison algorithm under different SNR. Because KADPC algorithm denoises the constellation and uses two clustering algorithms to extract the features of the constellation, which makes the constellation features more concentrated and sufficient, so it presents a better recognition effect as a result. Especially for BPSK, QPSK and 8PSK signals, the effect is ideal, and the applicability is good. Due to the difficulty of feature extraction with the increase of modulation order, the improvement is not obvious for 16QAM and 32QAM signals, but the recognition rate is higher than other comparison algorithms at low SNR. In general, the KADPC algorithm

model proposed in this chapter has a good effect on digital signal modulation recognition under different SNR conditions, especially for low SNR.

4. Conclusions

Aiming at the low recognition accuracy of traditional modulation recognition methods under the condition of low SNR, a modulation recognition algorithm model based on digital signal constellation features is proposed. Firstly, the signal is transformed into constellation, and the density clustering algorithm is used for noise reduction. After the features extracted by KADPC algorithm are input into multi-level SVM classifier, the modulation recognition results are obtained. This method reduces the complexity of feature extraction and the cost of classification. Compared with other five modulation recognition algorithm, the overall recognition performance of five common types of digital modulation signals is improved by about 5%. Under the condition of SNR = −2 dB, the correct recognition rate of modulation mode is improved by 8–10% on average. However, with the increase of modulation order, the difficulty of feature extraction of this algorithm increases, resulting in the slow improvement of recognition rate in high-order modulation. How to reduce the complexity of feature extraction of high-order digital signal modulation mode is a research focus in the future.

Author Contributions: Conceptualization, supervision G.L., K.J., A.W. and H.L.; methodology, software, validation, writing—review and editing G.L., X.Q. and A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Auto density-based spatial clustering technique in noise	(ADBSCAN)
Automatic modulation recognition	(AMR)
Attention Relation Network	(ARN)
Adversarial transfer learning architecture	(ATLA)
Backward propagation	(BP)
Binary Phase Shift Keying	(BPSK)
Distributed Space-Time Block Coding	(D-STBC)
Density peak clustering	(DPC)
Fuzzy K-means	(FK)
Gate Recurrent Unit	(GRU)
K-means combined with auto DPC	L-(KADPC)
M-nearest neighbor	N-(KNN)
Long Short-Term Memory	(LSTM)
Quadrature Amplitude Modulation	(QAM)
One-versus-rest	(OVR)
Quadrature Phase Shift Keying	(QPSK)
Phase Shift Keying	(PSK)
Random forest	(RF)
Support vector machine	(SVM)

References

1. Xiang, H.; Jing, X.; Chen, D.; Zhao, Y.G.; Tao, W. A modulation recognition algorithm for UAV emitter. *Mod. Def. Technol.* **2021**, *49*, 98–104.
2. Yihan, X.; Liang, W.; Yuxia, G. Radar signal modulation type recognition based on denoising convolutional neural network. *J. Electron. Inf. Technol.* **2021**, *43*, 2300–2307.
3. Gaohui, L.; Mingtao, X. Research on a modulation recognition method for the FBMC-OQAM signals in 5G mobile communication system. In Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications, Wuhan, China, 31 May–2 June 2018; pp. 2544–2547. [CrossRef]

4. Ge, Y.; Zhang, X.; Zhou, Q. Modulation recognition of underwater acoustic communication signals based on joint feature extraction. In Proceedings of the 2019 IEEE International Conference on Signal, Information and Data Processing, Chongqing, China, 11–13 December 2019; pp. 1–4. [CrossRef]
5. Liu, Y.; Liu, Y.; Yang, C. Modulation recognition with graph convolutional network. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 624–627. [CrossRef]
6. Tian, X.; Sun, X.; Yu, X.; Li, X. Modulation pattern recognition of communication signals based on fractional low-order choi-williams distribution and convolutional neural network in impulsive noise environment. In Proceedings of the 2019 IEEE 19th International Conference on Communication Technology, Xi'an, China, 16–19 October 2019; pp. 188–192. [CrossRef]
7. Tingting, Z.; Yuqiang, F.; Lei, H. Automatic modulation recognition based on depth residual network. *Comput. Simul.* **2021**, *38*, 178–180.
8. Bu, K.; He, Y.; Jing, X.; Han, J. Adversarial transfer learning for deep learning based automatic modulation classification. *IEEE Signal Process. Lett.* **2020**, *27*, 880–884. [CrossRef]
9. Fugang, L.; Ziwei, Z.; Ruolin, Z. Automatic modulation recognition based on CNN and GRU. *Tsinghua Sci. Technol.* **2022**, *27*, 422–431.
10. Zhang, Z.; Li, Y.; Gao, M. Few-shot learning of signal modulation recognition based on attention relation network. In Proceedings of the 2020 28th European Signal Processing Conference, Amsterdam, The Netherlands, 18–21 January 2021; pp. 1372–1376. [CrossRef]
11. Wu, Y.; Li, X.; Fang, J. A deep learning approach for modulation recognition via exploiting temporal correlations. In Proceedings of the 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications, Kalamata, Greece, 25–28 June 2018; pp. 1–5. [CrossRef]
12. Yang, J.; Liu, F. Modulation recognition using wavelet transform based on AlexNet. In Proceedings of the 2019 IEEE 7th International Conference on Computer Science and Network Technology, Kalamata, Greece, 25–28 June 2019; pp. 339–342. [CrossRef]
13. Ji, X.; Wang, J.; Li, Y.; Sun, Q.; Xu, C. Modulation recognition in maritime multipath channels: A blind equalization-aided deep learning approach. *China Commun.* **2020**, *17*, 12–25. [CrossRef]
14. Zhendong, C.; Weining, J.; Changbo, X.; Min, L. Modulation recognition based on constellation diagram for M-QAM signals. In Proceedings of the 2013 IEEE 11th International Conference on Electronic Measurement & Instruments, Harbin, China, 16–19 August 2013; pp. 70–74. [CrossRef]
15. Tayakout, H.; Ghanem, K.; Bousbia-Salah, H. On classifiers for feature-based automatic modulation recognition over D-STBC cooperative networks. In Proceedings of the 2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting, Atlanta, GA, USA, 7–12 July 2019; pp. 1839–1840. [CrossRef]
16. Wang, L.; Li, Y. Constellation based signal modulation recognition for MQAM. In Proceedings of the 2017 IEEE 9th International Conference on Communication Software and Networks, Guangzhou, China, 6–8 May 2017; pp. 826–829. [CrossRef]
17. Punith Kumar, H.L.; Shrinivasan, L. Automatic digital modulation recognition using minimum feature extraction. In Proceedings of the 2015 2nd International Conference on Computing for Sustainable Global Development, New Delhi, India, 11–13 March 2015; pp. 772–775.
18. Deng, D. DBSCAN clustering algorithm based on density. In Proceedings of the 2020 7th International Forum on Electrical Engineering and Automation, Hefei, China, 25–27 September 2020; pp. 949–953. [CrossRef]
19. Aref, Y.; Cemal, K.; Asef, Y.; Amir, S. Automatic fuzzy-DBSCAN algorithm for morphological and overlapping datasets. *J. Syst. Eng. Electron.* **2020**, *31*, 1245–1253. [CrossRef]

electronics

MDPI

Review

Deep Learning in the Phase Extraction of Electronic Speckle Pattern Interferometry

Wenbo Jiang [1,2,*], Tong Ren [1,2] and Qianhua Fu [1,2]

[1] School of Electrical Engineering and Electronic Information, Xihua University, Chengdu 610039, China; rentong@stu.xhu.edu.cn (T.R.); qhfu8@mail.xhu.edu.cn (Q.F.)
[2] Sichuan Provincial Key Laboratory of Signal and Information Processing, Xihua University, Chengdu 610039, China
* Correspondence: jiangwenbo@mail.xhu.edu.cn

Abstract: Electronic speckle pattern interferometry (ESPI) is widely used in fields such as materials science, biomedical research, surface morphology analysis, and optical component inspection because of its high measurement accuracy, broad frequency range, and ease of measurement. Phase extraction is a critical stage in ESPI. However, conventional phase extraction methods exhibit problems such as low accuracy, slow processing speed, and poor generalization. With the continuous development of deep learning in image processing, the application of deep learning in phase extraction from electronic speckle interferometry images has become a critical topic of research. This paper reviews the principles and characteristics of ESPI and comprehensively analyzes the phase extraction processes for fringe patterns and wrapped phase maps. The application, advantages, and limitations of deep learning techniques in filtering, fringe skeleton line extraction, and phase unwrapping algorithms are discussed based on the representation of measurement results. Finally, this paper provides a perspective on future trends, such as the construction of physical models for electronic speckle interferometry, improvement and optimization of deep learning models, and quantitative evaluation of phase extraction quality, in this field.

Keywords: electronic speckle pattern interferometry (ESPI); phase extraction; deep learning; fringe pattern; wrapped phase maps

1. Introduction

Electronic speckle pattern interferometry (ESPI) is a nondestructive optical measurement technique known for its simplicity, high speed, large field of view, excellent anti-interference performance, and suitability for dynamic measurements [1–4]. ESPI has numerous research applications in fields such as industry and engineering, including strain [5,6], displacement [7,8], vibration measurements [9,10], and material defect detection [11,12]. Phase information can be extracted by analyzing the interference fringe image, and the measured physical quantities, such as displacement and deformation, are closely related to the phase. As a result, one challenge in ESPI is the accurate extraction of phase information from interference fringe images for the analysis of the measured physical phenomena. Therefore, the difficulty, accuracy, generalization, and extraction speed of phase extraction methods are crucial for this technology.

Typically, image sensors and human eyes perceive and record the intensity distribution of light field through the photoelectric conversion effect. Because the frequency of a light wave being much larger than the sampling frequency, it is impossible to directly perceive and record the phase information of a light field. Fortunately, the phase delay changes the intensity distribution as the light field propagates. Consequently, the corresponding phase distribution can be deduced from the recorded light field intensity distribution, and the physical quantity to be measured can then be calculated through the phase transformation [13]. This process is known as phase extraction.

Current methods for phase extraction in ESPI can be categorized into two types, namely classical phase extraction methods, which are represented by techniques such as phase shifting [14] and carrier methods [15], and emerging data-driven methods based on deep learning. Phase shifting methods consider the known phase differences between multiple frames of fringe patterns, whereas carrier methods consider the frequency spectrum distribution characteristics of fringe patterns to extract phase information. Classical phase extraction methods have been widely used in industrial engineering, providing rich phase information and enabling economical and efficient noncontact full-field measurements. However, these methods involve complex and cumbersome steps requiring manual parameter tuning. Furthermore, dealing with noise, background interference, and phase discontinuities remains challenging. These challenges necessitate researchers to investigate novel methods for addressing these problems.

Deep learning network models, with convolutional neural networks (CNN) as a representative, have demonstrated the capability to extract complex features and patterns from images and achieved satisfying results in computer vision, data image processing, and related fields [16]. Some scholars have introduced deep learning into ESPI phase extraction and achieved notable results [17].

Because of the differences in measurement methods, the measurement results of ESPI technology typically manifest as ESPI fringe patterns or ESPI-wrapped phase maps. Deep learning methods are increasingly being used in both these results. This paper provides an overview of the fundamental principles and image characteristics of ESPI, emphasizing the application of existing deep learning methods in these two distinct measurement results. In addition, this paper compares the performance of various deep learning methods and highlights their advantages in addressing the challenges faced by conventional methods. Furthermore, the limitations of existing methods are detailed, and a prospective analysis of the development trends, covering aspects such as the construction of physical models for ESPI, improvement and optimization of deep learning models, and quantitative evaluation of phase extraction quality, is presented. This paper provided valuable insights for the advancement of phase extraction in ESPI.

The main contributions of this paper are as follows:

(1) The application of the existing deep learning methods for phase extraction in the field of ESPI is comprehensively analyzed and summarized. By comparing the performance of different methods and the potential advantages of deep learning compared with traditional methods, this paper provides readers with a comprehensive understanding and evaluation of this topic.
(2) The development prospects of deep learning methods for phase extraction in the field of ESPI are explored. This paper discusses the possible challenges of current deep learning methods in ESPI phase extraction, and it proposes directions for future improvement and development to provide a reference for researchers.
(3) This paper provides knowledge and guidance for scholars who want to use deep learning methods to measure ESPI strain and displacement in the future.

2. Production of ESPI Images

Speckle was initially regarded as a common optical phenomenon. However, in holography, speckles considerably affect the quality of holographic images Therefore, in holography, speckles are considered noise, and reduction of speckle interference is a critical topic of research. In 1966, Ennos [18] discovered that randomly distributed speckles can be used as information carriers, including information on light intensity and phase. Since then, people have conducted considerable research on speckle measurement technology. In ESPI, a charge-coupled device is used to record speckle images, which are processed on a computer. Speckle is a noncontact full-field optical measurement method. This technology overcomes the limitations of previous optical measurement technology applications and is a speckle interferometric measurement method implemented by electronics and digital methods. This method has high accuracy, rapid response, and full-field noncontact.

Currently, the ESPI technology measurement system consists of several key components, including lasers, CCD cameras, computers, image acquisition cards, and some optical elements. Figure 1 displays a typical ESPI measurement optical path, where M represents a reflective prism, and BS_1 and BS_2 are spectral prisms. The laser beam is categorized into a measurement beam and a reference beam after passing through spectral prism BS_1, and the measurement beam is irradiated onto the measured object. After passing through the beam expander, the measurement beam interferes with the reference beam after passing through reflective prism BS_2, and the CCD camera records interference fringes, transfers the image information to the computer for processing, and displays the test results in real time.

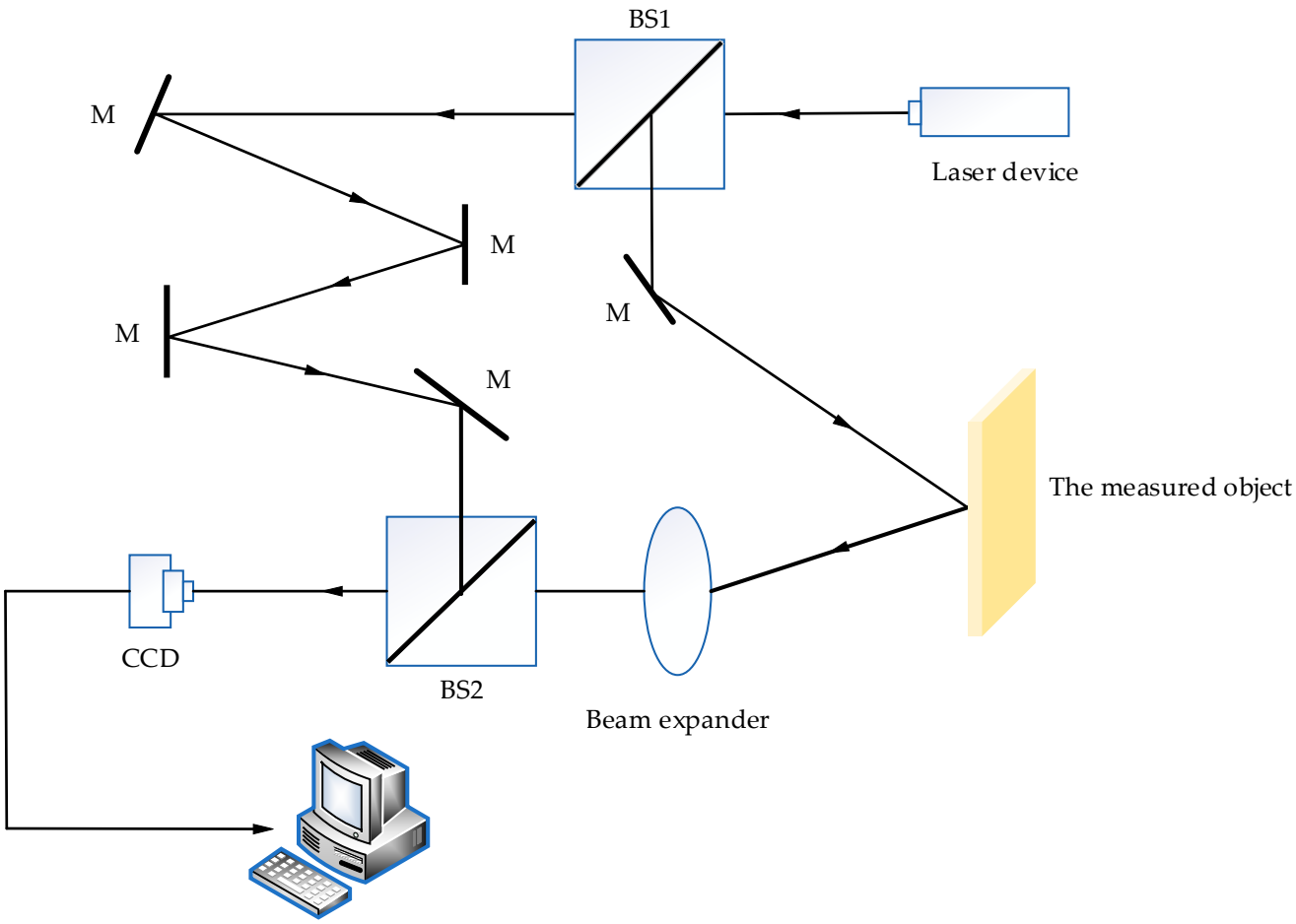


Figure 1. A typical ESPI measurement optical path.

The recorded speckle information before and after object deformation is expressed as follows:

$$I_1 = I_0 + I_r + 2\sqrt{I_0 I_r}\cos\theta \tag{1}$$

$$I_2 = I_0 + I_r + 2\sqrt{I_0 I_r}\cos(\theta + \varphi) \tag{2}$$

where I_1 and I_2 represent the intensity of the speckle field before and after deformation, respectively; I_0 and I_r represent the intensity of the two coherent beams, respectively; θ is the phase difference in the speckle field on the object plane; φ is the change inphase difference caused by object deformation. The relationship between φ and deformation displacement W is as follows:

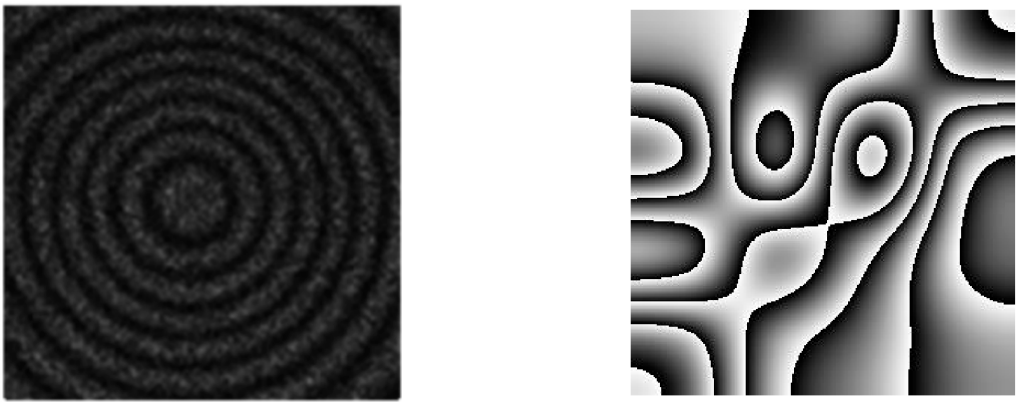
$$\varphi = \frac{2\pi}{\lambda} W \tag{3}$$

By subtracting the light intensity of the speckle pattern before and after the deformation of the measured object, as follows:

$$\Delta I = I_1 - I_2 = 4\sqrt{I_0 I_r}\sin\frac{2\theta+\varphi}{2}\sin\frac{\varphi}{2} \tag{4}$$

With the change in the value of φ, the light intensity difference ΔI also changes, which allows us to observe fringe information that is not visible in a certain light field intensity, resulting in alternating bright and dark fringe images.

The measurement results of ESPI are usually the ESPI fringe pattern derived from the formula and the ESPI-wrapped phase map obtained by installing a phase shifter on the reference optical path and using a stepwise phase shift. As displayed in Figure 2, both contain considerable noise, and the ESPI fringe pattern presents a black and white, alternating light and dark fringe structure, with the density and direction of the fringes reflecting the phase difference changes on the object surface. The ESPI-wrapped phase map limits the phase information to the range $[-\pi, \pi]$ and displays it in grayscale or pseudo-color. The physical quantity to be measured is directly related to the phase distribution information of both. Therefore, studying advanced electronic speckle interferometry phase extraction methods is critical for quickly and accurately analyzing the measured physical quantity from the ESPI fringe pattern or wrapped phase map and has become a critical topic of research in current ESPI.



(**a**) (**b**)

Figure 2. Two measurement results of ESPI technology ((**a**) ESPI fringe pattern and (**b**) ESPI-wrapped phase map).

3. Application of Deep Learning in the Phase Extraction of ESPI Fringe Patterns

The fringe centerline method is a classic phase extraction method for ESPI fringe patterns. Compared with other phase extraction methods, such as Fourier transform and wavelet transform, the fringe centerline method is the most direct phase extraction method based on a single fringe pattern. In this method, first, the fringe pattern is filtered for noise removal, the skeleton line of the ESPI fringe pattern is extracted, the series of skeleton lines is identified, and the phase values at the location of the skeleton lines are obtained. Finally, a specific algorithm was used to interpolate the identified skeleton lines to obtain full-field phase information. Figure 3 displays the steps of the method. The following sections explain the application of deep learning in this method.

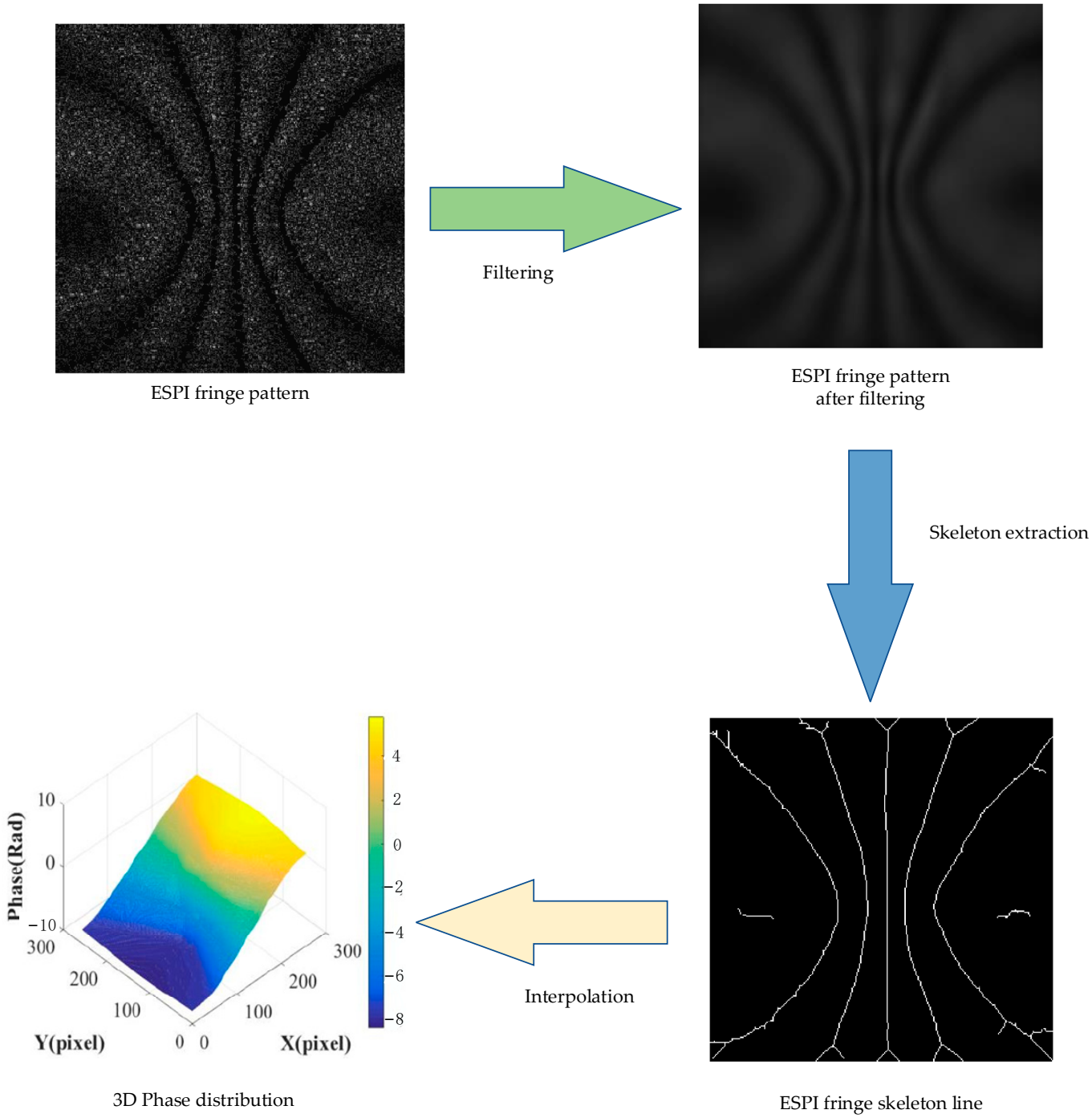


Figure 3. Phase extraction steps of ESPI fringe pattern [19].

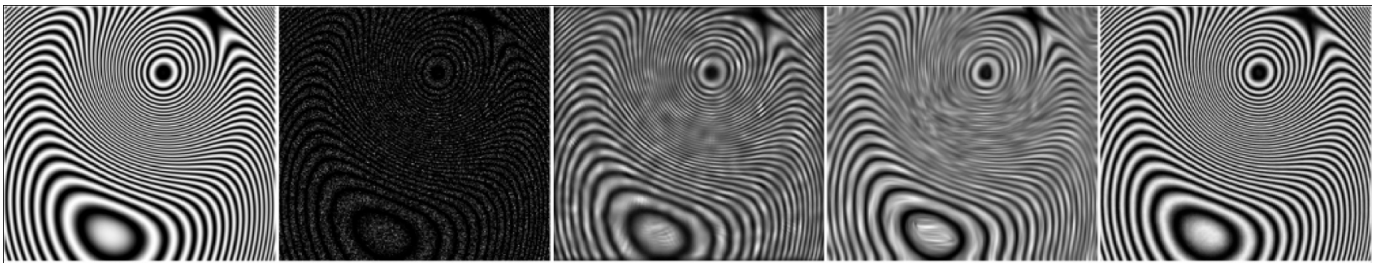
3.1. Application of Deep Learning in ESPI Fringe Pattern Filtering

The ESPI fringe pattern obtained from actual measurement contains considerable speckle noise. Therefore, prior to extracting the phase, the first step is to perform filtering processing. According to the various densities of fringes, the ESPI fringe pattern can be categorized into low-density fringe pattern, high-density fringe pattern, variable-density fringe pattern, and discontinuous fringe pattern. Conventional fringe pattern filtering methods include the sine and cosine mean method, the window Fourier transform method, and the two-dimensional wavelet transform method [20]. These methods are classified according to various objects. In the actual processing process, the following problems are persistent:

(1) Slow processing speed, rendering handling multiple frames simultaneously impossible.
(2) Suboptimal denoising effect on complex images, such as high- and variable-density ESPI fringe patterns.
(3) Poor generalization, necessitating repeated fine-tuning for various objects, and lack of a general model.
(4) Substantial computational parameters.

To address the problems of slow speed and the inability to process multiple frames simultaneously in conventional filtering methods, in 2019, Hao et al. from Tianjin University were inspired by the FFD-Net model [21] and introduced it into ESPI fringe pattern denoising [22]. This method can process 50 images simultaneously and requires only 8 s for processing, but the speed of the method is slow.

In 2020, Lin et al. [23] from Shandong University addressed the challenge of poor denoising performance of conventional methods for ESPI fringe patterns with high variation in density. They introduced the FPD–CNN network, which considerably outperformed the conventional windowed Fourier transform (WFT) method and coherence enhancing diffusion (CED) method, achieving the peak-signal-to-noise ratio (PSNR), the mean of structural similarity index (SSIM), and the mean absolute error (MAS) metrics of 27.88 (dB), 0.972, and 7.642, respectively, at a processing speed of just 0.57 s (Figure 4). The FPD–CNN method provides superior denoising performance for fringe patterns with varying densities. In 2022, Wang et al. [24] from Shenyang Aerospace University incorporated the squeeze and excitation (SE) attention mechanism into the DnCNN denoising model, introducing the AD-CNN network for ESPI fringe pattern denoising. This method achieved a PSNR of 20.1532 (dB) on a self-made dataset, surpassing the performance of the classic denoising model DnCNN and the WFF method, which achieved PSNR values of 17.0934 and 17.0068 dB, respectively.



(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 4. Comparison of denoising high-variation density fringe patterns obtained using various methods [23]. (**a**) Noise-free image. (**b**) Noisy image. (**c**) WFF method. (**d**) CED method. (**e**) FPD-CNN method.

To improve the generalization ability of the model, in 2022, Xu et al. from Tianjin University [25] constructed an MDD-Net network model using a skip connection method to filter various types of ESPI fringe patterns. The team applied this method to the measurement of the out-of-plane displacement of thermal changes in PCB boards, and its measurement accuracy can reach the nanometer level. This method outperforms existing SOOPDE, WFF, and BL-Hilbert-L2 methods in terms of detail protection, speckle reduction, and shape preservation under various types of density fringes. MDD-Net outperforms the FFD-Net proposed in reference [22] in discontinuous density fringe filtering denoising, the former having an SNR of 40.6 and SSIM of 0.991, which are superior to the latter's SNR of 38.387 and SSIM of 0.965, indicating its superior generalization ability. Figure 5 displays the specific filtering effect.

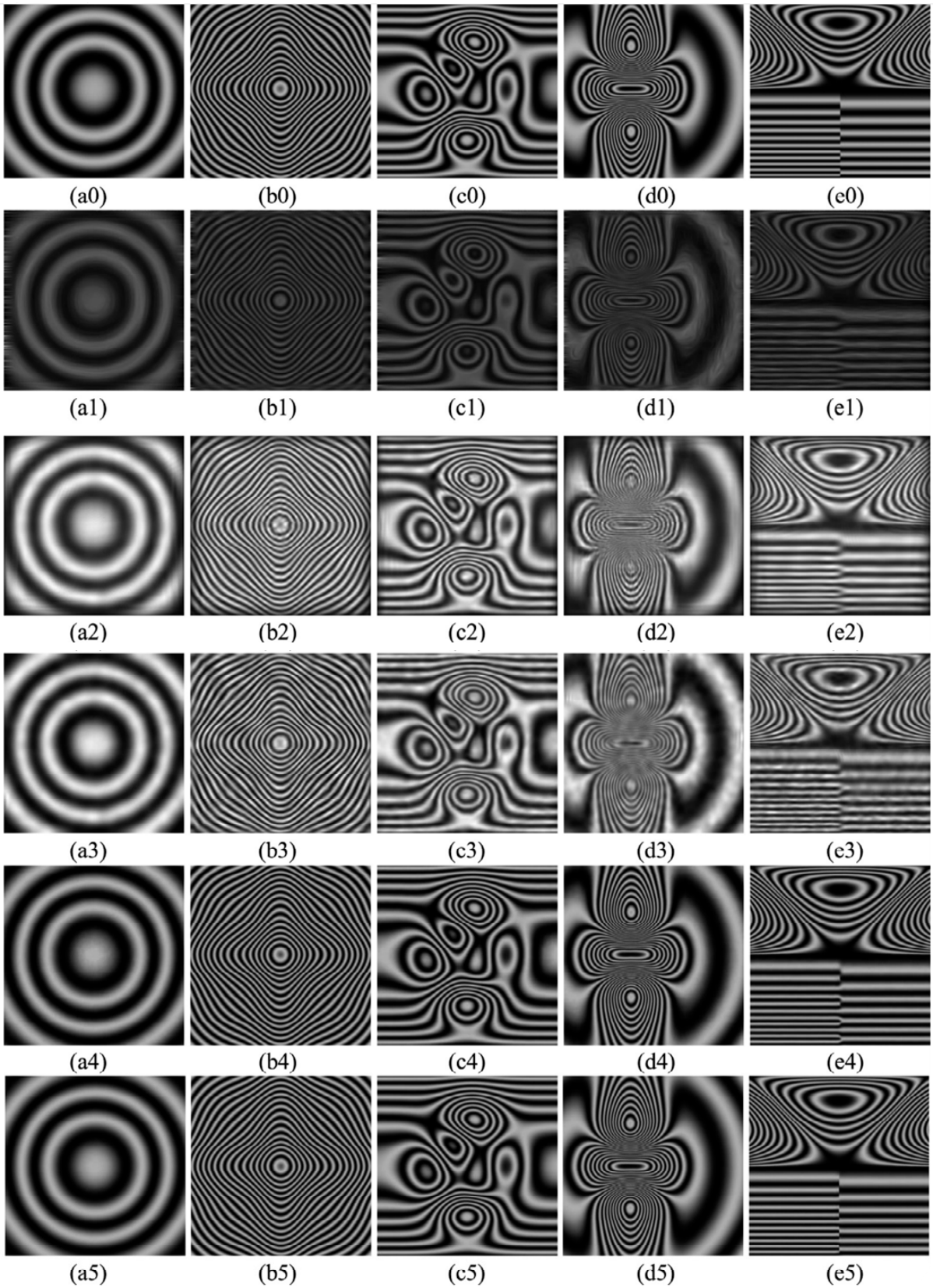


Figure 5. Comparison experiments of ESPI fringe pattern filtering (**a**–**e**), respectively represent low-density, high-density, variable-density, greatly variable density, and discontinuous fringe samples. (**a0**–**e0**), (**a1**–**e1**), (**a2**–**e2**), (**a3**–**e3**), (**a4**–**e4**), and (**a5**–**e5**) are the true labels of the simulated ESPI fringe patterns, SOOPDE, WFF, BL-Hilbert-L2, FDD-Net, and MDD-Net methods, respectively) [25].

This research achieved considerable success in terms of generalization and accuracy, but increased the model parameter size and reduced the processing speed. To solve this problem, in 2022, Gurrola-Ramos et al. from the Mathematics Research Center, Guanajuato, Mexico, proposed a lightweight residual dense network based on the U-Net structure (LRDUNet) [26]. This network reduced the number of parameters using grouped convolution and density convolution modules. The number of parameters in this method was only 4.21 M, and the processing speed reached 0.00665 s. This method has excellent denoising performance, with a PSNR of 40.1427 dB. Figure 6 displays the specific structure of this network.

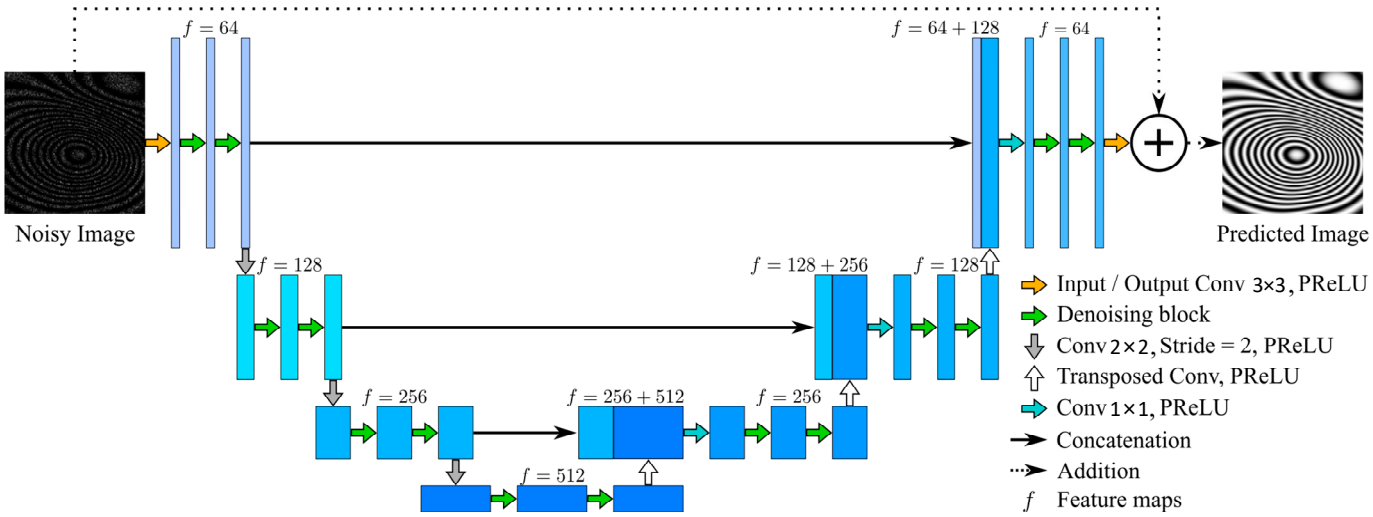


Figure 6. LRDUNet network model [26].

Table 1 compares the denoising performance indicators for each model, where the "Parameters" values of the FFD-Net and MDD-Net model were converted from the literature [22,26], respectively. The conversion formula is 1 M = 1024 K.

$$MAE = \frac{1}{m}\sum_{i=1}^{m}\left|y_i - y_i'\right| \tag{5}$$

$$MSE = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - y_i'\right)^2 \tag{6}$$

$$PSNR = 10log_{10}\frac{(MAX_I)^2}{MSE} = 20log_{10}\frac{MAX_I}{\sqrt{MSE}} \tag{7}$$

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{8}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(y_i - y_i'\right)^2} \tag{9}$$

In the above formula, m is the number of samples in the dataset, and y_i and y_i' prime represent the true value and predicted value of the ith sample, respectively. MAX_I is the maximum value of the image point color; if each sample point is represented by 8 bits, then it is 255. μ_x and μ_y represent the mean values of images X and Y, respectively. σ_x and σ_y represent the variances of images X and Y, respectively. σ_{XY} represents the covariance of images X and Y. c_1 and c_2 are constants.

Table 1. Comparison of denoising performance indicators for each model [21,23–26].

Performance Indicators	FFD-Net [21]	AD-CNN [24]	FPD-CNN [23]	MDD-Net [25]	LRDUNet [26]
PSNR (dB)	35.2269	25.0397	27.88	/	40.1427
SSIM	0.9932	0.9487	0.972	0.991	0.9980
MAE	0.0117	/	0.0135	/	0.0067
Parameters (M)	0.475	/	1.86	1.445	4.21
Running time (s)	0.024	0.0469	0.57	0.041	0.00665

Note: 1. The above data are the experimental results under their respective data sets. 2. The calculation formula for evaluation indicators is as follows:

The existing deep learning ESPI fringe pattern denoising methods require large models with poor generalization to different types of fringe patterns. Therefore, future development requires adaptability, high accuracy, and small model parameters.

3.2. Application of Deep Learning in the Fringe Skeleton Extraction Method

After filtering, the fringe skeleton lines should be extracted. The quality of skeleton line extraction directly affects the accuracy of phase extraction. The conventional fringe skeleton line is based on image binarization technology, which performs morphological operations and connectivity analysis on binarized images to extract fringe skeleton lines. Early representative skeleton extraction methods included a binarization threshold refinement algorithm [27], stripe peak tracking method [28], and binary graph method based on two-dimensional derivative symbols [19]. Subsequent studies have considered the characteristics of stripe direction and proposed new stripe skeleton line extraction methods, such as the coupled gradient vector flow field method [29], direction-coupled gradient vector flow field extraction method [30], anisotropic gradient vector flow field extraction method [31], and gradient vector field of variational image segmentation solution [32]. The conventional fringe skeleton line extraction method considers the image features and directionality of fringes, but has the following problems:

(1) Low accuracy, poor generalization, slow extraction speed, and inability to process multiple frames simultaneously.
(2) Poor extraction of damaged fringe patterns.

Deep learning has a higher tolerance to structural changes and requires less prior knowledge of the input image structure than conventional methods. Therefore, this method is used for extracting fringe skeleton lines. According to the processing methods, this method can be divided into image segmentation tasks and image generation tasks.

For (1), in 2019, Li et al. at Tianjin University proposed using the U-Net network for ESPI fringe skeleton line extraction [33]. This method treats skeleton line extraction as an image segmentation task from the background, successfully achieving skeleton line extraction. The method achieved an accuracy of 0.91 and a processing speed of 0.5 FPS and can extract different types of fringe skeleton lines with excellent generalization. However, this method performs poorly on fringe patterns with low contrast and considerable speckle noise. In 2022, the team improved on previous work by introducing the Channel-wise Cross Fusion Transformer into the U-Net network [34], which solved the problem of low accuracy. This method achieved an accuracy of 0.9878 and simultaneously performed phase extraction on the extracted fringe skeleton lines. Compared with conventional methods, the phase obtained by this method is smoother, and the performance regarding evaluation metrics is better, with an SSIM of 0.908 and root mean square error (RMSE) of 0.003. However, this method did not mention the average time for fringe extraction. Figure 7 displays a comparison of phase extraction results of various methods in the literature [34].

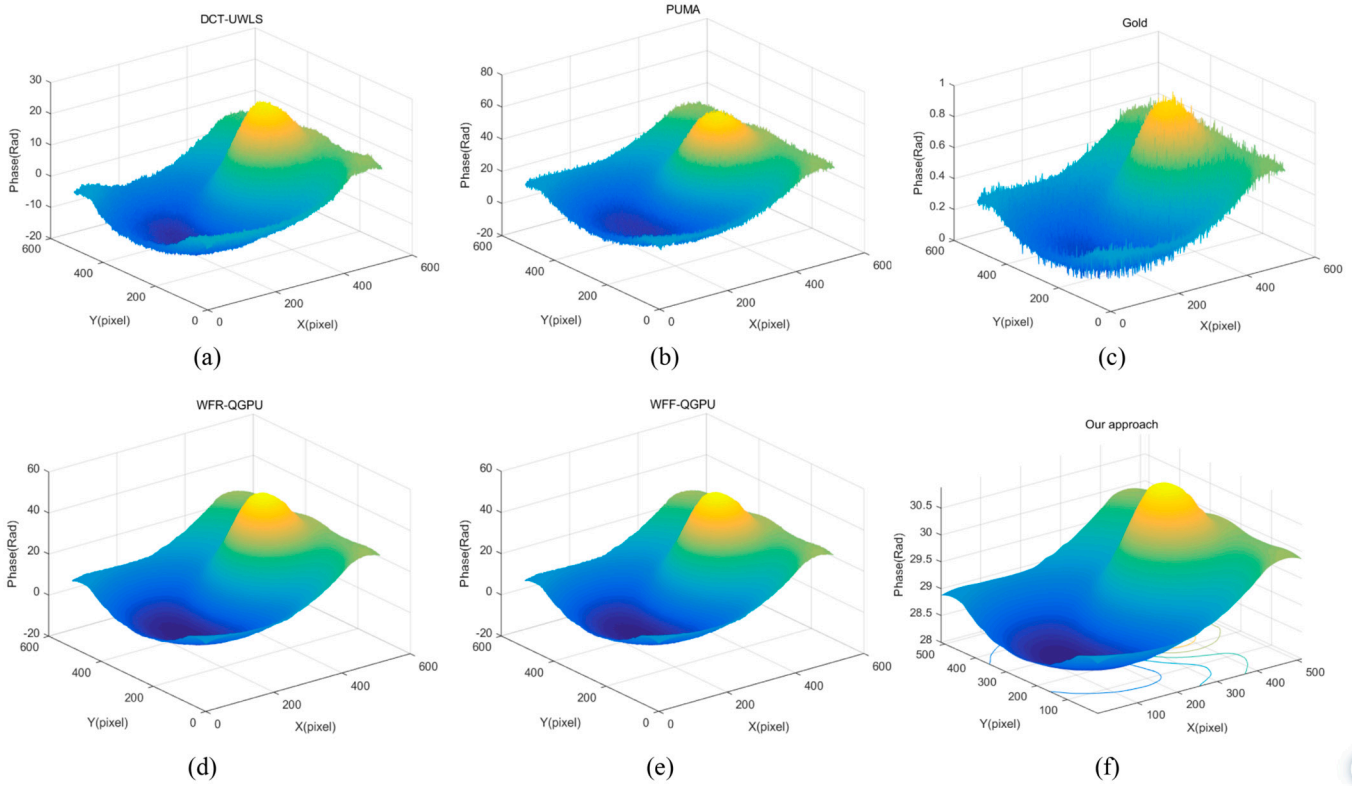


Figure 7. Comparison of phase extraction results using different methods (**a**–**f**) show the methods of UWLS, Graphcut, Goldstein, WFR-QG, WFF-QG, and the literature [34], respectively).

For problem (2), Liu et al. from Tianjin University proposed the M-Net network in 2020, which is an image segmentation task [35]. This method was used to analyze the experimental results obtained from the dynamic measurement of the thermal deformation of ceramic plates under strong light. Compared with conventional gradient vector field (VID-GVF) based on variational image decomposition [32] and U-Net [33] networks, M-Net achieved a superior extraction effect on the skeleton broken or missing stripes and can automatically fill in the damaged and broken parts. In terms of evaluation indicators, the accuracy, recall rate, and AUC score of the M-Net network were 0.959, 0.680, and 0.968, respectively, which were higher than those of VID-GVF (0.911, 0.531, and 0.735) and U-Net (0.952, 0.582, and 0.949). Figure 8 displays the specific experimental results, with red boxes representing damaged fringes.

At Hebei University of Technology, Wang et al. [36] considered skeleton line extraction as an image generation task. In the literature [36], Wang et al. used Pix2pix cGan [37] as a skeleton line extraction network, which can extract stripe skeleton lines and solve problem (1). This method is fast, reaching 11.7 FPS and an accuracy of 0.9704. Table 2 exhibits the comparison of the performance of five methods for extracting skeleton lines.

$$\text{Acc} = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{11}$$

$$Specificity = \frac{TN}{(FP + TN)} \tag{12}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{13}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{14}$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

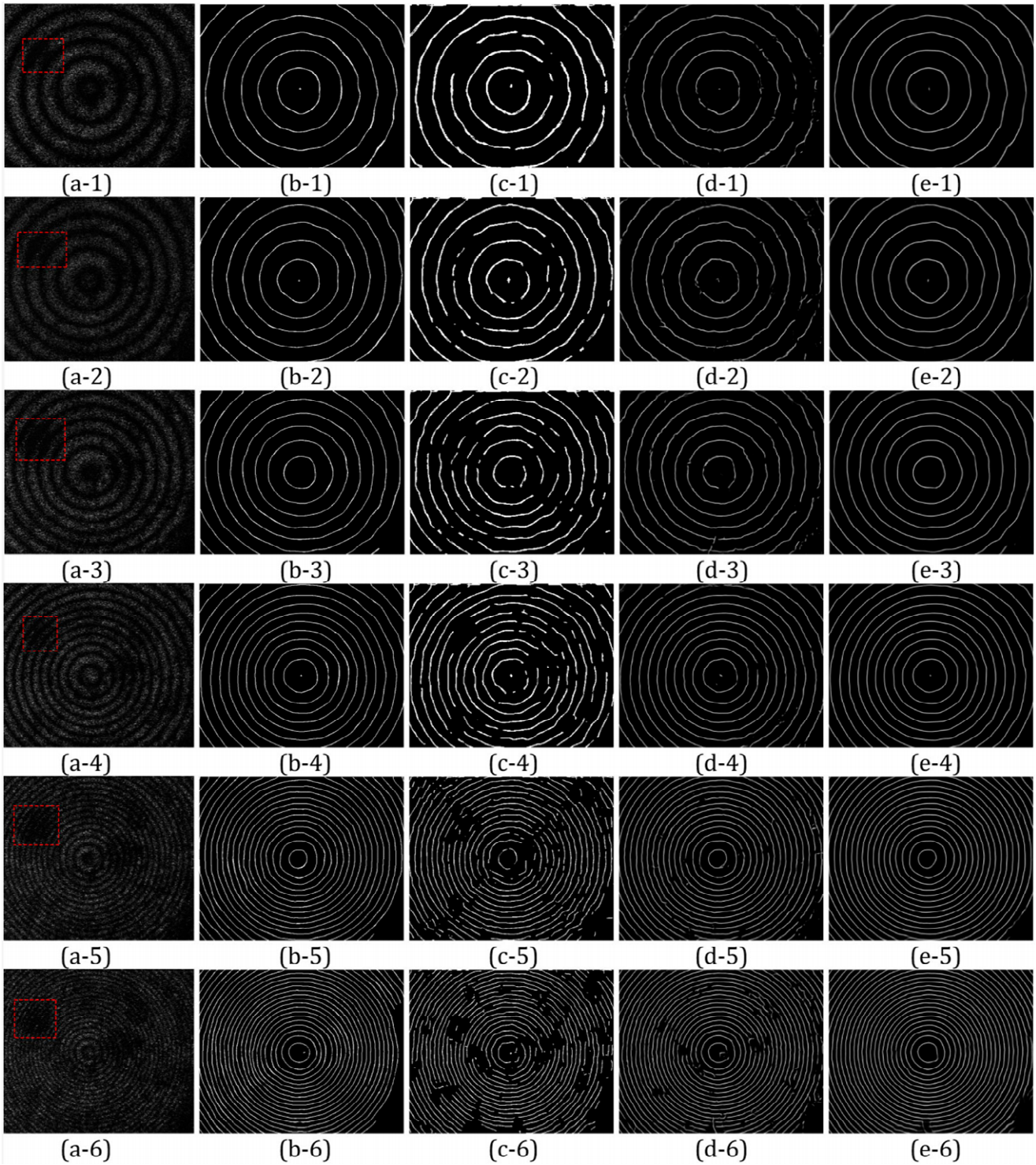


Figure 8. Comparison of skeleton line extraction methods under thermal deformation of ceramic plates [35]. (**a**–**e**) Extracted skeleton lines using the measured fringe pattern and ideal skeleton line. (**a-1**–**a-6**) Input test images. (**b-1**–**b-6**) Ideal skeletons corresponding to the test images. (**c-1**–**c-6**) VID-GVF extraction method. (**d-1**–**d-6**) U-net extraction method. (**e-1**–**e-6**) M-Net extraction method. Red boxes represent damaged fringes.

Table 2. Comparison of the performance of five methods for extracting skeleton lines [32–36].

Methods	Acc	Sensitivity	Specificity	F1	CC	Running Time(s)
D-GVF [32]	/	/	/	/	0.9775	/
U-net [33]	0.9133	0.6884	0.8621	0.6254	0.9408	0.5
CTrans U-net [34]	0.9878	0.8890	0.9887	0.8881	0.9905	/
M-Net [35]	0.9571	0.7412	0.9411	0.7411	0.9521	/
Pix2pix cGan [36]	0.9704	/	/	0.8174	0.8633	11.7

Note: The above data are the experimental results under their respective data sets. The calculation formula for evaluation indicators is as follows:

True Positive (*TP*): the number of samples that are actually positive and correctly predicted as positive.

True Negative (*TN*): the number of samples that are actually negative and correctly predicted as negative.

False Positive (*FP*): the number of samples that are actually negative but incorrectly predicted as positive.

False Negative (*FN*): the number of samples that are actually positive but incorrectly predicted as negative.

Existing deep learning methods have fewer applications in fringe skeleton line extraction, partly because the physical model for filtering denoising is more explicit, and more mature CNN models have been devised in other image processing fields, which can be directly imported into ESPI denoising. Most deep learning-based fringe skeleton line extraction methods add structural components in the encoder and decoder paths to adapt to ESPI fringe processing. Compared with the following methods, the previous method still requires specific algorithms for skeleton line extraction after filtering denoising, which increases the difficulty of fringe skeleton line extraction. Therefore, directly extracting fringe skeleton lines from the original fringe image without preprocessing could be the focus of research in the future.

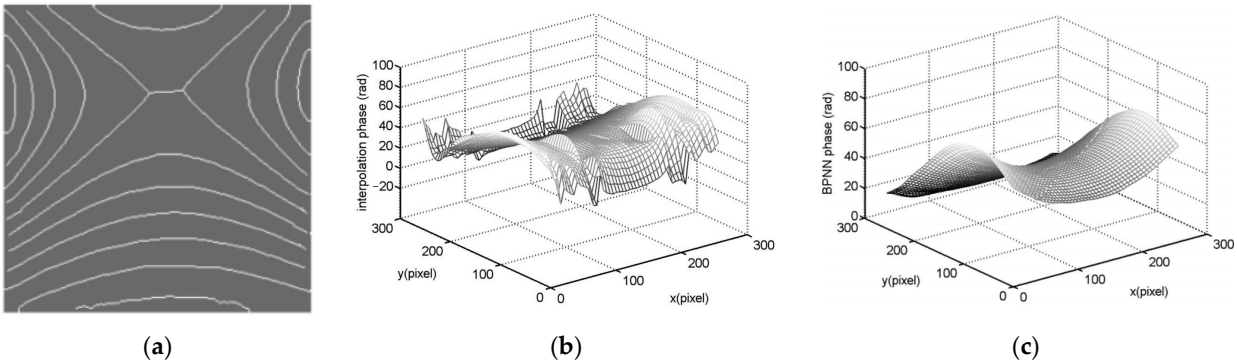
3.3. Application of Deep Learning in the Interpolation Algorithm

The extracted fringe skeleton lines are discrete sampling points. To obtain the phase information of the entire field, an interpolation algorithm is required to predict the unsampled phase distribution of the entire field. Classic interpolation algorithms include bilinear interpolation [38], nearest neighbor interpolation [39], C spline interpolation [40], and interpolation methods based on the heat conduction theory [31]. Although these methods are simple and fast, they exhibit the following problems:

(1) The interpolation result is not sufficiently accurate, and the resulting phase is not sufficiently smooth.
(2) The quality requirements for the skeleton line are high.
(3) Interpolation speed is slow.

Currently, the phase extraction interpolation algorithm for ESPI fringe patterns still relies on neural networks or simple machine learning methods, as the previous method displays in image format, whereas the subsequent one is related to mathematical methods.

In 2007, Tang et al. at Tianjin University first proposed using a BP neural network for fringe skeleton line interpolation [41]. Figure 9 displays the results, which reveal that the phase information obtained by the network is smoother. However, this method requires specialized partial differential equation methods for preprocessing and lacks specific evaluation metrics to quantify the interpolation results. Determining the true phase accuracy obtained from the interpolated image alone is impossible.

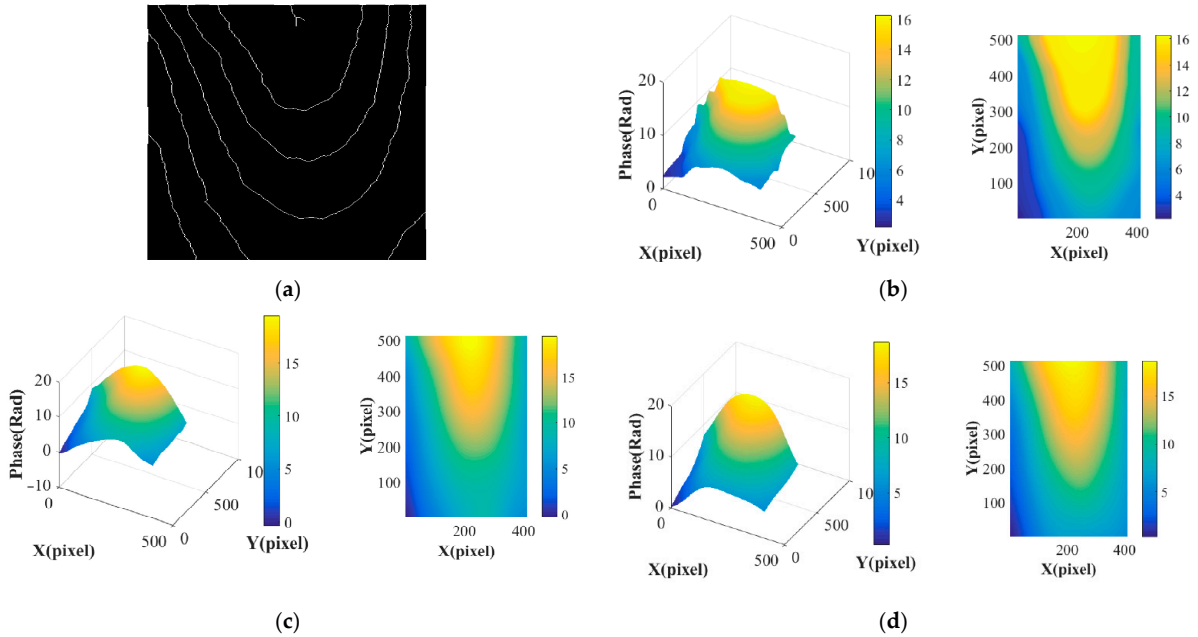


(**a**) (**b**) (**c**)

Figure 9. Interpolation results of different methods [41] (**a**–**c**) white striped skeleton line, C spline interpolation method 's result, and BPNN method's result).

In 2011, Wang et al. from North China University of Technology proposed the radial basis function (RBF) neural network interpolation method, which solves the problem of high-quality requirements for skeleton lines and can obtain excellent results even for severely fractured skeleton lines [42]. However, similar to [41], specific quantitative metrics for evaluating the phase information obtained after interpolation were not provided. In addition, this method did not fully leverage the experimental data information from ESPI fringe patterns.

In 2020, Chen from Tianjin University proposed using hierarchical recurrent neural networks for interpolation processing, which solved the problem of slow processing speed [19]. In the thermal deformation experiment of steel plates under high-intensity laser irradiation, this method could perform interpolation operations on the fringe patterns obtained from dynamic measurements in only 4 s, which considerably lowered the 25 s required in the conventional method. Compared with the BP neural network method, the processing time of both methods was identical, but the phase information of the previous method was smoother. Figure 10 displays the specific interpolation results.



(**a**) (**b**) (**c**) (**d**)

Figure 10. Three-dimensional phase maps and two-dimensional pseudo-color phase maps interpolated using three different methods [19]. (**a**) Dynamic measurement fringe pattern of steel plate hot deformation. (**b**–**d**) Three-dimensional phase pattern and two-dimensional pseudo-color phase pattern from C-spline interpolation, a BP neural network, and reference [19], respectively.

Currently, interpolation algorithms have not been improved considerably, and the existing literature lacks quantitative evaluation metrics for phase information after interpolation. In this paper, because the evaluation of phase information only based on the smoothness of the image was inadequate, other evaluation metrics were required for quantification, For example, comparing the phase information after interpolation of classic methods with that based on neural network interpolation, calculating SSIM or RMSE between the two can make the performance comparison between methods more specific.

4. Application of Deep Learning in Phase Extraction of the ESPI-Wrapped Phase Image

Figure 11 displays the phase extraction method based on wrapped phase maps. The wrapped phase map is obtained using phase shifting, is then filtered and denoised, and finally unwrapped to obtain the final full-field phase information. The following sections describe the application of deep learning in the phase extraction of ESPI-wrapped phase maps.

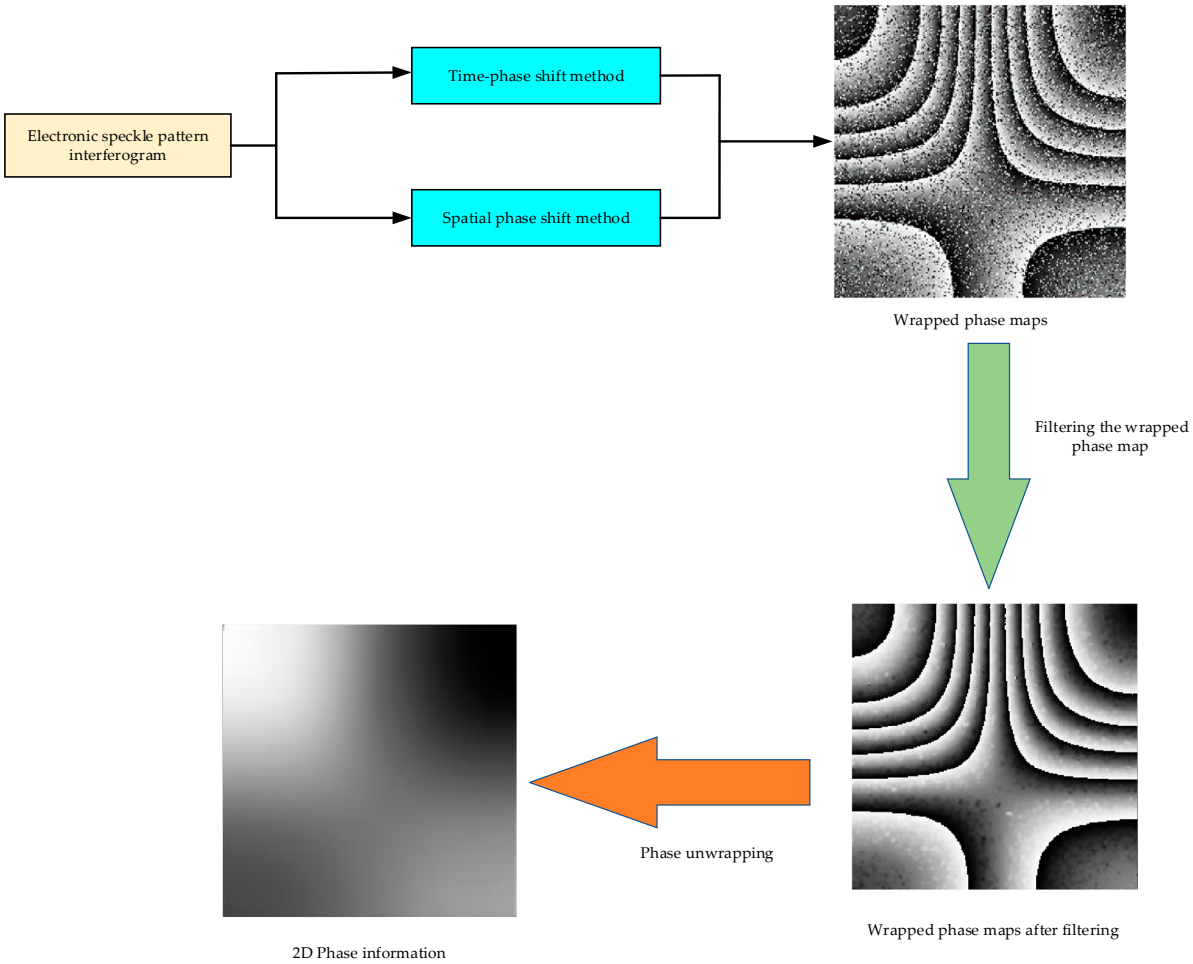


Figure 11. Schematic of the phase extraction method based on the wrapped phase map [19].

4.1. Application of Deep Learning in the Phase Shift Method

According to the different methods of changing the phase difference, the classical phase shift method can be divided into the time phase shift and spatial phase shift method. The time phase shift method changes the phase shift amount on the time series to measure the deformation of the object and obtain deformation information. The spatial phase shift method obtains phase information by applying various phase shifts at one time. According to the different phase shift amounts, the three-step phase shift method [43], four-step phase shift method [44], five-step phase shift method [45], and so on have emerged

successively. Their main difference is in the number of interferometric images used and the measurement accuracy. This article is focused on the most commonly used four-step phase shift algorithm. The four-step phase shift method collects four speckle patterns before and after object deformation, with a phase shift of 90° for each time in the speckle pattern collection. The intensity distribution of the four speckle interference patterns collected by the CCD can be expressed as follows:

$$\begin{array}{c} I_1(x,y) = A(x,y) + B(x,y)cos[\varphi(x,y)] \\ I_2(x,y) = A(x,y) + B(x,y)cos[\varphi(x,y) + \pi/2] \\ I_3(x,y) = A(x,y) + B(x,y)cos[\varphi(x,y) + \pi] \\ I_4(x,y) = A(x,y) + B(x,y)cos[\varphi(x,y) + 3\pi/2] \end{array} \tag{16}$$

Solving the above equations simultaneously, we can obtain the following phase of the optical field:

$$\varphi(x) = arctan\frac{I_4(X,Y) - I_2(X,Y)}{I_1(X,Y) - I_3(X,Y)} \tag{17}$$

Improved four-step phase shift methods such as 4 + 2 [46] and 4 + 1 time phase shift techniques [47] have since emerged.

From a mathematical perspective, the classical phase shifting method involves adding a known phase shift under various conditions to increase the number of equations, rendering the number of equations greater than or equal to the unknowns, and subsequently solving for the phase value. When using conventional phase shifting methods for phase extraction, selecting an appropriate method based on the differences in processing objects and acquisition speed is necessary. In practice, the following problems are observed:

(1) Conventional phase shifting requires the use of multiple images for phase extraction, whereas phase extraction based on a single fringe pattern exhibits lower accuracy. Balancing the two to achieve higher phase accuracy using fewer images remains challenging.
(2) The process of time phase shifting requires the acquisition of multiple images, which requires considerable time and limits dynamic measurement.

In 2019, at the Nanjing University of Science and Technology, Feng et al. first demonstrated that deep neural networks can be trained to perform fringe analysis [48]. This study proposed a two-stage convolutional neural network. In the first network, the background intensity information in the fringe pattern was removed, and the output of the second network was the numerator and denominator of the arctangent function, as shown in Equation (17). Finally, the phase was calculated based on the output of network using the arctangent function. In this method, the phase was extracted using a single fringe pattern, and conventional phase shifting methods were combined to solve problem (1). The average absolute error of phase extraction using this method, Fourier transform (FT), and WFT was 0.087, 20, and 0.19 rad, respectively. Thus, the phase extraction accuracy improved considerably. However, this method requires the calculation of numerous accurate intermediate data from experimental data before training.

To solve problem (2), Zhang et al., at Shenzhen University, proposed a training strategy for phase shifting networks (PSNs) in 2021 [49]. The input of the method is a single frame of built-in interferogram, whereas the output comprises multiple phase shifting interferograms. This simulates the process of acquiring multiple images in conventional phase shifting. The speed of this method is only 65 ms, which is faster than the 160 ms of classic four-step phase shifting and slightly lower than the 30 ms of the single-frame interferogram phase extraction method. However, in terms of phase extraction RMSE, this method is only 0.11 rad, which is considerably lower than the 0.51 rad of the single-frame interferogram phase extraction method and close to the 0.09 rad of the classic four-step phase shifting. Figure 12 displays the specific method of reference [49].

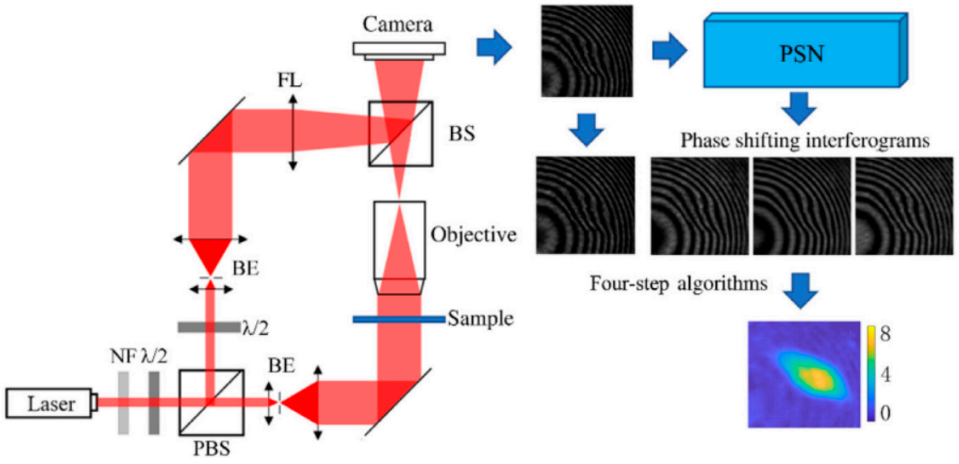


Figure 12. Method strategy from the literature [49].

4.2. Application of Deep Learning in ESPI-Wrapped Phase Image Filtering

The result obtained using the phase shifting method is an ESPI-wrapped phase map. The phase of the wrapped phase map is typically in the range $[-\pi, \pi]$ and contains considerable noise. Therefore, filtering should be performed before applying the unwrapping algorithm to obtain the full-field phase. Compared with ESPI fringe filtering, the filtering of ESPI-wrapped phase maps mainly deals with noise and false phase jumps. Classic filtering methods include the module 2π filtering method and the window Fourier filtering method based on convex optimization [19], but the following problems exist:

When severe noise is present in the ESPI-wrapped phase map, it leads to numerous false phase jumps. Conventional methods encounter challenges in handling false phase jumps during the denoising process, resulting in inaccurate unwrapping of denoised results.

(1) Conventional filtering methods are time-consuming and require considerable parameter calculation.
(2) Conventional methods have poor denoising capabilities for high-density and unevenly distributed phase maps.

To solve problem (1), Yan et al. from Shanghai University and Nanyang Technological University in Singapore used CNN to denoise the numerator and denominator of the noisy arctan function and subsequently obtain a clear wrapped phase map using the denoised numerator and denominator [50]. This method can remove noise levels with an SNR of −4 dB at a speed of only 8.4 s. Furthermore, using this method to denoise the wrapped phase map for phase extraction, the RMSE increased with the increase in the phase frequency but was less than 0.055 rad. Figure 13 displays the specific method.

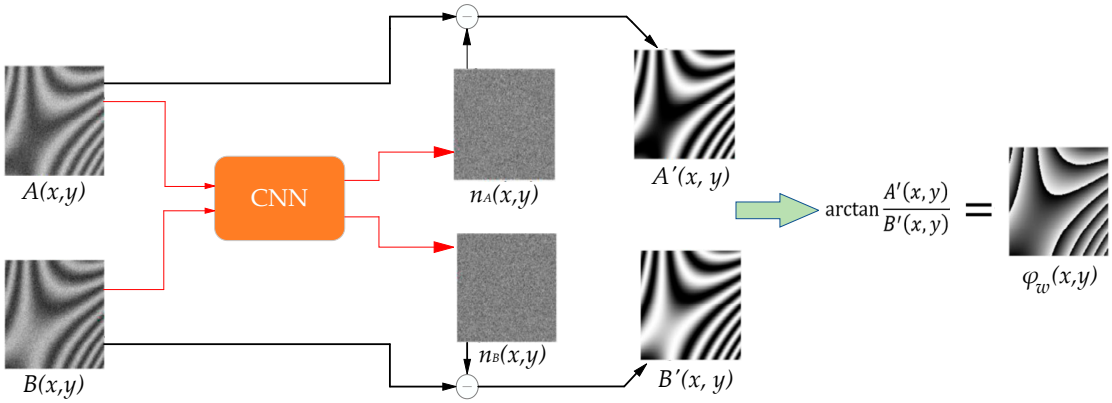


Figure 13. Schematic of the denoising method proposed in [50]. Here, (n_A (x, y) and n_B (x, y) are the results of $A(x, y)$ and $B(x, y)$ input into CNN, respectively).

To solve problem (2), in 2021, Tianjin University's Li et al. introduced the DBDNet denoising network from other image denoising fields into the field of ESPI-wrapped phase map denoising [51]. The input of this network is the noisy wrapped phase map and the denoised phase map is the output, which theoretically reduces network parameters. Compared with conventional denoising methods OCPDE, WFLPF, and deep learning method FFD-Net, this method had a PSNR of 20.5648, which is significantly higher than the values of the previous three methods, 14.0865, 12.7554, and 17.3057, respectively. However, the method does not mention specific network calculation parameters. In subsequent work, the team improved and proposed a new network DBDNet2 in 2022 to solve problem (3) [52]. In ablation experiments, the denoising effect quantitative indexes (PSNR and SSIM) of this method were 19.1034 and 0.9078, respectively, which were higher than DBDNet's 18.8819 and 0.8984, respectively. Table 3 summaries the ESPI package phase map-filtering methods based on deep learning.

Table 3. Comparison of ESPI package phase map filtering methods based on deep learning [50–52].

Research Studies	Method	Proposed Solution	Advantages	Disadvantages
Yao et al. [50]	Improved ResNet	Solves the problem of false phase jumps.	Excellent noise reduction performance; solves the problem of false phase jumps.	The model calculation parameters are too large.
Li et al. [51]	DBDNet	Solves the problem of phase denoising for large model calculation parameters and high-density and high-speckle noise.	Small number of model parameters; strong ability to recognize details.	Poor ability to obtain more comprehensive phase information.
Li et al. [52]	DBDNet2	Solves the problem of poor denoising ability for high-density noise uneven phase maps.	Excellent denoising performance and strong feature extraction ability.	May not be versatile.

Conventional ESPI image filtering methods and deep learning-based filtering methods have achieved considerable success, with their own advantages and disadvantages. Table 4 is a comparison of both methods.

Table 4. Conventional and deep learning-based filtering methods.

Method	Advantages	Disadvantages
Conventional filtering method	The physical model has strong interpretability; it does not require a large amount of training data; and some algorithms are relatively simple.	Rely on accurate physical models; sensitive to parameter adjustments; unable to handle multiple objects simultaneously.
Filtering method based on deep learning	Strong adaptability; no parameter tuning is required after network training; batch processing is possible.	The demand for a large amount of labeled data; poor interpretability.

Thus, conventional ESPI image filtering methods exhibit considerable advantages in physical model interpretability and algorithm simplicity, whereas deep learning methods have advantages in adaptability and handling complex relationships. Therefore, for datasets in various application scenarios, appropriate methods should be selected according to
the needs.

4.3. Application of Deep Learning in the Phase Unwrapping Algorithm

The phase distribution obtained after the phase shifting process is typically encapsulated in an arctangent function, as shown in Formula (17). This phase distribution is wrapped, that is, the phase value only changes within the range $[-\pi, \pi]$. This is a discontinuous phase distribution that does not match the true phase distribution. To obtain continuous and meaningful phase information, a phase unwrapping operation is required. Phase unwrapping is the process of unfolding the wrapped speckle phase into a continuous phase distribution to eliminate the periodic accumulation of phases, rendering the phase continuous and meaningful throughout the domain.

Currently, phase unwrapping methods based on deep learning can be classified into two categories, namely image segmentation-based methods and one-step direct unwrapping methods. "Segmentation" denotes determining the addition of two unknown integral multiples at each pixel in the wrapped phase to recover the true phase. One-step direct unwrapping methods learn the feature mapping between the wrapped phase map and the unwrapped image through convolutional networks.

4.3.1. Unwrapping Algorithm Based on Image Segmentation

In some cases, phase unwrapping is considered an image segmentation task. In 2018, G.E. Spoorthi et al. [53] of the Indian Institute of Technology proposed the deep learning phase unwrapping framework PhaseNet. Compared with the unwrap function in MATLAB and the quality-guided phase unwrap method (QGPU), PhaseNet has a phase MSE of 2 and a time cost of 0.18 s at an SNR of 0 dB, which is considerably better than the values obtained by the unwrap function (11 and 24 s for phase MSE and time cost, respectively) and QGPU (17 and 0.05 s, respectively). However, this method requires post-processing steps and can only unwrap phases within the range −36 to +36 radians. In 2020, the team proposed PhaseNet2.0, which increased the unwrapping range to 100 radians and eliminated the necessity for post-processing in PhaseNet [54]. Compared to QGPU, MATLAB unwrap function, least square phase unwrapping (LSPU), and calibrated least square phase unwrapping (CLSPU) methods, PhaseNet2.0 unwraps phases more smoothly on the image, with root mean square of 2.5%, which is significantly better than the values of 178.2%, 85.6%, 24.3%, and 26.3%, respectively, obtained by the former methods. Figure 14 displays the specific unwrapping effect.

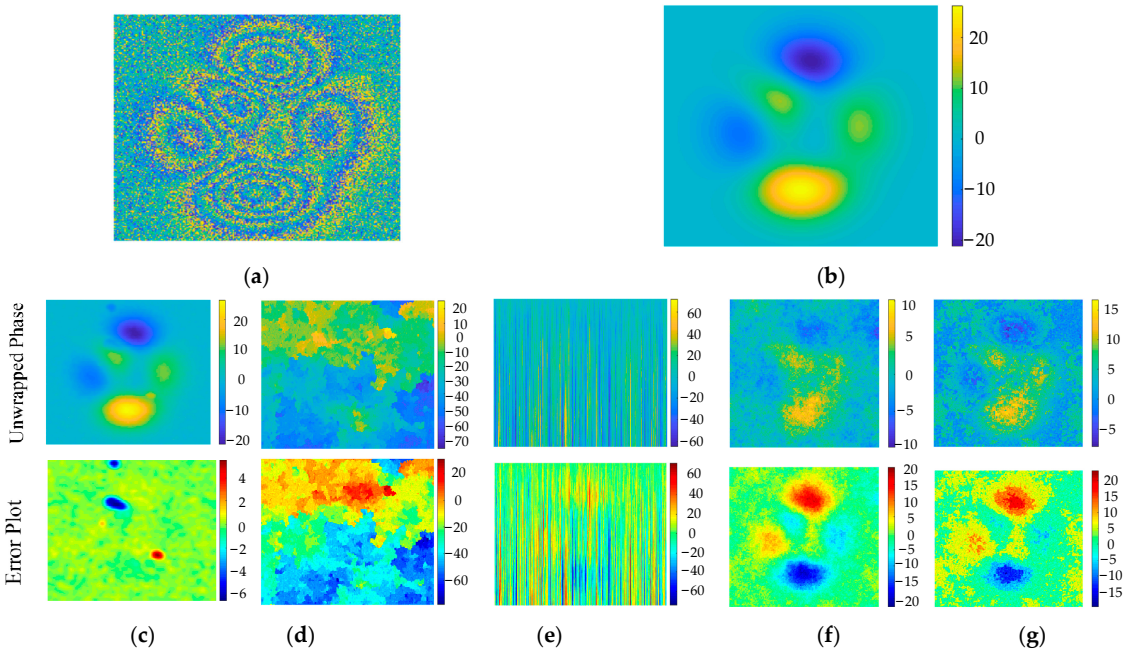


Figure 14. Comparison of the unwrapping phase method in reference [54] to other methods. (**a**) Wrapped phase at SNR = −4 dB. (**b**) Ground truth. (**c**) Reference [54]'s method. Qualitative analysis

of the proposed method at SNR = −4 dB compared to (**d**) quality-guided phase unwrapping (QGPU), (**e**) MATLAB's unwrap function, (**f**) least square phase unwrapping (LSPU), and (**g**) calibrated least square phase unwrapping (CLSPU).

Similarly, considering phase unwrapping as an image segmentation task, in 2019, Zhang et al. from Hangzhou Dianzi University used an extended DeepLabV3+ network as the backbone to solve the problem of conventional methods being prone to failure in high-noise datasets [55]. Using a Gaussian white noise dataset for phase unwrapping, the RMSE of this method is 1.2325, demonstrating considerably superior unwrapping performance compared with the conventional path-following method with an RMSE of 19.7591. In 2019, Zhang et al. from Shenyang Institute of Automation of the Chinese Academy of Sciences and the School of Optical Sciences of the University of Arizona in the United States combined CNN denoising network and CNN segmentation network to improve performance [56]. The RMSE of this method is only 0.0027, and the processing speed of the GPU is 0.22 s, which is significantly better than 6.5202 and 19.53 s of the branch cutting algorithm, and 6.2683 and 330.44 s of the QGPU method, respectively.

4.3.2. One-Step Direct Unwrapping Algorithm

Another group of scholars considered the one-step unwrapping method. In 2019, at the Northwestern Polytechnical University, Wang et al. introduced Residual Block into the U-Net network and proposed a one-step unwrapping method deep learning phase unwrapping (DLPU) [57], as displayed in Figure 15. The SSIM of the phase-unwrapped image and the reference image reached 0.991. In terms of robustness, compared with the least squares (LS) method and the quality guidance (QG) method, when the true phase height increased from 5 to 100, the SSIM index of DLPU decreased from 0.99 to 0.80, which was significantly better than those LS and QG methods (down to 0.2). The maximum error pixel ratio of DLPU is 0.2, which is 3.55 times lower than the ratios of the LS and QG methods (0.71). Subsequently, at the Beijing University of Technology in 2020, Qin et al. were inspired by VGG, U-Net, and ResNet, and proposed the VUR-Net method [58]. Xu et al. from Tianjin University [59] considered reducing the computational burden when improving the accuracy of phase unwrapping and proposed the PU-M-Net method. This approach effectively reduced speckles during the phase unwrapping process. On a high-noise dataset, PU-M-Net achieved an SSIM of 0.810 and MSE of 1.673, outperforming the LS method with 6.870 and 0.562, as well as the QGPU method with 2.585 and 0.771, respectively. Table 5 compares all the deep learning-based unwrapping methods mentioned above.

Based on the analysis, phase unwrapping methods based on image segmentation usually use simple networks and require preprocessing or post-processing to obtain reliable results. However, one-step direct unwrapping methods use more complex networks and can directly obtain reliable results.

Compared with conventional unwrapping algorithms, deep learning-based unwrapping methods exhibit a strong generalization ability, fast processing speed, and accurate phase information. However, these methods still require a large-scale dataset. Therefore, improving the accuracy and feature extraction efficiency of the network as well as reducing the requirements of the dataset are problems that need to be addressed in the current research. Addressing dataset requirements while achieving one-step direct unwrapping of the phase with high accuracy becomes a prominent topic in this field.

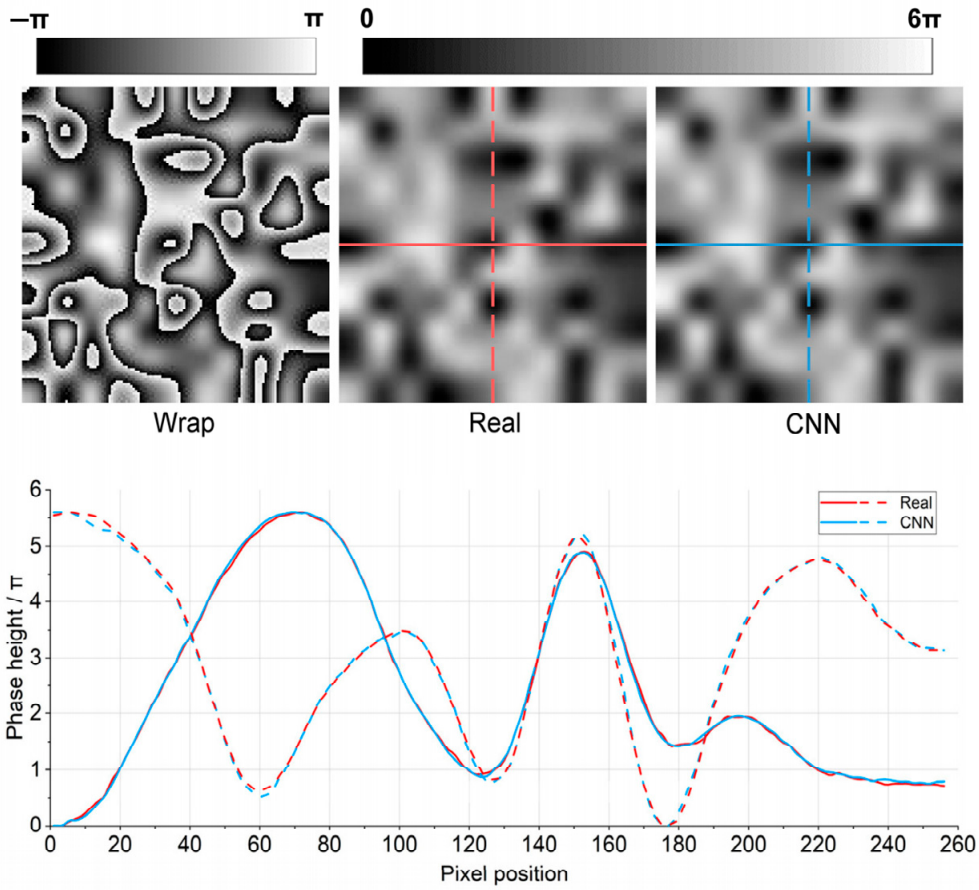


Figure 15. Comparison of the DLPU method unwrapping and phase method [57]. Upper row: the wrapped, real and CNN-output unwrapped phase images; Lower row: the comparison of phase height across the central lines indicated in real phase (red lines) and CNN-output unwrapped phase (blue lines).

Table 5. Summary of unwrapping methods based on deep learning [53–59].

Researchers	Get Phases Directly	SSIM	MSE	RMSE	Advantages	Disadvantages
G.E.Spoorthi et al. [53]	No	/	2	/	High accuracy	Requires post-processing
G.E.Spoorthi et al. [54]	No	/	/	/	High accuracy	Lack of comparison with similar methods
Zhang et al. [55]	No	/	1.5608	1.2325	High accuracy	There is a significant error in high-speckle conditions
Zhang et al. [56]	No	/	/	0.0027	It can identify phase discontinuity	Requires pre- and post-processing
Wang et al. [57]	Yes	0.991	3.150	/	Directly obtain phase and high accuracy	Limited dynamic range, and the effect at phase jump points still needs to be improved
Qin et al. [58]	Yes	0.816	1.737	/	Excellent performance	Significantly increases training costs
Xu et al. [59]	Yes	0.810	1.673	/	High computational efficiency	Lack of generalization analysis

Note: The above data are the experimental results under their respective data sets.

5. Summary and Outlook

Compared with conventional methods of phase extraction from electronic speckle interferometric images, deep learning methods have demonstrated advancements in terms

of accuracy, processing speed, and generalization. Research in this field is still in its nascent stages, and several promising avenues warrant in-depth exploration:

(1) Integration of the physical model into the deep learning model.

The generation mechanism of ESPI images is well understood, whereas deep learning models are considered "black boxes" with challenging internal mechanisms. Integrating prior knowledge with the mature generation process of ESPI images into deep learning models for different application scenarios can become a novel research direction for developing new data-driven deep learning models.

(2) Improvement and optimization of deep learning models.

Most existing deep learning models focus on the classical models such as CNN and their enhancements. With the continuous development of deep learning technology, novel models are expected to emerge. In the next phase, investigating the application of various models to phase extraction in electronic speckle interferometric images is crucial. Moreover, existing deep learning models are limited to specific steps in the phase extraction process and lack an end-to-end complete model. Therefore, introducing new deep learning models and constructing end-to-end phase extraction models will become a prominent research direction in the future.

(3) Quantitative evaluation of phase extraction quality.

Existing phase extraction methods based on deep learning models are typically applied to specific steps, such as filtering and fringe skeleton extraction. Evaluation metrics for the algorithm are confined to these intermediate steps, without focusing on the quality of phase extraction. This renders the quantitative evaluation and comparative analysis of phase extraction quality challenging. Therefore, addressing the urgent need for a quantitative evaluation of phase extraction quality is essential.

(4) Limited practical engineering applications.

At present, the ESPI phase extraction method based on deep learning is in its early stage of development, and thus is mainly only applied in the laboratory. There are only a few notable applications of these algorithms outside the laboratory, and there are currently very few practical applications in the field of industrial engineering. The application of deep learning methods to ESPI phase extraction has great potential in industrial engineering and other fields. This application may result in more efficient and accurate phase extraction methods and provide better support for industrial applications in related fields.

(5) Expanding the dataset.

Currently, limited publicly available ESPI image datasets are available in the industry, hindering the validation of the practical generalization ability of deep learning models and impeding meaningful horizontal comparisons between research outcomes. In the next step, studies should focus on generating ESPI image datasets through numerical simulation and constructing experimental systems in conjunction with practical application scenarios. A comprehensive analysis of factors influencing image quality and further dataset expansion is essential.

Author Contributions: Conceptualization, W.J. and T.R.; methodology, T.R. and Q.F.; validation, W.J. and T.R.; data curation, W.J. and T.R.; investigation, T.R. and Q.F.; writing—original draft preparation, T.R. and Q.F.; writing—review and editing, W.J.; supervision, W.J.; funding acquisition, W.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Sichuan Science and Technology Program, under grant no. 2021JDJQ0027, and the Natural Science Foundation of China, under grant no. 61875166.

Data Availability Statement: The data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

Acknowledgments: Wenbo Jiang would like to acknowledge the Sichuan Provincial Academic and Technical Leader Training Plan, and the Overseas Training Plan of Xihua University (09/2014-09/2015,

University of Michigan, Ann Arbor, US). We would also like to thank LetPub (www.letpub.com accessed on 15 January 2024) for its linguistic assistance and scientific consultation during the preparation of this manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Vaz, B.F.; Ferracini, E.; Santos, A.A.; Gonçalves, R. Measuring stress fields in connecting rods using laser interferometry (ESPI). *J. Test. Eval.* **2015**, *43*, 735–748. [CrossRef]
2. Martínez, A.; Rayas, J.A.; Rodríguez-Vera, R.; Puga, H.J. Three-dimensional deformation measurement from the combination of in-plane and out-of-plane electronic speckle pattern interferometers. *Appl. Opt.* **2004**, *43*, 4652–4658. [CrossRef] [PubMed]
3. Nitta, I.; Sato, T.; Tsukiyama, Y. 3D surface profile measurement by single shot interferometry with wide field of view. In Proceedings of the Asian Conference on Experimental Mechanics, Niigata City, Japan, 7–10 October 2019.
4. Kumar, M.; Gaur, K.K.; Shakher, C. Measurement of material constants (Young's modulus and Poisson's ratio) of polypropylene using digital speckle pattern interferometry (DSPI). *J. Jpn. Soc. Exp. Mech.* **2015**, *15*, s87–s91.
5. Kennedy, D.M.; Schauperl, Z.; Greene, S. Application of ESPI-method for strain analysis in thin wall cylinders. *Opt. Lasers Eng.* **2004**, *41*, 585–594. [CrossRef]
6. Labbe, F. Strain-rate measurements by electronic speckle-pattern interferometry (ESPI). *Opt. Lasers Eng.* **2007**, *45*, 827–833. [CrossRef]
7. Loukil, M.S.; Ayadi, Z.; Varna, J. ESPI analysis of crack face displacements in damaged laminates. *Compos. Sci. Technol.* **2014**, *94*, 80–88. [CrossRef]
8. Yang, L.; Xie, X.; Zhu, L.; Wu, S.; Wang, Y. Review of electronic speckle pattern interferometry (ESPI) for three dimensional displacement measurement. *Chin. J. Mech. Eng.* **2014**, *27*, 1–13. [CrossRef]
9. Shuhai, J.; Kaiduan, Y.; Yushan, T. The system of double-optical-path ESPI for the vibration measurement. *Opt. Lasers Eng.* **2000**, *34*, 67–74. [CrossRef]
10. Farrant, D.I.; Petzing, J.N.; Tyrer, J.R. Geometrically qualified ESPI vibration analysis of an engine. *Opt. Lasers Eng.* **2004**, *41*, 659–671. [CrossRef]
11. Li, X.; Sung, P.; Patterson, E.; Wang, W.; Christian, W. Identification of defects in composite laminates by comparison of mode shapes from electronic speckle pattern interferometry. *Opt. Lasers Eng.* **2023**, *163*, 107444. [CrossRef]
12. Ferretti, D.; Rossi, M.; Royer-Carfagni, G. An ESPI experimental study on the phenomenon of fracture in glass. Is it brittle or plastic? *J. Mech. Phys. Solids* **2011**, *59*, 1338–1354. [CrossRef]
13. Wang, K.; Song, L.; Wang, C.; Ren, Z.; Zhao, G.; Dou, J.; Di, J.; Barbastathis, G.; Zhou, R.; Zhao, J.; et al. On the use of deep learning for phase recovery. *Light Sci. Appl.* **2024**, *13*, 4. [CrossRef] [PubMed]
14. Zuo, C.; Feng, S.; Huang, L.; Tao, T.; Yin, W.; Chen, Q. Phase shifting algorithms for fringe projection profilometry: A review. *Opt. Lasers Eng.* **2018**, *109*, 23–59. [CrossRef]
15. de Groot, P. Phase-shift calibration errors in interferometers with spherical Fizeau cavities. *Appl. Opt.* **1995**, *34*, 2856–2863. [CrossRef] [PubMed]
16. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
17. Zuo, C.; Qian, J.; Feng, S.; Yin, W.; Li, Y.; Fan, P.; Han, J.; Qian, K.; Chen, Q. Deep learning in optical metrology: A review. *Light Sci. Appl.* **2022**, *11*, 39. [CrossRef] [PubMed]
18. Archbold, E.; Burch, J.M.; Ennos, A.E.; Taylor, P.A. Visual observation of surface vibration nodal patterns. *Nature* **1969**, *222*, 263–265. [CrossRef]
19. Chen, M. Research and Application of A New Method for Phase Extraction in Electronic Speckle Interferometry Based on Machine Learning. Ph.D. Thesis, Tianjin University, Tianjin, China, 2020. (In Chinese)
20. Tang, C.; Han, L.; Ren, H.; Zhou, D.; Chang, Y.; Wang, X.; Cui, X. Second-order oriented partial-differential equations for denoising in electronic-speckle-pattern interferometry fringes. *Opt. Lett.* **2008**, *33*, 2179–2181. [CrossRef]
21. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Trans. Image Process.* **2018**, *27*, 4608–4622. [CrossRef]
22. Hao, F.; Tang, C.; Xu, M.; Lei, Z. Batch denoising of ESPI fringe patterns based on convolutional neural network. *Appl. Opt.* **2019**, *58*, 3338–3346. [CrossRef]
23. Lin, B.; Fu, S.; Zhang, C.; Wang, F.; Li, Y. Optical fringe patterns filtering based on multi-stage convolution neural network. *Opt. Lasers Eng.* **2020**, *126*, 105853. [CrossRef]
24. Wang, L.; Li, R.; Tian, F.; Fang, X. Application of attention-DnCNN for ESPI fringe patterns denoising. *J. Opt. Soc. Am. A Opt. Image Sci. Vis.* **2022**, *39*, 2110–2123. [CrossRef] [PubMed]
25. Xu, M.; Tang, C.; Hong, N.; Lei, Z. MDD-Net: A generalized network for speckle removal with structure protection and shape preservation for various kinds of ESPI fringe patterns. *Opt. Lasers Eng.* **2022**, *154*, 107017. [CrossRef]
26. Gurrola-Ramos, J.; Dalmau, O.; Alarcon, T. U-Net based neural network for fringe pattern denoising. *Opt. Lasers Eng.* **2022**, *149*, 106829. [CrossRef]

27. Lam, L.; Lee, S.W.; Suen, C.Y. Thinning methodologies a comprehensive survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **1992**, *14*, 869–885. [CrossRef]
28. Quan, C.; Tay, C.J.; Yang, F.; He, X. Phase extraction from a single fringe pattern based on guidance of an extreme map. *Appl. Opt.* **2005**, *44*, 4814–4821. [CrossRef] [PubMed]
29. Tang, C.; Lu, W.; Cai, Y.; Han, L.; Wang, G. Nearly preprocessing-free method for skeletonization of gray-scale electronic speckle pattern interferometry fringe patterns via partial differential equations. *Opt. Lett.* **2008**, *33*, 183–185. [CrossRef]
30. Tang, C.; Ren, H.; Wang, L.; Wang, Z.; Han, L.; Gao, T. Oriented couple gradient vector fields for skeletonization of gray-scale optical fringe patterns with high density. *Appl. Opt.* **2010**, *49*, 2979–2984. [CrossRef]
31. Zhang, F.; Wang, D.; Xiao, Z.; Geng, L.; Wu, J.; Xu, Z.; Sun, J.; Wang, J.; Xi, J. Skeleton extraction and phase interpolation for single ESPI fringe pattern based on the partial differential equations. *Opt. Express* **2015**, *23*, 29625–29638. [CrossRef]
32. Chen, X.; Tang, C.; Li, B.; Su, Y. Gradient vector fields based on variational image decomposition for skeletonization of electronic speckle pattern interferometry fringe patterns with variable density and their applications. *Appl. Opt.* **2016**, *55*, 6893–6902. [CrossRef]
33. Li, B.; Tang, C.; Zheng, T.; Lei, Z. Fully automated extraction of the fringe skeletons in dynamic electronic speckle pattern interferometry using a U-Net convolutional neural network. *Opt. Eng.* **2019**, *58*, 023105. [CrossRef]
34. Li, B.; Li, Z.; Zhang, J.; Sun, G.; Mei, J.; Yan, J. Channel transformer U-Net: An automatic and effective skeleton extraction network for electronic speckle pattern interferometry. *Appl. Opt.* **2023**, *62*, 325–334. [CrossRef] [PubMed]
35. Liu, C.; Tang, C.; Xu, M.; Hao, F.; Lei, Z. Skeleton extraction and inpainting from poor, broken ESPI fringe with an M-net convolutional neural network. *Appl. Opt.* **2020**, *59*, 5300–5308. [CrossRef] [PubMed]
36. Wang, H.; Zhang, Z.; Zhu, Q.; Wang, X.; Dong, Z.; Men, G.; Wang, J.; Lei, J.; Wang, W. Batch skeleton extraction from ESPI fringe patterns using pix2pix conditional generative adversarial network. *Opt. Rev.* **2022**, *29*, 97–105. [CrossRef]
37. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–27 July 2017; pp. 5967–5976.
38. Parker, J.A.; Kenyon, R.V.; Troxel, D.E. Comparison of interpolating methods for image resampling. *IEEE Trans. Med. Imaging* **1983**, *2*, 31–39. [CrossRef] [PubMed]
39. Lehmann, T.M.; Gonner, C.; Spitzer, K. Survey: Interpolation methods in medical image processing. *IEEE Trans. Med. Imaging* **1999**, *18*, 1049–1075. [CrossRef] [PubMed]
40. Maeland, E. On the comparison of interpolation methods. *IEEE Trans. Med. Imaging* **1988**, *7*, 213–217. [CrossRef]
41. Tang, C.; Lu, W.; Chen, S.; Zhang, Z.; Li, B.; Wang, W.; Han, L. Denoising by coupled partial differential equations and extracting phase by backpropagation neural networks for electronic speckle pattern interferometry. *Appl. Opt.* **2007**, *46*, 7475–7484. [CrossRef]
42. Wang, G.; Li, Y.J.; Zhou, H.C. Application of the radial basis function interpolation to phase extraction from a single electronic speckle pattern interferometric fringe. *Appl. Opt.* **2011**, *50*, 3110–3117. [CrossRef]
43. Huang, P.S.; Zhang, S. Fast three-step phase-shifting algorithm. *Appl. Opt.* **2006**, *45*, 5086–5091. [CrossRef]
44. Schwider, J.; Falkenstoerfer, O.R.; Schreiber, H.; Zoeller, A.; Streibl, N. New compensating four-phase algorithm for phase-shift interferometry. *Opt. Eng.* **1993**, *32*, 1883–1885. [CrossRef]
45. Novak, J. Five-step phase-shifting algorithms with unknown values of phase shift. *Optik* **2003**, *114*, 63–68. [CrossRef]
46. Ponslet, E.; Steinzig, M. Residual stress measurement using the hole drilling method and laser speckle interferometry part III: Analysis technique. *Exp. Tech.* **2003**, *27*, 45–48. [CrossRef]
47. Bi, H.; Zhang, Y.; Ling, K.V.; Wen, C. Class of 4 + 1-phase algorithms with error compensation. *Appl. Opt.* **2004**, *43*, 4199–4207. [CrossRef] [PubMed]
48. Feng, S.; Chen, Q.; Gu, G.; Tao, T.; Zhang, L.; Hu, Y.; Yin, W.; Zuo, C. Fringe pattern analysis using deep learning. *Adv. Photonics* **2019**, *1*, 025001. [CrossRef]
49. Zhang, Q.; Lu, S.; Li, J.; Li, D.; Lu, X.; Zhong, L.; Tian, J. Phase-shifting interferometry from single frame in-line interferogram using deep learning phase-shifting technology. *Opt. Commun.* **2021**, *498*, 127226. [CrossRef]
50. Yan, K.; Yu, Y.; Sun, T.; Asundi, A.; Kemao, Q. Wrapped phase denoising using convolutional neural networks. *Opt. Lasers Eng.* **2020**, *128*, 105999. [CrossRef]
51. Li, J.; Tang, C.; Xu, M.; Fan, Z.; Lei, Z. DBDNet for denoising in ESPI wrapped phase patterns with high density and high speckle noise. *Appl. Opt.* **2021**, *60*, 10070–10079. [CrossRef]
52. Li, J.; Tang, C.; Xu, M.; Lei, Z. Uneven wrapped phase pattern denoising using a deep neural network. *Appl. Opt.* **2022**, *61*, 7150–7157. [CrossRef]
53. Spoorthi, G.E.; Gorthi, S.; Gorthi RK, S.S. PhaseNet: A deep convolutional neural network for two-dimensional phase unwraping. *IEEE Signal Process. Lett.* **2018**, *26*, 54–58. [CrossRef]
54. Spoorthi, G.E.; Gorthi RK, S.S.; Gorthi, S. PhaseNet 2.0: Phase unwrapping of noisy data based on deep learning approach. *IEEE Trans. Image Process.* **2020**, *29*, 4862–4872. [CrossRef]
55. Zhang, T.; Jiang, S.; Zhao, Z.; Dixit, K.; Zhou, X.; Hou, J.; Zhang, Y.; Yan, C. Rapid and robust two-dimensional phase unwrapping via deep learning. *Opt. Express* **2019**, *27*, 23173–23185. [CrossRef] [PubMed]
56. Zhang, J.; Tian, X.; Shao, J.; Luo, H.; Liang, R. Phase unwrapping in optical metrology via denoised and convolutional segmentation networks. *Opt. Express* **2019**, *27*, 14903–14912. [CrossRef] [PubMed]

57. Wang, K.; Li, Y.; Kemao, Q.; Di, J.; Zhao, J. One-step robust deep learning phase unwrapping. *Opt. Express* **2019**, *27*, 15100–15115. [CrossRef]
58. Qin, Y.; Wan, S.; Wan, Y.; Weng, J.; Liu, W.; Gong, Q. Direct and accurate phase unwrapping with deep neural network. *Appl. Opt.* **2020**, *59*, 7258–7267. [CrossRef]
59. Xu, M.; Tang, C.; Shen, Y.; Hong, N.; Lei, Z. PU-M-Net for phase unwrapping with speckle reduction and structure protection in ESPI. *Opt. Lasers Eng.* **2022**, *151*, 106824. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

Electronics Editorial Office
E-mail: electronics@mdpi.com
www.mdpi.com/journal/electronics

Disclaimer/Publisher's Note: The title and front matter of this reprint are at the discretion of the Guest Editors. The publisher is not responsible for their content or any associated concerns. The statements, opinions and data contained in all individual articles are solely those of the individual Editors and contributors and not of MDPI. MDPI disclaims responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-4372-5