agriculture

Special Issue Reprint

Application of Vision Technology and Artificial Intelligence in Smart Farming

Edited by
Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

mdpi.com/journal/agriculture

MDPI

Application of Vision Technology and Artificial Intelligence in Smart Farming

Application of Vision Technology and Artificial Intelligence in Smart Farming

Editors

Xiuguo Zou
Zheng Liu
Xiaochen Zhu
Wentian Zhang
Yan Qian
Yuhua Li

MDPI

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

Editors

Xiuguo Zou
Nanjing Agricultural University
Nanjing, China

Zheng Liu
University of British Columbia
Kelowna, BC, Canada

Xiaochen Zhu
Nanjing University of Information Science and Technology
Nanjing, China

Wentian Zhang
University of Technology Sydney
Sydney, NSW, Australia

Yan Qian
Nanjing Agricultural University
Nanjing, China

Yuhua Li
Nanjing Agricultural University
Nanjing, China

Editorial Office
MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Agriculture* (ISSN 2077-0472) (available at: https://www.mdpi.com/journal/agriculture/special_issues/6T02FN367V).

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 978-3-03928-597-6 (Hbk)
ISBN 978-3-03928-598-3 (PDF)
doi.org/10.3390/books978-3-03928-598-3

Contents

About the Editors

Xiuguo Zou

Dr. Xiuguo Zou received his Ph.D. in Agricultural Bio-Environment and Energy Engineering from Nanjing Agricultural University in 2013. Since 2013, he has been an associate professor at Nanjing Agricultural University. He is a senior member of the China Electronics Society and also a member of the following organizations: the IEEE, United States; the Rural Professional Technology Association of Jiangsu Province; the Agricultural Automation Special Committee of Jiangsu Automation Society; the Facility Agricultural Equipment Branch of Jiangsu Agricultural Society; and the Artificial Intelligence Society of Jiangsu Province. His current research mainly focuses on agricultural electronics and information technology, machine vision and image processing, agricultural biological environment control and equipment, and agricultural big data technology.

Zheng Liu

Dr. Zheng Liu received his Doctorate in Engineering from Kyoto University (Kyoto, Japan) in 2000 and earned a second Ph.D. from the University of Ottawa in 2007. From 2000 to 2001, he was a research fellow at Nanyang Technological University (Singapore). He then joined the National Research Council (NRC) of Canada (Ottawa, Ontario) as a Governmental Laboratory Visiting Fellow nominated by the NSERC in 2001. In 2002, he became a research officer associated with two research institutes of NRC (Aerospace Research (IAR) and Research in Construction (IRC)). From 2012 to 2015, Dr. Liu worked as a full professor for Toyota Technological Institute in Nagoya, Japan. In August 2015, Dr. Liu joined the University of British Columbia (Okanagan campus) in Kelowna, BC, Canada. His research interests include non-destructive inspection and evaluation, condition assessment, predictive maintenance, data/information fusion, computer/machine vision, pattern recognition, machine learning, and sensor/sensor networks, digital twins, and digital twin computing. Dr. Liu is a fellow of the SPIE and a senior member of the IEEE. He has served as the vice president for publication at the IEEE Instrumentation and Measurement Society (2016–2017) and co-chairs the IEEE IMS TC-1. He holds professional engineer licenses in both Ontario and British Columbia. Dr. Liu also serves on the editorial board of a number of peer-reviewed journals.

Xiaochen Zhu

Dr. Xiaochen Zhu received his Ph.D. degree from the School of Geography of Nanjing University of Information Science and Technology, Nanjing, China, in 2017. He works at Nanjing University of Information Science and Technology. His research interests include refined meteorological services, meteorological disaster prevention and reduction, and AI meteorology.

Wentian Zhang

Dr. Wentian Zhang received his Ph.D. degree from the University of Technology Sydney, Sydney, Australia, in 2020. He was a postdoctoral research associate with Prof. Steven Su at the University of Technology Sydney. His research interests include electronic nose systems, machine learning, signal processing, and medical devices.

Yan Qian

Dr. Yan Qian received her Ph.D. in Agricultural Electrification and Automation Engineering from Nanjing Agricultural University in 2014. Since 2006, she has been a lecturer at Nanjing Agricultural University, College of Engineering, and is now an associate professor at Nanjing

Agricultural University, College of Artificial Intelligence. Her research areas include agricultural robots, computer vision technology, and virtual reality 3D reconstruction.

Yuhua Li

Dr. Yuhua Li received her Ph.D. in Information and Communication Engineering from Xi'an Jiaotong University, Xi'an, China, in 2015. Since 2015, she has been a university lecturer at the Department of Electronic Information of Nanjing Algaculture University, Nanjing, China. Over the past decade, she has published over 20 papers spanning a range of theoretical and applied problems in machine learning and computer vision. Her research has been funded by the National Natural Science Foundation of China and the Fundamental Research Funds for the Central Universities. Her current research interests include image processing and pattern recognition, animal and plant phenotyping, and the applications of computer vision in algaculture.

agriculture

MDPI

Editorial

Application of Vision Technology and Artificial Intelligence in Smart Farming

Xiuguo Zou [1,2,*], Zheng Liu [2], Xiaochen Zhu [3], Wentian Zhang [4], Yan Qian [1] and Yuhua Li [1]

[1] College of Artificial Intelligence, Nanjing Agricultural University, Nanjing 210031, China
[2] Faculty of Applied Science, University of British Columbia, Kelowna, BC V1V 1V7, Canada
[3] School of Applied Meteorology, Nanjing University of Information Science and Technology, Nanjing 210044, China
[4] Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, NSW 2007, Australia
* Correspondence: zouxiuguo@njau.edu.cn or xzou05@mail.ubc.ca; Tel.: +86-25-58606585

With the rapid advancement of technology, traditional farming is gradually transitioning into smart farming. Smart farming is an agricultural production system that harnesses modern technologies such as artificial intelligence (AI), big data, and automation technology, which assists farmers in effectively managing and optimizing the production process through data and image analysis. The ultimate goal of smart farming is to achieve precision agriculture and enhance efficiency and quality in agricultural production.

This Special Issue focuses on the application of visual technology and artificial intelligence in smart farming. Researchers from Asia and Europe have contributed a total of fourteen papers, including thirteen articles and one review, to this issue. The key information for each paper is shown in Table 1. These papers encompass a broad spectrum of technologies, such as image recognition algorithms, machine learning techniques, remote sensing technology, and 3D point cloud technology. The objective is to employ these technologies for monitoring the phenotypes of plants and animals, as well as their growth environments. By offering theoretical and technical support, personalized agricultural management solutions are made accessible to farmers.

Machine learning is a widely used modeling technique that leverages large quantities of data to acquire knowledge and make predictions. It can be applied to forecast trends in the growth environment of animals and plants. For example, Ren et al. [1] developed a prediction model for relative soil moisture (RHs 10 cm) in the 0–10 cm soil layer using the extreme gradient boosting (XGBoost) algorithm based on atmospheric and soil factors, which is capable of reasonably predicting the development process of drought events. Analyzing and predicting the growth environment of crops using data analysis methods can provide preventive measures for potential hazards. This method is equally effective for poultry farming.

Deep learning is a specialized machine learning field based on artificial neural networks. It involves learning and training through multi-layered neural networks to extract complex features and patterns from data, enabling more precise predictions. It is particularly suitable for handling complex, large-scale data and high-dimensional features and has wide applications in the field of computer vision. Applying deep learning models to analyze RGB images, remote sensing images, and 3D point cloud data in agriculture can enable more accurate monitoring of phenotypes of animals and plants, facilitating precision farming management. Jiao et al. [2] developed predictive models for the initial flowering period of Platycladus orientalis (Chinese thuja) using recurrent neural networks (RNN), long short-term memory networks (LSTM), and gated recurrent units (GRU). Shapely Additive Explanation (SHAP) was used to analyze the contribution rates of meteorological factors. The accuracy of all three models was significantly higher than that of a regression model based on the accumulated temperature. Among them, the GRU model performed

Citation: Zou, X.; Liu, Z.; Zhu, X.; Zhang, W.; Qian, Y.; Li, Y. Application of Vision Technology and Artificial Intelligence in Smart Farming. *Agriculture* **2023**, *13*, 2106. https://doi.org/10.3390/agriculture13112106

Received: 10 October 2023
Revised: 18 October 2023
Accepted: 31 October 2023
Published: 6 November 2023

the best, with an average accuracy exceeding 98%. Guo et al. [3] proposed an improved YOLOv5 object detection model, integrating the coordinate attention module and the deformable convolution module for accurately detecting mature Zanthoxylum on a mobile picking platform, addressing the issues of irregular shape and occlusion caused by branches and leaves. Li et al. [4] proposed a lightweight wheat growth stage detection model and a dynamic migration algorithm, which utilizes edge computing to migrate the detection model to the wireless network edge server for processing, improving efficiency significantly compared to the local implementation. By accurately monitoring the growth trends of animals and plants through deep learning and computer vision technologies, they can effectively improve production efficiency. In addition, this technology can also be applied to classification tasks to achieve personalized management of the same type of subjects. Zhang et al. [5] proposed a method for identifying individual dairy cattle in large-scale dairy farms. They used the DeepOtsu model to binarize the body pattern image for primary classification and the EfficientNet-B1 model for secondary classification, and the overall identification accuracy reached 98.5%. Cui et al. [6] proposed an improved CNN-LSTM model for classifying high-yield and low-yield cow udders that have undergone fine-grained segmentation using the SOLOv2 method, which could allocate them to different production groups.

Table 1. The key information for each paper.

Authors	Objects	Models	Contributions
Ren et al. [1]	Soil Moisture	Extreme Gradient Boosting	Establish prediction models of soil relative humidity
Jiao et al. [2]	Platycladus Orientalis	Gated Recurrent Unit	Predict the initial flowering period
Guo et al. [3]	Zanthoxylum	YOLOv5	Identify mature Zanthoxylum fruits
Li et al. [4]	Wheat	A Lightweight CNN	Identify wheat growth stages
Zhang et al. [5]	Cow	DeepOtsu EfficientNet	Identify individual cows
Cui et al. [6]	Cow	SOLOv2 CNN-LSTM	Divide cow production groups
Ding et al. [7]	Apple	RFCA ResNet	Identify apple leaf diseases
Hao et al. [8]	Hens	Faster R-CNN	Monitor the feeding behavior of hens
Lee et al. [9]	Honey Bee	BFMatcher	Monitor bee mites and diseases
Yu et al. [10]	Grain	FcsNet	Recognize grain pest species
He et al. [11]	Rice Seed	Multimodal Fusion	Classify rice varieties
Sun et al. [12]	Soybean	RandLA-Net BAAF-Net	Construct an annotated three-dimensional model dataset
Xu et al. [13]	Maize	Multi-view Registration Algorithm Iterative Nearest Point Algorithm	Realize early variety selection at the seedling stage
Karunathilake et al. [14]	Multi Objects	Multi Models	A review of the latest advances in precision agriculture

Furthermore, visual technology can detect diseases based on the phenotypic information of animals and plants, providing early warnings for farmers to reduce economic losses. Therefore, the accuracy of model recognition is particularly important for the practical application. Several authors in this issue have researched model improvement. Ding et al. [7] proposed a novel model called RFCA ResNet, which incorporates multi-scale feature extraction and a dual attention mechanism for classifying and recognizing apple leaf diseases. Additionally, the adverse effects of imbalanced datasets on classification accuracy were effectively minimized using the class balance technique in conjunction with focal loss. Hao et al. [8] proposed an improved Faster R-CNN network characterized by the fusion of a 101 layers-deep residual network (ResNet101) and Path Aggregation Network (PAN) for monitoring the feeding behavior of hens, and the ability of the model to extract features is greatly enhanced according to the visualization results of the feature map output by the convolutional layer at each stage of the network. Lee et al. [9] proposed

an image-processing method based on a keypoint detection algorithm and image-matching algorithms for detecting small-sized honey mites attached to bees, which can result in economic losses. Additionally, they employed Contrast Limited Adaptive Histogram Equalization (CLAHE) based on the RGB color model to enhance image quality. Their method demonstrated effective performance when applied to the measured 300 mm data. Yu et al. [10] proposed a stored grain pest identification method based on a triple-attention module (FCS), namely, frequency domain attention (FAM), channel attention (CAM), and spatial attention (SAM) to solve pest-detection and segmentation tasks.

However, the improvement in the recognition accuracy of models relying solely on single-image information is limited, making it difficult to capture image features and abstract concepts in complex tasks, which leads to less accurate or complete processing results. Some models require a large amount of annotated data for training, and the lack of sufficient data can affect the effectiveness and generalization ability. When image processing techniques are combined with other technologies, including multidimensional images, data fusion, etc., it enables more precise monitoring of subjects in farming. He et al. [11] proposed a novel decision-making method based on a multimodal fusion detection model, and multiple models were used to predict the rice seed varieties according to 2D images and 3D point cloud datasets to calculate a comprehensive score vector. Finally, the predicted probabilities from 2D and 3D were jointly weighted to obtain the final predicted probability, which could combine the advantages of different data modalities and significantly improve the final prediction results. Sun et al. [12] used multi-view stereoscopic technology (MVS) to reconstruct the entire growth period (13 stages) of five different soybean varieties in three dimensions, constructed a 3D dataset named Soybean-MVS with the labels of the entire soybean growth period, and used RandLA-Net and BAAF-Net two point cloud semantic segmentation models to verify its usability, which can provide usable basic data support for the 3D crop model segmentation models. Xu et al. [13] proposed a reconstruction algorithm based on 3D information for the detection of maize phenotypic traits, utilizing a multi-view registration algorithm and iterative closest point (ICP) algorithm for the global 3D reconstruction of maize seedling populations, which contributes to precise and intelligent early management of maize.

The works received for this Special Issue demonstrate the feasibility of applying artificial intelligence and visual technologies in smart farming. Karunathilake et al. [14] provide a comprehensive overview of the recent innovations in smart farming technology, such as drones, sensors, and automation. The agricultural environment is practically diverse and challenging, with variations in soil conditions, climate patterns, and so on. To develop effective models for smart farming, it is essential to create algorithms that can generalize well across different environments. Moreover, the health and growth of plants and animals are influenced by a range of factors, so the integration of multimodal data and the fusion of multiple features hold great potential for improving the accuracy of predictions and identifications in smart farming applications. Despite the potential benefits of smart farming technology, there are challenges to promoting and popularizing its application. The technology must not only meet practical application requirements, but also address issues such as cost and farmer acceptance. On the one hand, hardware, software, and maintenance costs should be affordable for small and medium-sized farmers. On the other hand, the widespread implementation of smart farming technology requires a strong digital infrastructure and connectivity. In many rural areas, the lack of reliable internet and mobile networks can hinder the deployment of smart farming solutions, and this issue may be addressed well by cloud-edge coordinated computing, achieving more efficient computing and data processing. This Special Issue contains papers related to the above innovative research and will hopefully stimulate further research in these areas.

Acknowledgments: We are thankful to Sunyuan Wang, Qiuhong Zhang, Xinyu Zhang and Tao Liu, who have contributed to helping with paper information sorting.

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

References

1. Ren, Y.; Ling, F.; Wang, Y. Research on provincial-level soil moisture prediction based on extreme gradient boosting model. *Agriculture* **2023**, *13*, 927. [CrossRef]
2. Jiao, G.; Shentu, X.; Zhu, X.; Song, W.; Song, Y.; Yang, K. Utility of deep learning algorithms in initial flowering period prediction models. *Agriculture* **2022**, *12*, 2161. [CrossRef]
3. Guo, J.; Xiao, X.; Miao, J.; Tian, B.; Zhao, J.; Lan, Y. Design and experiment of a visual detection system for zanthoxylum-harvesting robot based on improved YOLOv5 model. *Agriculture* **2023**, *13*, 821. [CrossRef]
4. Li, Y.; Liu, H.; Wei, J.; Ma, X.; Zheng, G.; Xi, L. Research on winter wheat growth stages recognition based on mobile edge computing. *Agriculture* **2023**, *13*, 534. [CrossRef]
5. Zhang, R.; Ji, J.; Zhao, K.; Wang, J.; Zhang, M.; Wang, M. A cascaded individual cow identification method based on DeepOtsu and EfficientNet. *Agriculture* **2023**, *13*, 279. [CrossRef]
6. Cui, G.; Qiao, L.; Li, Y.; Chen, Z.; Liang, Z.; Xin, C.; Xiao, M.; Zou, X. Division of cow production groups based on SOLOv2 and Improved CNN-LSTM. *Agriculture* **2023**, *13*, 1562. [CrossRef]
7. Ding, J.; Zhang, C.; Cheng, X.; Yue, Y.; Fan, G.; Wu, Y.; Zhang, Y. Method for classifying apple leaf diseases based on dual attention and multi-scale feature extraction. *Agriculture* **2023**, *13*, 940. [CrossRef]
8. Hao, H.; Fang, P.; Jiang, W.; Sun, X.; Wang, L.; Wang, H. Research on laying hens feeding behavior detection and model visualization based on convolutional neural network. *Agriculture* **2022**, *12*, 2141. [CrossRef]
9. Lee, H.; Kim, M.; Kim, S.; Lee, S.; Lee, H.; Sin, J.; Mo, C. Identifying an image-processing method for detection of bee mite in honey bee based on keypoint analysis. *Agriculture* **2023**, *13*, 1511. [CrossRef]
10. Yu, J.; Shen, Y.; Liu, N.; Pan, Q. Frequency-enhanced channel-spatial attention module for grain pests classification. *Agriculture* **2022**, *12*, 2046. [CrossRef]
11. He, X.; Cai, Q.; Zou, X.; Feng, X.; Yin, W.; Qian, Y. Multi-modal late fusion rice seed variety classification based on an improved voting method. *Agriculture* **2023**, *13*, 597. [CrossRef]
12. Sun, Y.; Zhang, Z.; Sun, K.; Li, S.; Yu, J.; Miao, L.; Zhang, Z.; Li, Y.; Zhao, H.; Hu, Z.; et al. Soybean-MVS: Annotated three-dimensional model dataset of whole growth period soybeans for 3D plant organ segmentation. *Agriculture* **2023**, *13*, 1321. [CrossRef]
13. Xu, N.; Sun, G.; Bai, Y.; Zhou, X.; Cai, J.; Huang, Y. Global reconstruction method of maize population at seedling stage based on Kinect sensor. *Agriculture* **2023**, *13*, 348. [CrossRef]
14. Karunathilake, E.; Le, A.; Heo, S.; Chung, Y.; Mansoor, S. The path to smart farming: Innovations and opportunities in precision agriculture. *Agriculture* **2023**, *13*, 1593. [CrossRef]

agriculture

MDPI

Article

Division of Cow Production Groups Based on SOLOv2 and Improved CNN-LSTM

Guanying Cui [1,†], Lulu Qiao [1,†], Yuhua Li [1,*], Zhilong Chen [1], Zhenyu Liang [1], Chengrui Xin [2], Maohua Xiao [2] and Xiuguo Zou [1,3]

[1] College of Artificial Intelligence, Nanjing Agricultural University, Nanjing 210031, China; 9203020903@stu.njau.edu.cn (G.C.); 19220117@stu.njau.edu.cn (L.Q.); 2021819069@stu.njau.edu.cn (Z.C.); 2022819080@stu.njau.edu.cn (Z.L.); xzou05@mail.ubc.ca (X.Z.)
[2] College of Engineering, Nanjing Agricultural University, Nanjing 210031, China; 2021112042@stu.njau.edu.cn (C.X.); xiaomaohua@njau.edu.cn (M.X.)
[3] Faculty of Applied Science, University of British Columbia, Kelowna, BC V1V 1V7, Canada
* Correspondence: lyhresearch@njau.edu.cn; Tel.: +86-25-58606585
† These authors contributed equally to this work.

Abstract: Udder conformation traits interact with cow milk yield, and it is essential to study the udder characteristics at different levels of production to predict milk yield for managing cows on farms. This study aims to develop an effective method based on instance segmentation and an improved neural network to divide cow production groups according to udders of high- and low-yielding cows. Firstly, the SOLOv2 (Segmenting Objects by LOcations) method was utilized to finely segment the cow udders. Secondly, feature extraction and data processing were conducted to define several cow udder features. Finally, the improved CNN-LSTM (Convolution Neural Network-Long Short-Term Memory) neural network was adopted to classify high- and low-yielding udders. The research compared the improved CNN-LSTM model and the other five classifiers, and the results show that CNN-LSTM achieved an overall accuracy of 96.44%. The proposed method indicates that the SOLOv2 and CNN-LSTM methods combined with analysis of udder traits have the potential for assigning cows to different production groups.

Keywords: cow udder classification; udder features; instance segmentation; CNN-LSTM; udder conformation

Citation: Cui, G.; Qiao, L.; Li, Y.; Chen, Z.; Liang, Z.; Xin, C.; Xiao, M.; Zou, X. Division of Cow Production Groups Based on SOLOv2 and Improved CNN-LSTM. *Agriculture* **2023**, *13*, 1562. https://doi.org/10.3390/agriculture13081562

Academic Editor: Michael Schutz

Received: 6 June 2023
Revised: 30 July 2023
Accepted: 1 August 2023
Published: 4 August 2023

1. Introduction

Milk and cow products are essential foods for daily life, which provide vital proteins required by humans [1]. Dairy is integral to China's modern agriculture and food industry and indispensable for a healthy China. With the improvement in quality of life, the scale of the dairy industry, milk production, and milk consumption are increasing. Cow breeding is the first step in the milk industry chain and is a prerequisite for obtaining high-quality milk production. In recent years, with economic growth, cow farming in China has gradually shifted from a traditional family-based mode to an intensive, large-scale, and facility-based mode [2]. However, there are still areas for improvement in farming management techniques because cows with different levels of milk production are often managed similarly by farmers, who are unable to manage high-yielding cows based on their characteristics, which affects milk yield and quality. Therefore, a reasonable grouping of cows in production areas based on milk production and the formulation of corresponding management practices for different production areas, such as forage to concentrate ratios and exercise levels, are important to promote the development of the milk industry in China.

Numerous studies have found a correlation between milk production and udder traits in cows. Pawlina et al. [3] found an increase in udder and teat size and a decrease in

udder distance from the floor between the first and third lactation in high-yielding cows. Okkema et al. [4] found that swollen teats in cows with edematous udders reduced milk production. Juozaitiene et al. [5] evaluated morphological indicators of cow udders and measured an increase in milk production of 2.72–3.01 kg in cows with a pelvic shape compared to cows with a round udder under the action of the milking machine, indicating that milk production was associated with cow udder shape. Miseikiene et al. [6] analyzed cows' milk production in different lactation zones. After measuring, cows produced about 4.6 kg (42.2%) of milk in the anterior lactation area and 6.32 kg (57.8%) in the posterior lactation area, indicating a correlation between relative udder capacity and milk production. The research problem is whether the cow production groups could be assigned according to udder characteristics.

Feature extraction from the udder is vital for the analysis of udder traits. Recent domestic and foreign researches have divided udder measurement methods into two main categories. The first category uses manual measurement methods, and the second category uses computer vision techniques to extract cow udder traits. The first category method usually uses tools such as a body ruler [7], aluminum foil [8], and a dynamometer [9,10] for udder traits extraction. However, it is time-consuming. The second is the extraction of cow feature points, which can be realized in several ways. For example, feature point labeling is performed manually [11,12], template matching with images with standard feature points is utilized to obtain feature points [13], and contour maps are obtained from 3D point clouds to compute feature points [14]. Finally, it calculates the cow eigenvalues from the eigen points. Contrasted with manual measurement, it is more automatic and effective [15]. However, the selection and number of feature points greatly impact the calculation of feature values, so there is a certain error between those points and the true feature values. Therefore, the aim of this study is to automate the extraction and analysis of udder features using computer vision, deep learning, and other technologies, and to explore the efficiency of udder features in classifying high- and low-yielding cows.

Nowadays, with the continuous innovation and development of artificial intelligence technology, instance segmentation algorithms can achieve mask segmentation of target objects [16–19]. The neural network can fit nonlinear relationships to analyze and predict unknown data attributes [20] of production groups. Therefore, we discuss the importance of instance segmentation algorithms (SOLOv2, Mask R-CNN (Region-based Convolutional Neural Network) [21,22]), as well as neural network algorithms (CNN-LSTM, BPNN (Back Propagation Neural Network)), for the division of high- and low-yielding cows.

Our objectives were to construct an udder segmentation model to extract targets from the image; realize udder feature extraction, analyze high- and low-yielding udder features, and explore the most suitable classification features; select appropriate classification methods to explore the effectiveness of udder features in cow classification; and apply the constructed scheme to the dairy farm to achieve the division of production groups and provide support for zoning management.

2. Materials and Methods

2.1. Cow Video Acquisition

The data for this study were collected in February 2023 at a 1000-cow farm owned by Jiangsu Yuhang Food Technology Co., Ltd., Yancheng, China, a large modern cow farm, in Bailin Village, a southwest suburb of Dongtai City, Yancheng City, Jiangsu Province. There were several passages inside the experimental site with a width of about 2.5 m and cow living areas on both sides of the passages. The cow farming areas were divided into high- and low-yielding areas based on agricultural experts, considering the factors of milk production, parity, and cow condition. The reference standard for milk production is that a cow producing more than 9000 kg of milk in a lactation (305 d) is a high-yielding cow and the rest are low-yielding cows (excluding unproductive cows). The cows' age ranged from around two to eight years old (excluding unproductive cows), in height from 130 to 145 cm and in weight from 550 to 750 kg. The bedding is sorted daily and changed monthly.

Based on the location of the cow's udder in the body region and the measurement method of udder characteristics, this study used a self-designed dairy farm inspection robot to collect images of different cows in the high- and low-production groups to reduce cow stress and improve image quality.

The cow farm inspection robot comprises a mobile chassis, a lifting bar, an industrial camera, a Jestson Nano, and corresponding control components. The mobile chassis refers to a modern automobile drive and steering structure, with DC (direct current) brush motors providing the driving force and digital servos controlling the steering. The wheels are 180 mm solid rubber wheels, of which the two rear wheels are the driving wheels to drive the chassis movement, and the two front wheels are the driven wheels to control the chassis steering. The mobile chassis adopts the SLAM (Simultaneous Localization and Mapping) algorithm, which can realize laser map building and autonomous navigation. The lift rod is a DC electric actuator with a stroke of 500 mm and a maximum height of 1160 mm. Relays control the direction of lift rod movement, which can meet the demand for udder height shooting. An industrial camera is mounted on top of the lift rod to capture the side udder image of the cow, with an image size of 640 × 640 and a frame rate of 30 fps. Then, the image is transmitted to the cloud platform via Jetson Nano. The robot body structure is based on a modern car body and was produced using 3D printing technology. During image acquisition, the robot inspects the passage, keeping the same distance from the cow and moving in the direction parallel to the cow's side, continuously captures the cow's udder side image, and uploads the video to the AliCloud OSS (Object Storage Service) object storage platform for data cloud transmission and storage. The actual view of the device on the cow farm is shown in Figure 1.

Figure 1. View of inspection robot in operation.

2.2. Keyframe Extraction

Since the cow images are acquired by intercepting the video taken by the inspection robot at a specific frame rate, considering the slow movement of the cows and the inspection robot, it may result in a certain amount of duplicate images. Therefore, this study proposes a method to extract keyframes from the video, which segments the video sequence with shots to obtain the distinct features of the images, and then extracts the critical information from the video to increase the amount of information in the dataset and reduce the redundancy.

This study uses the inter-frame difference method based on local maxima to extract keyframes, judging the changing size between adjacent images by differencing two adjacent frames according to the average pixel intensity. Then, the image with a large change compared to the previous image is extracted, which is the keyframe. The extracted before and after keyframes are shown in Figure 2. Based on a reasonable threshold, cows with different features can be obtained.

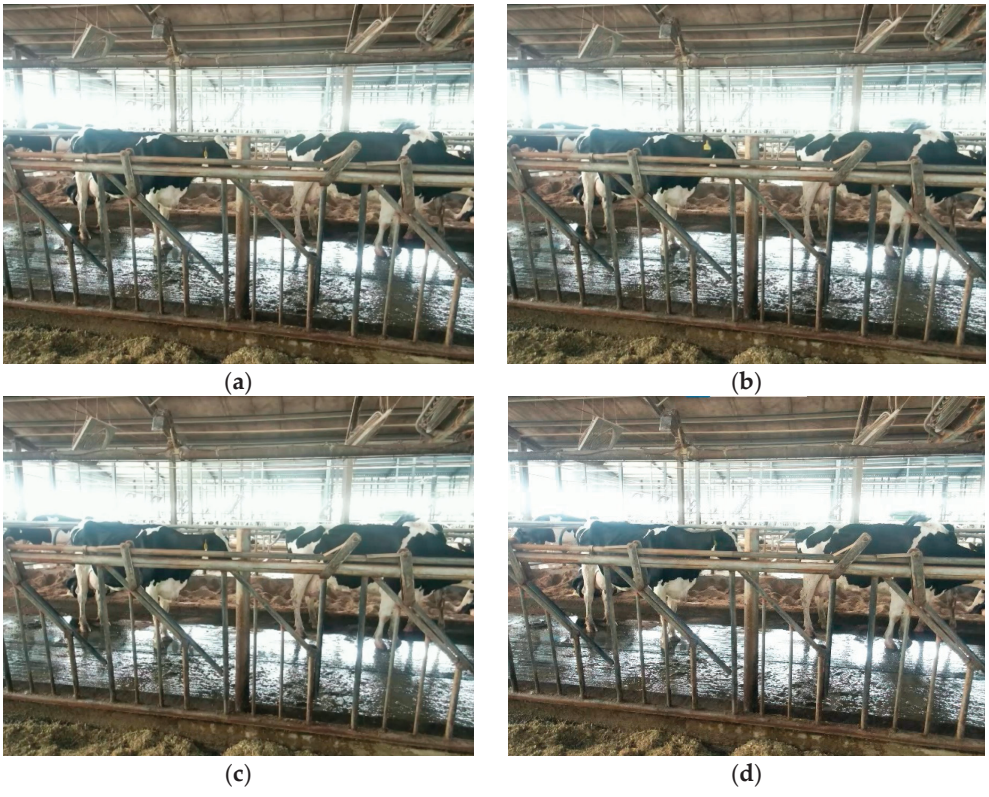


Figure 2. Adjacent keyframe images and origin images: (**a**) the previous keyframe; (**b**) the next keyframe; (**c**) the previous origin frame; (**d**) the next origin keyframe.

2.3. Image Augmentation

The randomness of cow movement and the instability of the inspection robot camera tracking led to insufficient initially acquired datasets, category imbalance, and problems in image quality. This study used image augmentation methods such as panning, mirroring, brightness adjustment, and contrast transformation to increase the diversity of the dataset, improve the image quality, meet the higher requirements of the deep learning algorithm model for the dataset, improve the accuracy of mask extraction, and lay the foundation for the subsequent neural network to classify the production groups.

Image augmentation is one of the data augmentation techniques used to address the problem of insufficient data required in deep neural network training in this study. Image augmentation can expand the dataset without collecting new samples [23]. Panning and mirroring are image augmentation methods based on geometric transformations. Panning is achieved by setting a threshold value to move the cow in a specific range along a random distance horizontally or vertically, in which the pixel size of the cow does not change, but only the filling of its background edges. Figure 3a shows where the edges after panning are filled with zero-pixel values. Mirroring refers to flipping the cow image left and right or up and down. This study mainly used left and right flipping to change the object's center position in the image to reduce the influence of the target object's position when taking pictures. Figure 3b shows that the cow image is flipped left and right. Luminance and contrast are image augmentation methods based on image color channel adjustment. The luminance adjustment can reduce the sensitivity of the model to color and reduce the influence of the light intensity of the cow farm on the shooting by setting a reasonable

threshold value. Figure 3c shows that the cow image is darkened after the luminance adjustment. Adjusting the image contrast can make a particular area in the image with a noticeable color difference more prominent. Combined with the cow's physical signs, the udder area will be more protuberant and facilitate feature extraction. Figure 3d shows that the cow udder outline is more transparent. The dataset was increased from 503 images to 1307 images by image augmentation, which enhances the diversity of samples.

Figure 3. Image augmentation: (**a**) image after panning; (**b**) image after mirroring; (**c**) image after adjusting brightness; (**d**) image after adjusting contrast.

2.4. Udder Segmentation Model

Instance segmentation combines object detection and semantic segmentation to achieve pixel-level individual segmentation and classification. Mask R-CNN and SOLOv2 are typical two-stage and one-stage models in instance segmentation, respectively. Mask R-CNN separates detection from segmentation and uses a top–down idea to predict the bounding box first and then segment individuals from each bounding box. SOLOv2 is an anchor-free instance segmentation model, which defines instance segmentation as a simultaneous detection task and segmentation task [24]. The two-stage detection model detects first and then segments, which has poor real-time performance, and the segmentation results correlate with excellent or low-quality bounding box localization. The one-stage model parallels detection with classification and has the characteristics of fast speed and high accuracy. However, such a model is strongly influenced by the detection accuracy. If the individuals have overlapping phenomena, the segmentation effect will be poor. Therefore, this study compared the effects of two segmentation models applied to cow udders, and selected a more suitable model. The parameter settings of the two segmentation models are shown in Table 1.

Table 1. Segmentation model parameter settings.

Instance Segmentation Algorithms	Parameter
SOLOv2	Max_iter = 60,000 Solver.Gamma = 0.1 Solver.Warmup_Factor = 1.0/100 Solver.Warmup_Iters = 10 Base_Lr: 0.0001 Batch size = 1 Epoches = 600 Steps per epoch = 100
Mask R-CNN	First 300 Epoches, Learning rate = 0.001, layers = 'heads' After 300 Epoches, Learning rate = 0.0001, layers = 'all' Batch size = 1

2.4.1. SOLOv2

The cow images were fed into the backbone network Res-101-FPN (Resnet-101-FeaturePyramidNetwork). Resnet ensures the correlation of gradients in the deep network during learning and avoids network degradation due to the increasing number of layers. FPN uses image pyramids to solve the multi-scale problem, fuses features from different convolutional layers during feature extraction to ensure the efficiency of detection of different-size cow udders, and obtains deeper semantic information, which in turn connects prediction of semantic categories and the instance mask of subsequent dynamic heads.

SOLOv2 continues the design of SOLOv1 but further improves the extraction efficiency and accuracy of the mask. Its network structure is shown in Figure 4. SOLOv2 is based on object detection and semantic segmentation. It transforms the segmentation problem into a location division problem by matching the target object's category to the instance's center. It divides the image into a grid of s $\times$ s. If the target object falls in the center of the grid, the grid performs semantic category prediction on the one hand and instance mask prediction on the other. When the overlap between the center region of the object and the grid is detected to be greater than a threshold, it is considered a positive sample, i.e., there is a category output. Accordingly, an instance mask corresponding to this output is generated. However, since there are often not many instances in the image so that the objects are sparsely distributed, there will be a channel (classifier) redundancy. SOLOv2 solves the output channel redundancy problem by decoupling the mask branch into the kernel branch and feature branch directly into convolutional kernel learning. For the post-processing step of repeated prediction, a matrix NMS (Non-Maximum Suppression) is proposed to accelerate the processing speed of the mask, and the generation of the target mask is more efficient and flexible compared with SOLOv1.

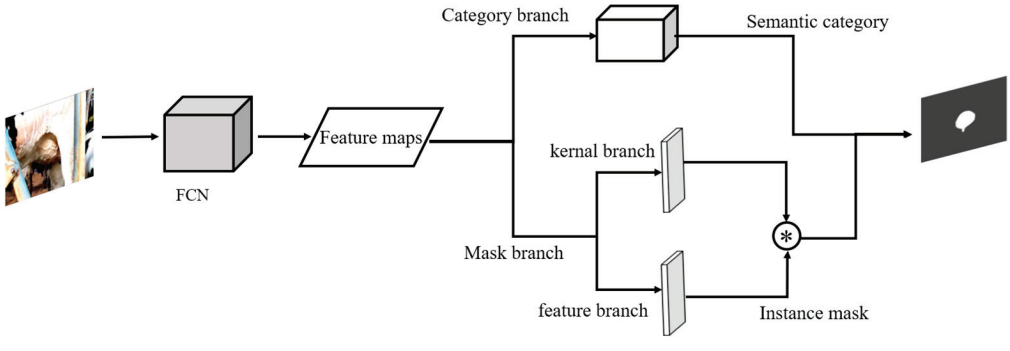


Figure 4. SOLOv2 network structure.

2.4.2. Mask R-CNN

Mask R-CNN also uses Resnet-FPN as the backbone network for feature extraction. Its network structure is shown in Figure 5. The model retains the RPN (Region Proposal Network) in Faster R-CNN for generating region proposals. The RPN input is the feature map generated in the feature extraction stage. To adapt to different target sizes, it generates nine anchor boxes of three scales and three aspect ratios for each point of the feature map. The obtained anchor boxes are processed in two ways: one is to perform foreground and background classification, i.e., to discriminate whether there is a target object in the anchor box and to score the likelihood; and the other is to perform regression to make the anchor frame closer to the ground truth box. Finally, the inaccurate anchor boxes are filtered to obtain the final RoI (Region of Interest). Then, the RoIAlign (Region of Interest Align) is used to adjust the feature map obtained by RPN to the same size. RoIAlign removes the quantization operation and instead uses bilinear interpolation for feature map reduction to avoid losing the information of the original feature map in the process. The feature map obtained by RoIAlign is input into the three-branch structure of Mask R-CNN to complete classification, bounding box regression, and segmentation mask prediction.

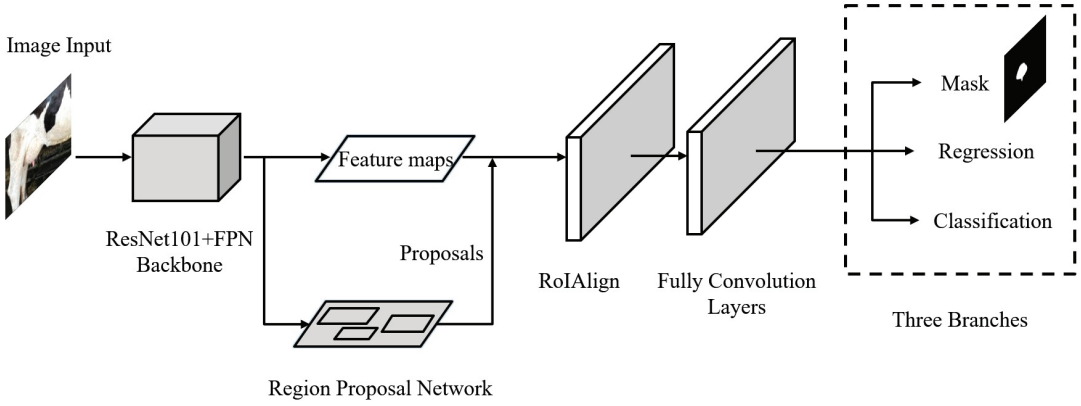


Figure 5. Mask R-CNN network structure.

The loss function equation of Mask R-CNN is shown in Equation (1).

$$L = L_{\text{cls}} + L_{\text{box}} + L_{\text{mask}} \tag{1}$$

where L_{cls} represents the classification loss, L_{box} represents the bounding-box loss, and L_{mask} represents the mask loss.

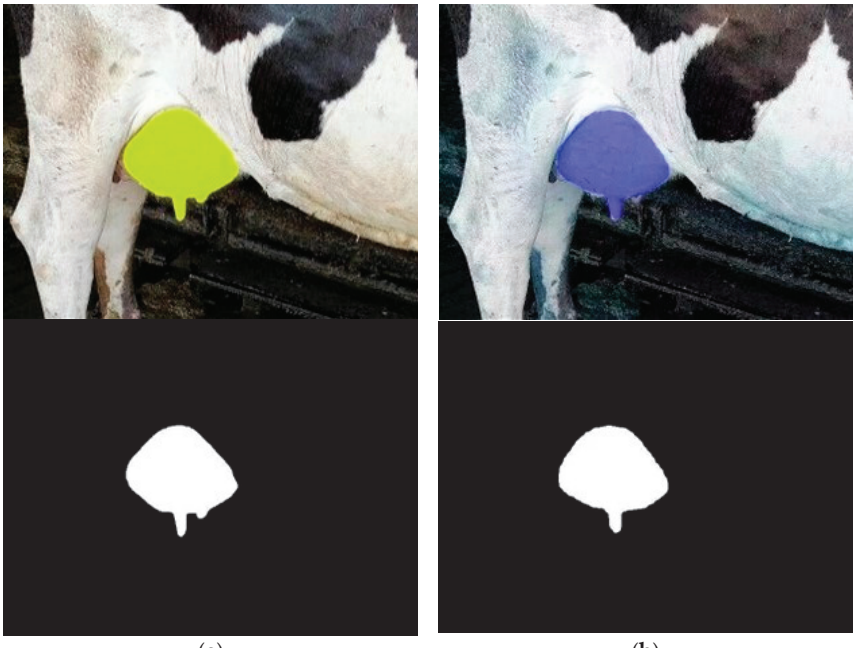
2.4.3. Comparison of Segmentation Effects

Figure 6 shows the comparison of SOLOv2 and Mask R-CNN segmentation results in the same environment, from which it can be seen that both algorithms segment well and the masks are close to the natural contours of the cow udder.

2.5. Udder Feature Extraction, Cleaning and Selection

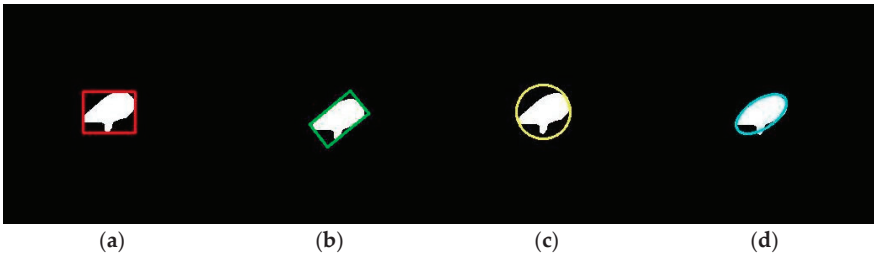
2.5.1. Udder Feature Extraction

In this study, 10 features were initially selected as neural network inputs: circumscribed regular rectangle width and height (max-width, max-height), minimum circumscribed rectangle width and height, aspect ratio (min-width, min-height, rect rate), circumcircle radius (radius), circumcircle area to contour area ratio (circle/contour), fitted elliptical length of major axis and minor axis, and major and minor axis ratio (elliptical a, elliptical b, elliptical rate). The feature values were extracted from the binary mask map extracted by the segmentation model, and its schematical map is shown in Figure 7.



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

Figure 6. Comparison of SOLOv2 and Mask R-CNN segmentation effects: (**a**) SOLOv2; (**b**) Mask R-CNN.



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

Figure 7. Contour feature extraction: (**a**) circumscribed regular rectangle contour; (**b**) minimum circumscribed rectangle contour; (**c**) circumcircle contour; (**d**) fitted elliptical contour.

2.5.2. Data Cleaning of Udder Features

In the acquisition of cow udder mask features, NaN (Not a Number) values and outliers with large deviations occurred due to the error of extracting the mask by instance segmentation and the influence of external environmental factors such as shooting angle and cows walking during the acquisition of cow images. In order to ensure the quality of the data, improve the accuracy of neural network prediction, and retain valuable data, this study conducted the mean replacement of missing values and outliers. It consisted of reading the CSV (Comma Separated Values) file through the Pandas, performing a lookup judgment, and applying the mean value of this feature data to replace the NaN, as shown in Table 2, where label 0 represents the actual low-yielding cows on the dairy farm and label 1 represents the high-yielding cows. In this study, based on the distribution of the data, the probability that the values are distributed in $(\mu - 2\sigma, \mu + 2\sigma)$ was 95.44% based on the 2σ principle. Considering the probability of falling outside $\pm 2\sigma$ was 4.56%, due to the influence of environmental errors and the sufficient data samples, the mean value replaces

the data with absolute values of errors vi > 2σ, and those with significant deviations from the mean are excluded.

Table 2. Comparison of example data before and after cleaning: (a) original data; (b) data after replacing null and outliers by mean values.

(**a**)

Max-Width	Max-Height	Rect Rate	Min-Width	Min-Height	Radius	Circle/Contour	Elliptical Rate	Elliptical a	Elliptical b	Production group
35	33	1.2381	0.3473	19.6373	20.3040	0.6562	0.4521	18.9112	41.8268	0
23	23	1.1304	23.2551	20.5718	12.4308	0.6417	0.8726	19.6495	22.5194	0
21	23	1.2381	23.2551	18.7830	11.9509	0.6474	0.7938	18.1305	22.8401	0
26	28	2.3684	31.8198	13.4350	16.1371	0.6562	0.7349	13.1578	33.0540	0
25	19	1.3333	24.0000	18.0000	12.5507	0.6830	0.7253	17.9755	24.7833	1
26	28	1.4000	29.6985	21.2132	15.1163	0.6491	0.6565	20.4981	31.2251	1
37	35	1.0167	33.8367	33.2820	18.9607	NaN	0.9455	33.3850	35.3099	1
36	38	1.0267	34.4354	33.5410	19.5209	0.7581	0.9622	34.0023	35.3373	1

(**b**)

Max-Width	Max-Height	Rect Rate	Min-Width	Min-Height	Radius	Circle/Contour	Elliptical Rate	Elliptical a	Elliptical b	Production group
35	33	2.0546	0.3473	19.6373	20.3040	0.3702	0.4521	18.9112	41.8268	0
23	23	1.1304	23.2551	20.5718	12.4308	0.6417	0.8726	19.6495	22.5194	0
21	23	1.2381	23.2551	18.7830	11.9509	0.6474	0.7938	18.1305	22.8401	0
26	28	1.2381	31.8198	13.4350	16.1371	0.6562	0.7349	13.1578	33.0540	0
25	32	1.3333	24.0000	18.0000	12.5507	0.6830	0.7253	17.9755	24.7833	1
26	28	1.4000	29.6985	21.2132	15.1163	0.6491	0.6565	20.4981	31.2251	1
37	35	1.0167	33.8367	33.2820	18.9607	0.7840	0.9455	33.3850	35.3099	1
36	38	1.0267	34.4354	33.5410	19.5209	0.7581	0.9622	34.0023	35.3373	1

2.5.3. Udder Feature Selection

Based on the data-cleaned cow udder trait dataset, correlation analysis was performed on the initially selected 10 traits. In this study, the Pearson correlation coefficient was used to analyze the correlation between the 10 features and the production group. The equation for calculating the Pearson correlation coefficient is shown in Equation (2).

$$\mathrm{r} = \frac{\sum_{i=1}^{n}\left(x_i-\overline{X}\right)\left(y_i-\overline{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(x_i-\overline{X}\right)^2\sum_{i=1}^{n}\left(y_i-\overline{Y}\right)^2}} \quad (2)$$

where r represents the Pearson correlation coefficient, x_i represents the i-th value in the sample of variable X, $\overline{X}$ represents the mean value of the sample of variable X, y_i represents the i-th value in the sample of variable Y, and $\overline{Y}$ represents the mean value in the sample of variable Y.

Figure 8a shows the correlation heat map of the data extracted based on the SOLOv2 mask, and Figure 8b shows the correlation heat map of the data extracted based on the Mask R-CNN mask. The color from dark to light indicates the correlation from low to high. The analysis shows that the Pearson correlation coefficients of the circumscribed regular rectangle width and height (max-width, max-height), the minimum circumscribed rectangle width and height (min-width, min-height), the circumcircle radius (radius), the fitted elliptical length of major axis and minor axis (elliptical a, elliptical b), and the production group are 0.49/0.51, 0.52/0.47, 0.52/0.60, 0.49/0.45, 0.57/0.58, 0.47/0.43, and 0.52/0.60, respectively, which are correlated between 0.4 and 0.6. The Pearson correlation coefficients of the minimum circumscribed rectangle aspect ratio (rect rate), the circumscribed circle area to contour area ratio (circle/contour), and the fitted elliptical major-to-minor axis ratio (elliptical rate) with the production group are 0.09/0.21, 0.00/−0.16, and −0.07/−0.27, respectively, with absolute values in the range 0.0–0.4. The absolute values of the correlation coefficients were found to be 0–0.3 (0 is not included) for weak correlation, 0.3–0.5 (0.3 is not included) for low correlation, 0.5–0.8 (0.5 is not included) for moderate correlation, and 0.8–1.0 (0.8 is not included) for high correlation. Therefore, to reduce the complexity of the neural network algorithm and improve the classification accuracy and efficiency, the weakly correlated features were excluded.

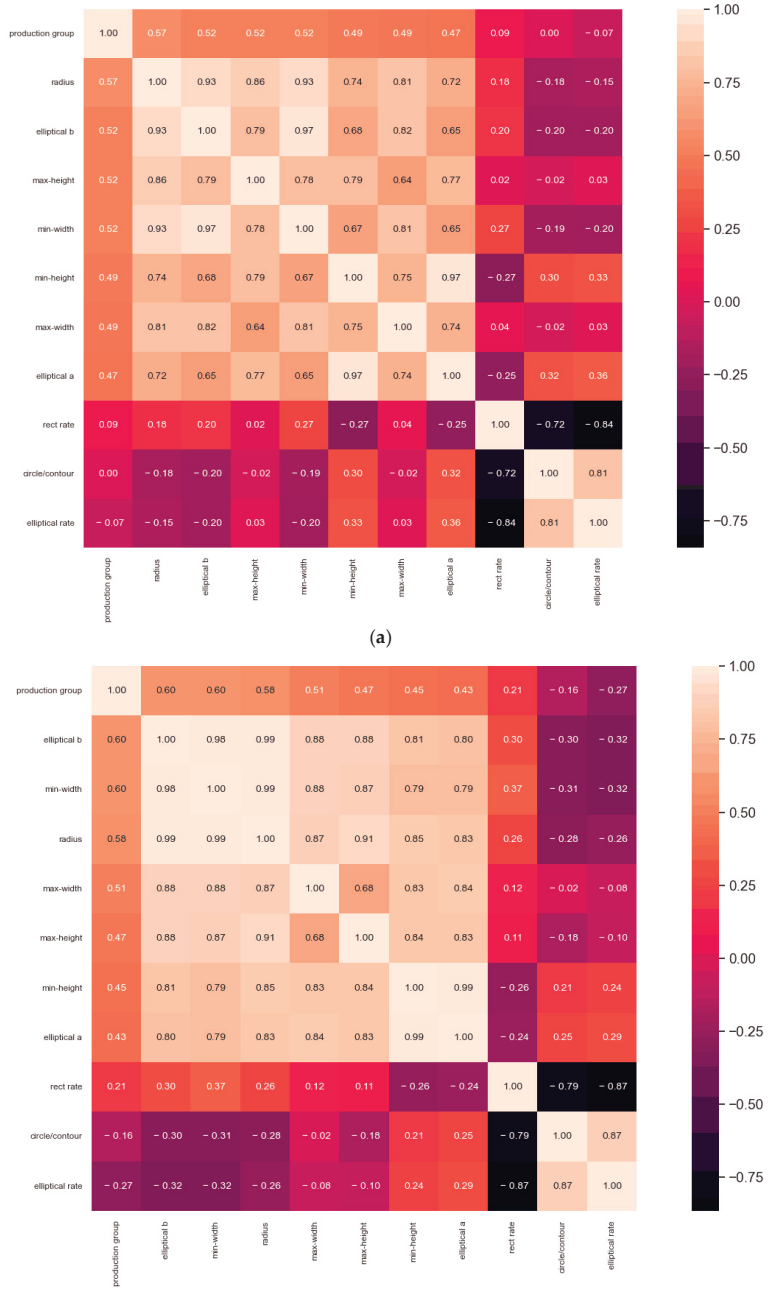


Figure 8. Heat map of correlation coefficients between udder features: (**a**) correlation coefficients of SOLOv2 mask map extracted data; (**b**) correlation coefficients of Mask R-CNN mask map extracted data.

2.5.4. Data Distribution

The data were analyzed to obtain the data distribution of samples with different characteristics of high- and low-yielding cows processed by the two algorithms. The

characteristic kernel density of high- and low-yielding cows was plotted by selecting the characteristic variables for the four different calculation methods, with the horizontal coordinates indicating the range of values taken and the vertical coordinates indicating the probability density of the occurrence of data points. Figure 9 shows that high-yielding cows have greater values than low-yielding cows in the max-width, the min-width, the radius, and the elliptical b, where the distribution is dense. A shaded variogram was used to visualize the relationship between the two characteristic variables, and the shading indicates the density of the data points, which can be used to visualize the distribution between the characteristic variables and the difference in the distribution of the characteristics of high- and low-yielding cows.

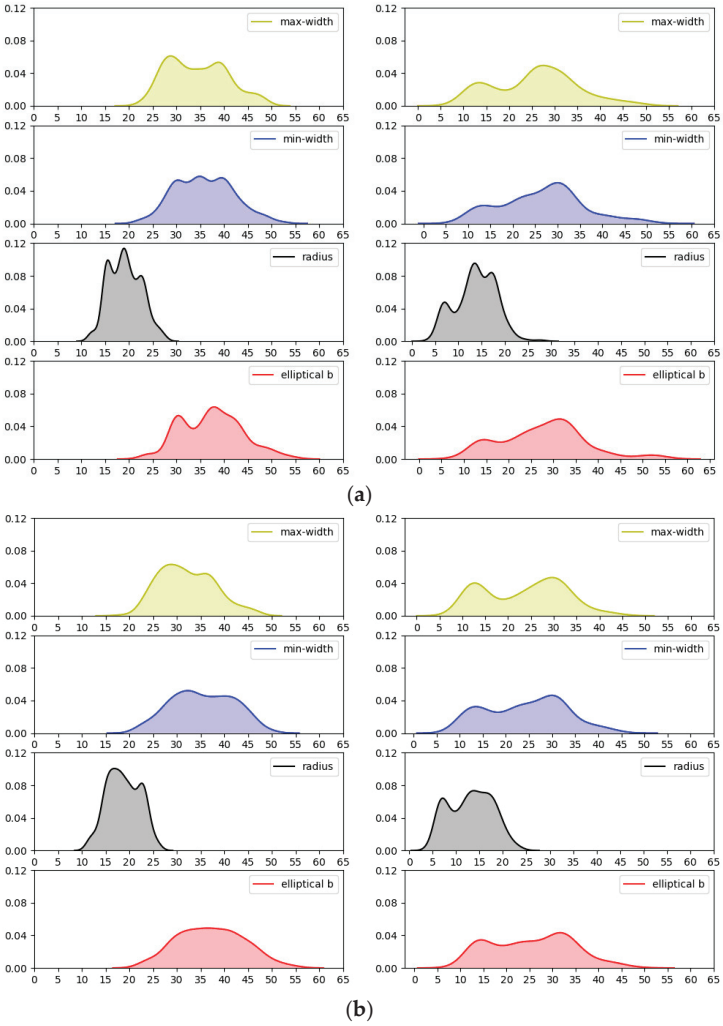


Figure 9. The high- and low-yielding cows are distributed in different feature variables, with the high yield on the left and low yield on the right: (**a**) SOLOv2 mask feature density; (**b**) Mask R-CNN mask feature density.

2.6. Production Groups Classification Model

This study focuses on improving the neural network model for the production groups cows. The neural network is the core of deep learning, which is connected by several

neurons. Its elemental composition is the input, hidden, and output layers. The neurons in the hidden layer refine the input features to enhance the model training effect. The neurons in the network adjust the weights and biases corresponding to different features by continuous learning, constantly normalize the input of the lower layers by using the activation function for nonlinear transformation, and connect different layers. The model parameters are updated by backpropagating the loss function to close the predicted value to the actual value and improve the classifier simultaneously. Finally, the neural network classifies the udder dataset based on the weight vector.

(1) CNN-LSTM

Convolutional neural networks have superior performance, and their application areas include image and data classification, object detection, video processing, natural language processing, speech recognition, etc. [25,26]. LSTM is a long short-term memory network, capable of handling sequential and textual problems [27], a variant of RNN (Recurrent Neural Network). It combines short-term memory with long-term memory through exquisite gate control. It solves the problem of gradient disappearance [28]. LSTM can learn long-term dependent information and generally targets back-and-forth logic, sequence problems with temporal concepts, and text problems. This study explored the effect of binary classification of the high- and low-yielding cow dataset with a certain temporal nature by improving the CNN-LSTM deep learning model. The convolution extracts deep features of the cow mask, and then adding LSTM further processes the output features of the convolution layer. The input layer is set as a sequence input layer with size $7 \times 1 \times 1$ (the dataset has seven features). The folding sequence layer converts the sequence data into the vector, then puts it into the convolutional network with two convolutional layers, which respectively have 16 and 32 convolutional kernels, both with sizes 2×1. Furthermore, a batch normalization layer is added before the activation function to speed up the model convergence and alleviate the gradient dispersion. The max pooling layer is chosen (i.e., downsampling, to compress the multiple features after convolution and filter out the unimportant features), and then the deep features are obtained after convolution, which are sequence unfolded and input into the LSTM layer. Some inconsequential features are discarded using the dropout layer to prevent the occurrence of the overfitting phenomenon. Finally, the output size of the fully connected layer is 2 for two classifications. The softmax activation function is employed to connect the classification layer; the network model is shown in Figure 10.

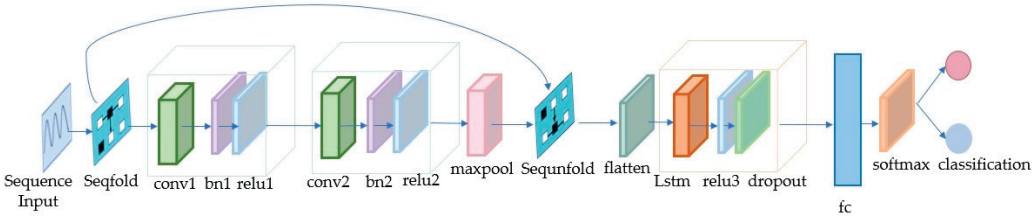


Figure 10. CNN-LSTM improvement model.

(2) BPNN

A BP neural network is a multilayer feedforward network using a backpropagation algorithm, and its basic idea is gradient descent. The BP neural network includes two processes: forward propagation of signals and backward propagation of errors. The sample data are input into the neural network through the input layer, and the hidden layer calculates the prediction result to complete the forward propagation. Then, according to the error between the prediction result and the actual result, the chain rule is used to calculate the error of each layer and to calculate the gradient according to the error to update the weights and biases of each layer to complete the backward propagation. The

neural network has a strong nonlinear mapping ability and can establish relationships between various udder characteristics and the production area. Therefore, based on the idea of the BP neural network, this study improved the primary BP neural network to make it suitable for classifying production groups. The neural network structure diagram is shown in Figure 11. There are seven feature values for the input data, and thus seven nodes were selected for the input layer. In order to ensure the low complexity of the network parameters and better map the relationship between the features and the production area, two hidden layers with six nodes were constructed. Due to the small number of classification samples, Bayesian regularization was selected as the training function to improve the model's generalization ability. The activation function of the hidden layer uses tansig; the equation is as in Equation (3) and the activation function of the output layer uses softmax to achieve classification; the equation is as in Equation (4). The number of nodes in the output layer was two with the same classification category.

$$tansig = \frac{2}{1+e^{-2x}} - 1 \tag{3}$$

where x represents the output value of the node.

$$softmax = \frac{e^{z_i}}{\Sigma_{c=1}^{C} e^{z_c}} \tag{4}$$

where z_i represents the i-th node's output value, and C represents the number of output nodes.

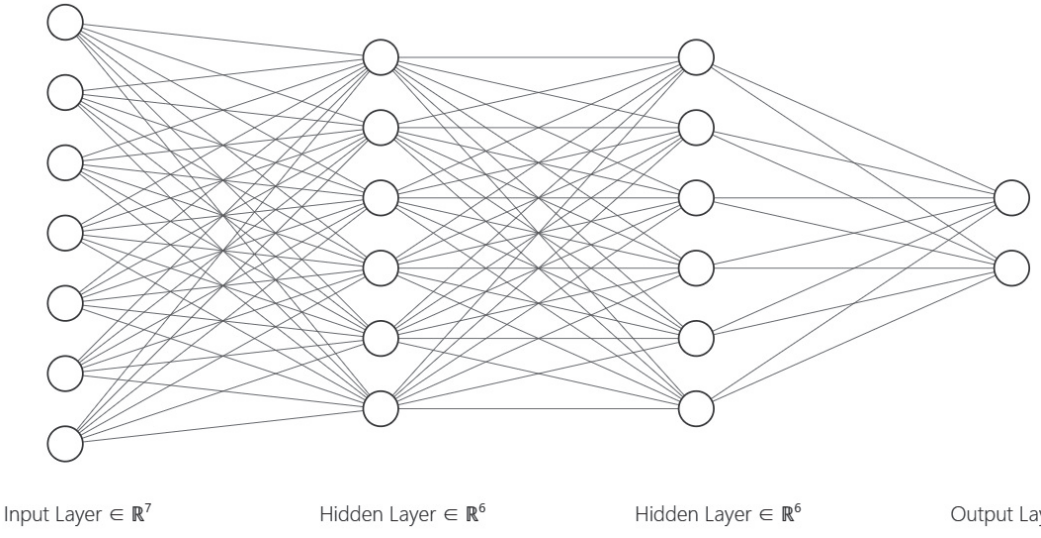


Figure 11. BP neural network structure.

2.7. Classification Assessment Indicators

In the classification task, classification results were classified into four categories: true positives (*TP*), false positives (*FP*), false negatives (*FN*), and true negatives (*TN*). In this study, accuracy, precision, recall, and F1-score metrics were chosen to assess the model classification performance.

The accuracy indicates the accuracy of the model prediction, i.e., the proportion of correctly predicted samples to the overall samples, and is calculated as in Equation (5).

$$Accuracy = \frac{n_{correct}}{n_{total}} = \frac{TP+TN}{TP+FP+FN+TN} \tag{5}$$

The precision reflects the ability of the model to discriminate negative samples, which is the proportion of samples predicted to be positives out of samples that are true positives, is calculated as in Equation (6).

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

The recall reflects the ability of the model to identify positive samples, which is the proportion of true positives predicted to be positives and is calculated as in Equation (7).

$$Recall = \frac{TP}{TP+FN} \tag{7}$$

The F1-score is the summed average of precision and recall, calculated as in Equation (8).

$$F1 = \frac{2 \times Presion \times Recall}{Presion + Recall} \tag{8}$$

2.8. Experimental Design and Setup

2.8.1. Experimental Environment

The software platforms used in this study are Labelme 5.1.1 (MIT, Cambridge, MA, USA) for image annotation, PyCharm 2022.1 Community Edition (JetBrains, Prague, Czech Republic) and Python 3.7 (Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands) for image augmentation and feature extraction, IBM SPSS Statistics 26 (IBM Corp, Armonk, NY, USA) for correlation analysis, and Matlab 2021b for neural network construction.

The deep learning network was GPU parallel-accelerated by CUDA 11.6, and cuDNN 8.8.1 was used as the acceleration library for deep convolutional neural networks. SOLOv2 was built based on Detectron2 and AdelaiDet, which are deep learning frameworks. Mask R-CNN was built based on the TensorFlow and Keras frameworks.

The hardware platform for this study was 11th Gen Intel® Core(TM) i5-11400H @ 2.70 GHz, 16 G RAM, and NVIDIA GeForce RTX 3050 Laptop GPU.

2.8.2. Instance Segmentation Dataset

After crucial frame extraction and image augmentation, 1093 cow udder images were gained, and of these 449 images were of high-yielding cows and 644 images of low-yielding cows. The training set and test set were divided according to the ratio of 7:3 to obtain 766 images in the training set and 327 images in the test set.

2.8.3. Classification Dataset

Two datasets, both of size 1307, were constructed by extracting the mask features of SOLOv2 and Mask R-CNN segmentation separately and randomly dividing the training and test sets according to the ratio of 7:3.

2.9. Cow Farm Management

2.9.1. Animal Welfare

The average weight of cows in this cattle farm was 660 kg, and the average age was 5. Their living conditions were good. In terms of diet, based on the physiological differences between high- and low-yielding cows, high-yielding cows have a high feed intake and high cow metabolism compared to low-yielding cows, and therefore need to be supplied with more feed and drinking water for the maintenance of physiological needs and metabolism, with high-yielding cows having up to 90 ± 10 kg of daily feed intake. According to the weather one must provide a reasonable amount of feeding water, when the weather is hot in summer, the water should be increased by five to six times; in terms of living environment, to ensure the cleanliness and comfort of the cows' living environment, milking aisles are cleaned up two times a day, lying feces are cleaned up three times a day, lying beds are tidied up at least one time a day, and the depth of plowing is more than 15 centimeters. This fully guarantees animal welfare.

2.9.2. Practice and Production

Based on the results of our classification result, high- and low-yielding cows can be categorized for zonal management. In actual management, feeding management is mainly focused on high-yielding cows to improve milk production. Compared with low-yielding cows, high-yielding cows have many unique physiological characteristics. Firstly, high-yielding cows have high nutrient requirements and high daily feed intake. Secondly, their basal metabolic rate is higher, and their respiratory and heart rate are higher than those of low-yielding cows. Therefore, when feeding, attention was given to the structure of the diet with a moderate forage to concentrate ratio, adopting a scientific feeding method, and controlling the amount and frequency of feeding. At the same time, cows were provided with a suitable barn environment and were cleaned regularly. Our classification of high- and low-yielding cows provides support for zoning and fine management of dairy farms and provides a boost to improve cow production.

3. Results and Discussions

3.1. Segmentation Model Evaluation

3.1.1. Loss Function

The loss functions of optimal models of SOLOv2 and Mask R-CNN are shown below, and both types of algorithms use weights that have been trained on the MSCOCO (Microsoft Common Objects in Context) dataset as pre-training weights. Utilizing the weights attained from training on the large-scale dataset to initialize the network model allows transferring the learned generic features to the new task, thus improving the performance and generalization of the model. As shown in Figure 12, both algorithms converge after a small number of iterations, taking low loss values.

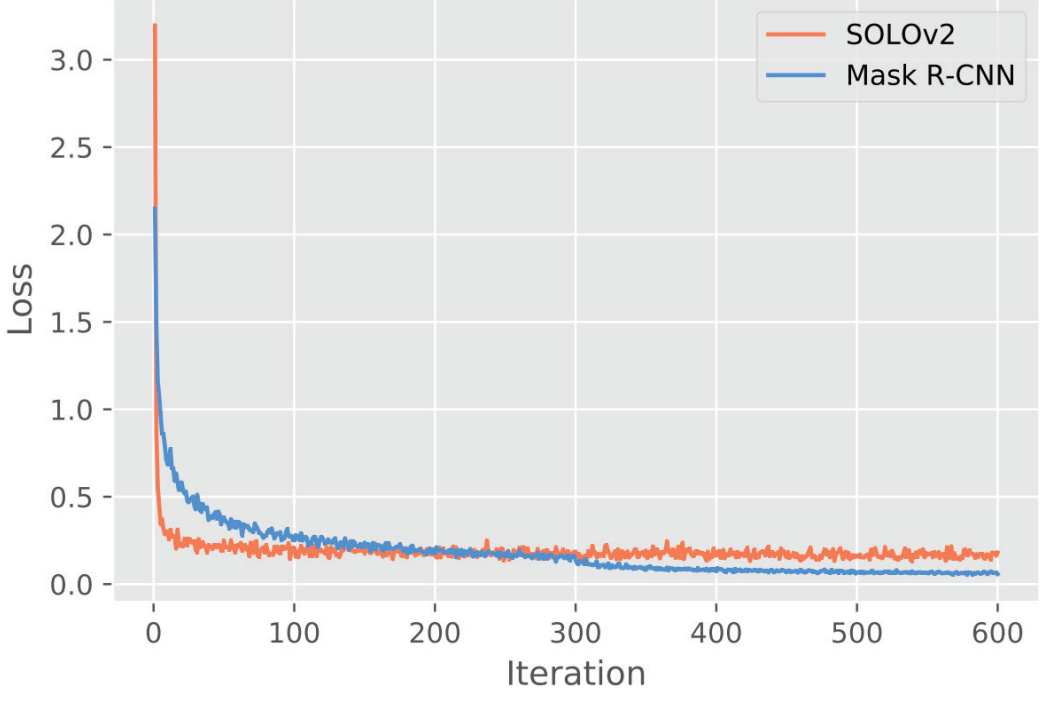


Figure 12. Segmentation model loss function.

3.1.2. Segmentation Accuracy

This study first exploits the idea of segmentation and then classification for dividing production groups. In the instance segmentation stage, the two-stage and one-stage segmentation models Mask R-CNN and SOLOv2, respectively, were compared. The mAP (mean Average Precision), AP50 (Average Precision), and AP75 were used as metrics to measure the performance of the two algorithms. As can be seen from Table 3, SOLOv2 outperformed Mask R-CNN in all three metrics, and mAP was 7.12% higher than Mask R-CNN. Extended analysis of the AP50 index in Table 3 shows that SOLOv2 and Mask R-CNN were 98.87% and 95.03%, respectively, implying that the vast majority of extracted masks of both algorithms was above 50% of the actual cow udder IoU (Intersection over Union) ratio, which can achieve complete cow udder segmentation more accurately. Since SOLOv2 outperformed Mask R-CNN for object edge segmentation, SOLOv2 performed better for targets with distinct edge features such as cow udders.

Table 3. Segmentation model accuracy.

Instance Segmentation Algorithms	mAP	AP50	AP75
SOLOv2	74.09%	98.87%	92.49%
Mask R-CNN	66.97%	95.03%	70.48%

Since the metrics selected cannot fully evaluate the performance of the segmentation model, this study extracted features from the mask maps segmented by both algorithms. The features were input into the classification algorithm to further analyze the performance of the segmentation model through the classification effect.

3.2. Classification Model Evaluation

3.2.1. Effect of Neural Network Model on Test Results

Based on the cow udder mask features datasets, two neural network models were improved in this study. The first one was because the udder mask feature dispersion had certain temporal nature characteristics. A variant LSTM of the recurrent neural network was introduced and convolutional layer and max pooling layer were added to optimize the network and boost the model performance. The second model was employed that improves the basic BP neural network, builds two hidden layers, and uses a backpropagation algorithm to reduce the prediction error. As can be seen from Table 4, the accuracy of the testing sets of the two neural network models is relatively ideal, and the accuracy of CNN-LSTM is superior to BPNN no matter whether for the dataset segmented by SOLOv2 or in the dataset segmented by Mask R-CNN. This is because the CNN-LSTM neural network, compared with the BP neural network, has added convolution layers and increased the number of neurons, making the network structure more complex. Additionally, the performance of CNN-LSTM and BPNN on the dataset segmented by SOLOv2 is superior to that of Mask R-CNN, with the highest accuracy of 96.44% (SOLOv2 + CNN − LSTM), which further indicates that the segmentation effect of SOLOv2 is better than that of Mask R-CNN. The loss function curves corresponding to the two segmentation models based on the CNN-LSTM neural network are shown in Figure 13, and the cross-entropy loss functions corresponding to the two segmentation models based on the BPNN neural network are shown in Figure 14.

Table 4. Improvement of neural network evaluation metrics.

Classification Algorithms	Instance Segmentation Algorithms	Accuracy	Precision	Recall	F1 Score
CNN-LSTM	SOLOv2	**96.44%**	**98.00%**	**96.47%**	**97.23%**
	Mask R-CNN	90.49%	92.40%	91.88%	92.14%
BPNN	SOLOv2	**93.13%**	**88.65%**	**91.91%**	**90.25%**
	Mask R-CNN	90.19%	87.70%	90.68%	89.17%

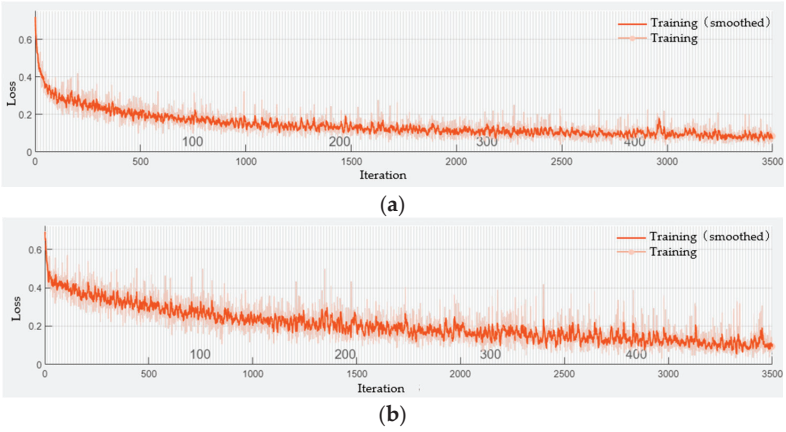


Figure 13. Improved neural network loss function: (**a**) SOLOv2; (**b**) Mask R-CNN.

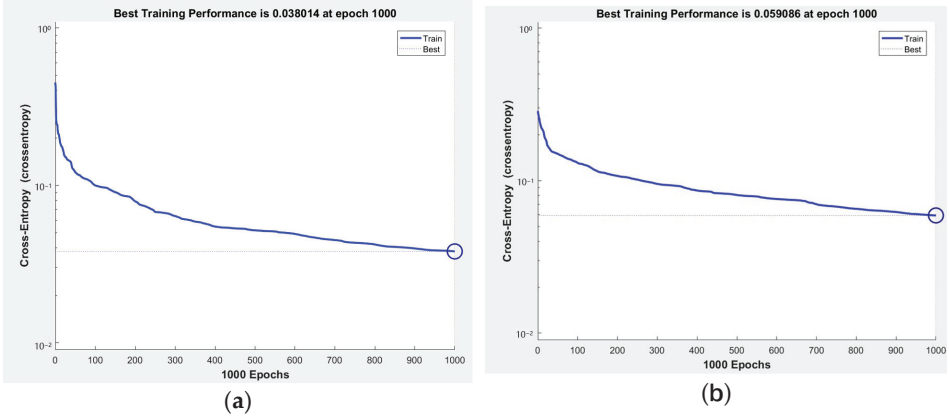


Figure 14. Cross-entropy loss functions: (**a**) SOLOv2; (**b**) Mask R-CNN.

3.2.2. Comparison of Test Results

In this study, four commonly used machine learning algorithms, namely naive Bayes, K-nearest neighbor, support vector machine, and random forest, were used to classify production groups and compare the effect with neural network classification. Performance metrics are shown in Table 5. The confusion matrix is shown in Figure 15. After analysis and comparison, K-nearest neighbor and random forest performed better among the four algorithms, with the accuracy of SOLOv2 reaching 92.62%/92.74% and Mask R-CNN reaching 85.93%/89.77%. However, both are lower than the two types of neural networks, reflecting the unique advantages of neural networks in multi-feature classification problems.

Table 5. Evaluation metrics of the machine learning algorithm.

Classification Algorithms	Instance Segmentation Algorithms	Accuracy	Precision	Recall	F1 Score
Naive Bayes	SOLOv2	75.72%	79.28%	84.66%	81.88%
	Mask R-CNN	76.35%	75.69%	84.62%	79.90%
K-Nearest Neighbor	SOLOv2	**92.62%**	**94.82%**	**93.70%**	**94.62%**
	Mask R-CNN	85.93%	86.79%	89.61%	88.18%
Support Vector Machines	SOLOv2	68.45%	67.02%	100%	80.25%
	Mask R-CNN	66.16%	61.97%	100%	76.52%
Random Forest	SOLOv2	**92.74%**	**91.37%**	**97.55%**	**94.36%**
	Mask R-CNN	89.77%	89.22%	92.86%	91.00%

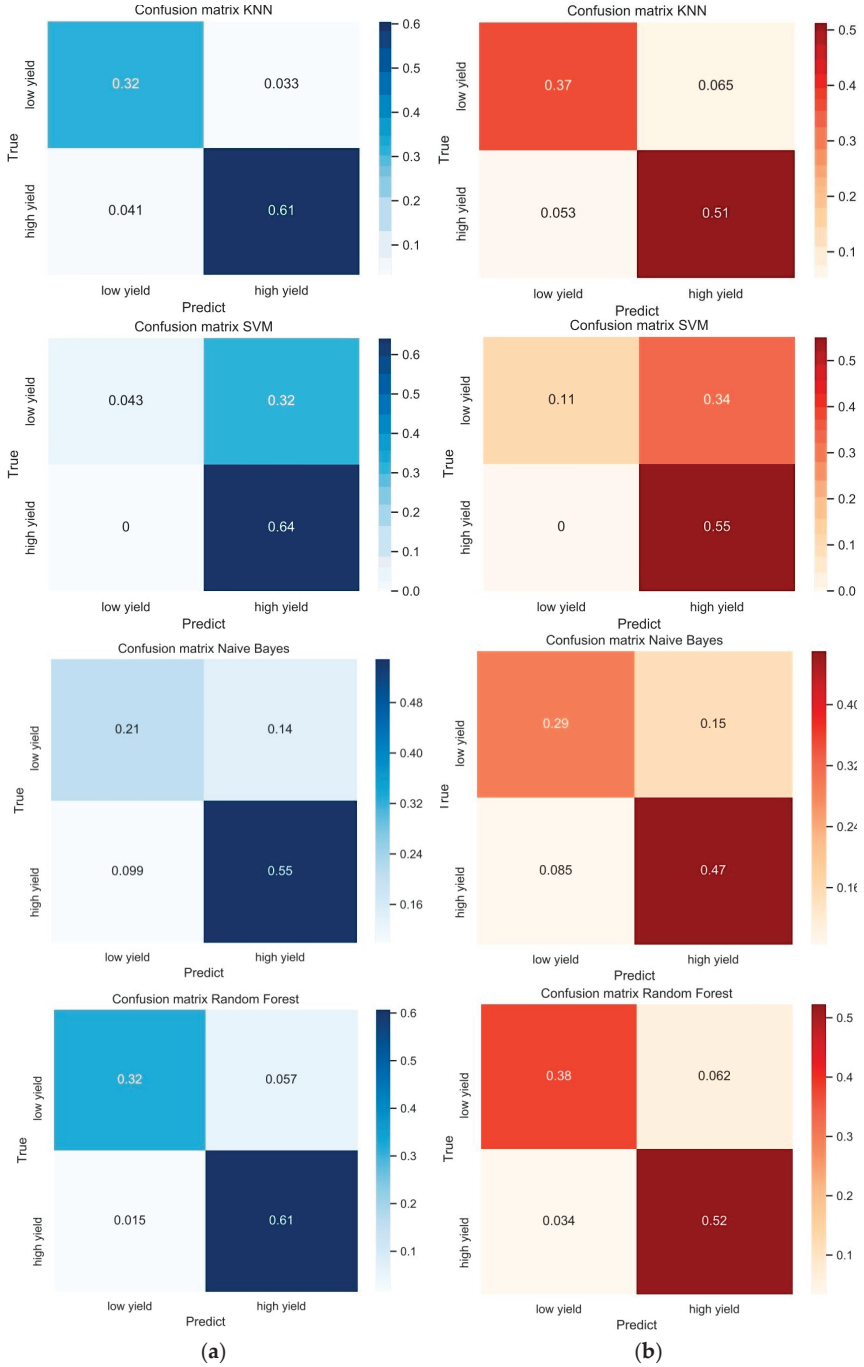


Figure 15. Confusion matrix obtained from 4 classification models: (**a**) input is the feature data extracted by SOLOv2 mask; (**b**) input is the feature data extracted by Mask R-CNN mask.

In this study, we introduced a method to divide high- and low-yielding cows already in their own pens according to their production levels based on the SOLOv2 and CNN-LSTM models. The main objectives were to investigate the potential of instance segmentation to extract the cow udders and establish a classification model for high- and low-production groups based on neural network. The segmentation effect of SOLOv2 and Mask R-CNN was evaluated; features that can well characterize cow udder traits were explored; and the effectiveness of the improved CNN-LSTM classifier for high- and low-yielding area division was verified.

The technology in the study allows for adjustments to be made to cows after they have been grouped. For example, if some cows in the high-yielding group have entered the low-yielding threshold, for large farms with many cows, it is labor-intensive to rely on manual labor to identify which cows need to be adjusted to the low-yielding group on a regular basis, whereas the technology in the study can be used to realize automatic and convenient identification and adjustment. Meanwhile, cows in the low-yielding group whose milk production capacity has been improved through effective management can also be adjusted to the high-yielding group by identification. This will help farmers to make a decision.

The technology used in this study has certain value and significance compared with grouping high- and low-yielding cows directly according to their actual milk production. Firstly, if the cows are divided by 305 d milk production, the statistical time is long, and it cannot divide the cows quickly and conveniently. In this study, the technology can directly realize the grouping of cows by obtaining cow images and recognizing cow udders under the condition of unknown milk production. Secondly, when the actual milk production is recorded manually, the workload is larger. However, the technology in the study does not need a large amount of data when classifying new cows, which reduces the labor cost and can directly obtain the grouping results. Thirdly, for some cows in the high-yielding group that enter the threshold of the low-yielding group, the techniques in this study allow for quick batch screening and then adjusting cows from the high-yielding group to the low-yielding group.

We compared our technique with several similar studies, and found that there were a few limitations of our technique's employment. A previous study [29] used multiple cameras simultaneously to obtain the depth maps of the cow's body in different directions, artificially labeled the different body parts of the cow, and classified body parts by pixels. The method can alleviate cow fences occlusions to a certain extent, which may seriously influence cow udders segmentation and classification results. The problem of cow fence occlusions also appeared in our research and should be well-handled in the follow-up work.

The environment of a cow barn is complex and weather causes vast variations in illumination, which greatly challenged the subsequent image processing procedure. Thus, the results and reliability of image-processing-based methods may decrease significantly when the conditions covered by training samples are insufficient. Bobbo et al. [30] compared multiple machine learning methods to predict udder health status based on somatic cell counts in dairy cows. Another study [31] utilized ultrasound echotexture analysis of the mammary gland and a deep learning algorithm to predict milk yield. Methodology in [32] proposed a Rfine mask two-stage instance segmentation, a combination of the convolutional neural network ConvNeXt and ECA modules. Inspired by these studies, division of high- and low-production groups by fusing multimodal data should be considered, such as physical and chemical data, visible light data, and ultrasound images. Moreover, attention modules can be integrated into CNN-LSTM to deal with small-target and multi-scale-target problems.

4. Conclusions

Based on the relationship between udder properties and milk production, this study proposed a method to divide production groups by segmentation first and then classification. In the segmentation stage, a self-designed inspection robot acquired the video of the cow's udder. Then, for the problem of many duplicated images but low diversity,

keyframe extraction and image augmentation were used to expand the dataset. After image preprocessing, to compare the performance of one-stage and two-stage segmentation models in this task, SOLOv2 and Mask R-CNN were selected to segment the images and extract the binary mask images. In the classification stage, 10 feature values were extracted from the mask images. Afterward, the data were cleaned, and features were selected to make the classification model training more efficient and accurate. The results show that the segmentation effect of SOLOv2 was better than Mask R-CNN with mAP up to 74.09%, and the classification effect of CNN-LSTM was better than BPNN. The segmentation using SOLOv2 and classification using CNN-LSTM obtained a production groups' classification accuracy of up to 96.44%, indicating that the proposed method based on the segmentation model and the neural network has effective results in cow production groups.

Author Contributions: Conceptualization, Y.L., M.X. and X.Z.; data curation, G.C., L.Q., Z.C., Z.L. and C.X.; formal analysis, G.C., L.Q., Z.C., Z.L. and C.X.; funding acquisition, M.X. and X.Z.; methodology, G.C., L.Q., Y.L. and X.Z.; project administration, Y.L., M.X. and X.Z.; supervision, Y.L., M.X. and X.Z.; visualization, G.C. and L.Q.; writing—original draft, G.C. and L.Q.; writing—review and editing, Y.L., M.X. and X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Program for International S&T Cooperation Projects of Jiangsu, China (BZ2021022), and the Student Innovative Training Program of Nanjing Agricultural University (202219XX476).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We are thankful to Yungang Bai, Sunyuan Wang and Hengtai Li, who have contributed to our field data collection and primary data analysis.

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

References

1. Feil, A.A.; Schreiber, D.; Haetinger, C.; Haberkamp, A.M.; Kist, J.I.; Rempel, C.; Maehler, A.E.; Gomes, M.C.; Silva, G.R. Sustainability in the dairy industry: A systematic literature review. *Environ. Sci. Pollut. Res. Int.* **2020**, *27*, 33527–33542. [CrossRef] [PubMed]
2. Teng, G.H. Information sensing and environment control of precision facility livestock and poultry farming. *Smart Agric.* **2019**, *1*, 1–12. [CrossRef]
3. Pawlina, E.; Wojciech, K.; Marian, K. The changes in udder size of Red & White breed cows in first and third lactation. *Med. Weter.* **2000**, *56*, 672–674.
4. Okkema, C.; Grandin, T. Graduate Student Literature Review: Udder edema in dairy cattle—A possible emerging animal welfare issue. *J. Dairy Sci.* **2021**, *104*, 7334–7341. [CrossRef]
5. Juozaitienė, V.; Saulius, T.; Evaldas, S. The correlation between cows udders morphology and milking characteristics. *Vet. Ir Zootech. (Vet. Med. Zoot)* **2007**, *38*, 17–21.
6. Mišeikienė, R.; Tušas, S.; Matusevičius, P.; Kerzienė, S. Quarter milking parameters by lactation in dairy cows. *Mljekarstvo Časopis Za Unaprjeđenje Proizv. I Prerade Mlijeka* **2019**, *69*, 108–115. [CrossRef]
7. Lin, C.Y.; Lee, A.J.; McAllister, A.J.; Batra, T.R.; Roy, G.L.; Vesely, J.A.; Wauthy, J.M.; Winter, K.A. Intercorrelations Among Milk Production Traits and Body and Udder Measurements in Holstein Heifers. *J. Dairy Sci.* **1987**, *70*, 2385–2393. [CrossRef]
8. Magaña-Sevilla, H.; Sandoval-Castro, C.A. Technical Note: Calibration of a Simple Udder Volume Measurement Technique. *J. Dairy Sci.* **2003**, *86*, 1985–1986. [CrossRef]
9. Franchi, G.A.; Jensen, M.B.; Foldager, L.; Larsen, M.; Herskin, M.S. Effects of dietary and milking frequency changes and administration of cabergoline on clinical udder characteristics in dairy cows during dry-off. *Res. Vet. Sci.* **2022**, *143*, 88–98. [CrossRef]
10. Bertulat, S.; Fischer-Tenhagen, C.; Werner, A.; Heuwieser, W. Technical note: Validating a dynamometer for noninvasive measuring of udder firmness in dairy cows. *J. Dairy Sci.* **2012**, *95*, 6550–6556. [CrossRef]
11. Chen, S.S.; Wang, M.H. Linearized Appraisal of Dairy Cow's Conformation Using Image Measurement Technique. *J. China Agric. Univ.* **1996**, *1*, 93–98.
12. Guo, H.; Wang, P.; Ma, Q.; Zhu, D.H.; Zhang, S.L.; Gao, Y.B. Acquisition of Appraisal Traits for Dairy Cow Based on Depth Image. *Trans. Chin. Soc. Agric. Mach.* **2013**, *44*, 273–276+229. [CrossRef]
13. Huang, J.R.; Qian, D.P.; Wang, W.D.; Chen, X.H. Developing Linear Appraisal of Dairy Cow Conformation System with Image Processing Technique. *Trans. Chin. Soc. Agric. Mach.* **2007**, *38*, 111–113+171. [CrossRef]

14. Hu, X.T.; Zhang, Y.C. Cow Breast Shape Features Analysis Method Based on Three-Dimensional Point Cloud. *J. Tianjin Univ. Sci. Technol.* **2012**, *27*, 61–64. [CrossRef]
15. Xie, Q.J.; Zhou, H.; Bao, J.; Li, Q.D. Review on Machine Vision-based Weight Assessment for livestock and Poultry. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53*, 1–15. [CrossRef]
16. Gao, Y.; Guo, J.L.; Li, X.; Lei, M.G.; Lu, J.; Tong, Y. Instance-level Segmentation Method for Group Pig Images Based on Deep Learning. *Trans. Chin. Soc. Agric. Mach.* **2019**, *50*, 179–187. [CrossRef]
17. Rossi, L.; Valenti, M.; Legler, S.E.; Prati, A. LDD: A Grape Diseases Dataset Detection and Instance Segmentation. *Image Anal. Process. –ICIAP* **2022**, *13232*, 383–393. [CrossRef]
18. Sun, X.M.; Fang, W.T.; Gao, C.Q.; Fu, L.S.; Majeed, Y.; Liu, X.J.; Gao, F.F.; Yang, R.Z.; Li, R. Remote estimation of grafted apple tree trunk diameter in modern orchard with RGB and point cloud based on SOLOv2. *Comput. Electron. Agric.* **2022**, *199*, 107209. [CrossRef]
19. Qiao, Y.L.; Truman, M.; Sukkarieh, S. Cattle segmentation and contour extraction based on Mask R-CNN for precision livestock farming. *Comput. Electron. Agric.* **2019**, *165*, 104958. [CrossRef]
20. Ma, L.; Xie, F.; Liu, D.; Wang, X.; Zhang, Z. An Application of Artificial Neural Network for Predicting Threshing Performance in a Flexible Threshing Device. *Agriculture* **2023**, *13*, 788. [CrossRef]
21. Kumar, G.; Bhatia, P.K. A detailed review of feature extraction in image processing systems. In Proceedings of the 2014 Fourth International Conference on Advanced Computing & Communication Technologies, Rohtak, India, 8–9 February 2014; pp. 5–12. [CrossRef]
22. Duan, E.; Hao, H.; Zhao, S.; Wang, H.; Bai, Z. Estimating Body Weight in Captive Rabbits Based on Improved Mask RCNN. *Agriculture* **2023**, *13*, 791. [CrossRef]
23. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
24. Huang, T.; Li, H.; Zhou, G.; Li, S.B.; Wang, Y. Survey of Research on Instance Segmentation Methods. *J. Front. Comput. Sci. Technol.* **2023**, *17*, 810–825. [CrossRef]
25. Alghamdi, H.; Turki, T. PDD-Net: Plant Disease Diagnoses Using Multilevel and Multiscale Convolutional Neural Network Features. *Agriculture* **2023**, *13*, 1072. [CrossRef]
26. Khan, A.; Sohail, A.; Zahoora, U.; Qureshi, A.S. A survey of the recent architectures of deep convolutional neural networks. *Artif. Intell. Rev.* **2020**, *53*, 5455–5516. [CrossRef]
27. Kim, J.G.; Lee, S.Y.; Lee, I.B. The Development of an LSTM Model to Predict Time Series Missing Data of Air Temperature inside Fattening Pig Houses. *Agriculture* **2023**, *13*, 795. [CrossRef]
28. Shi, X.J.; Chen, Z.R.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 802–810.
29. Salau, J.; Haas, J.H.; Junge, W.; Thaller, G. Determination of Body Parts in Holstein Friesian Cows Comparing Neural Networks and k Nearest Neighbour Classification. *Animals* **2021**, *11*, 50. [CrossRef]
30. Bobbo, T.; Biffani, S.; Taccioli, C.; Taccioli, M.; Cassandro, M. Comparison of machine learning methods to predict udder health status based on somatic cell counts in dairy cows. *Sci. Rep.* **2021**, *11*, 13642. [CrossRef]
31. Themistokleous, K.S.; Sakellariou, N.; Kiossis, E. A deep learning algorithm predicts milk yield and production stage of dairy cows utilizing ultrasound echotexture analysis of the mammary gland. *Comput. Electron. Agric.* **2022**, *198*, 106992. [CrossRef]
32. Zhao, H.; Mao, R.; Li, M.; Li, B.; Wang, M. SheepInst: A High-Performance Instance Segmentation of Sheep Images Based on Deep Learning. *Animals* **2023**, *13*, 1338. [CrossRef]

agriculture

MDPI

Article

Identifying an Image-Processing Method for Detection of Bee Mite in Honey Bee Based on Keypoint Analysis

Hong Gu Lee [1], Min-Jee Kim [2], Su-bae Kim [3], Sujin Lee [3], Hoyoung Lee [4], Jeong Yong Sin [5] and Changyeun Mo [1,5,*]

[1] Department of Interdisciplinary Program in Smart Agriculture, Kangwon National University, Chuncheon 24341, Republic of Korea; hgl@kangwon.ac.kr
[2] Agriculture and Life Sciences Research Institute, Kangwon National University, Chuncheon 24341, Republic of Korea; kim91618@kangwon.ac.kr
[3] Apiculture Division, National Institute of Agricultural Science, 310 Nongsaengmyeng-ro, Deokjin-gu, Jeonju 54875, Republic of Korea; subaekim@korea.kr (S.-b.K.); end0405@korea.kr (S.L.)
[4] Department of Mechatronics Engineering, Korea Polytechnics, 56 Munemi-ro 448 beon-gil, Bupyeong-gu, Incheon 21417, Republic of Korea; hoyoung.yi@gmail.com
[5] Department of Biosystems Engineering, Kangwon National University, Chuncheon 24341, Republic of Korea; kvhffh@kangwon.ac.kr
* Correspondence: cymoh100@kangwon.ac.kr; Tel.: +82-33-250-6494

Abstract: Economic and ecosystem issues associated with beekeeping may stem from bee mites rather than other bee diseases. The honey mites that stick to bees are small and possess a reddish-brown color, rendering it difficult to distinguish them with the naked eye. Objective and rapid technologies to detect bee mites are required. Image processing considerably improves detection performance. Therefore, this study proposes an image-processing method that can increase the detection performance of bee mites. A keypoint detection algorithm was implemented to identify keypoint location and frequencies in images of bees and bee mites. These parameters were analyzed to determine the rational measurement distance and image-processing. The change in the number of keypoints was analyzed by applying five-color model conversion, histogram normalization, and two-histogram equalization. The performance of the keypoints was verified by matching images with infested bees and mites. Among 30 given cases of image processing, the method applying normalization and equalization in the RGB color model image produced consistent quality data and was the most valid keypoint. Optimal image processing worked effectively in the measured 300 mm data in the range 300–1100 mm. The results of this study show that diverse image-processing techniques help to enhance the quality of bee mite detection significantly. This approach can be used in conjunction with an object detection deep-learning algorithm to monitor bee mites and diseases.

Keywords: bee mite; image processing; keypoint detection; image matching

Citation: Lee, H.G.; Kim, M.-J.; Kim, S.-b.; Lee, S.; Lee, H.; Sin, J.Y.; Mo, C. Identifying an Image-Processing Method for Detection of Bee Mite in Honey Bee Based on Keypoint Analysis. *Agriculture* **2023**, *13*, 1511. https://doi.org/10.3390/agriculture13081511

Academic Editors: Zheng Liu, Xiuguo Zou, Wentian Zhang, Xiaochen Zhu, Yan Qian and Yuhua Li

Received: 11 June 2023
Revised: 14 July 2023
Accepted: 20 July 2023
Published: 28 July 2023

1. Introduction

Honeybee is a pollinating insect that maintains the ecosystem. Honeybees possess the ability to produce honey, wax, and royal jelly for beekeeping. However, beekeeping is experiencing a dual crisis of earning-shock and colony collapse disorder due to climate change, pests, and disease [1,2].

Among pests, *Varroa destructor* is the most severe, and may lead to several economic disadvantages compared to other diseases [3]. Bee mites can parasitize larvae and bees, and this may further result in growth decline, wing deformity, abdominal reduction, and death [4]. Methods for detecting bee mites include sugar testing, brood testing, and floor testing, but they have limitations in providing objective, quantitative indicators. Bee mite management is one of the main tasks of beekeeping managers, and research exists to prevent and control it [5,6].

Managing pest inspection in the beehive state is required by beekeeping farmers. Typically, checking by humans involves observation with the naked eye and nonobjective

knowledge. Bee mites have a small size of 1.1 mm × 1.6 mm and are reddish brown. Their color is similar to that of the bee pattern. Hence, their identification is difficult, and a distinct deviation may be present, depending on the skill of the beekeepers. This has led to the need for rapid and objective detection methods.

Considerable visual information can be distinguished during beekeeping. A few examples include honey, bees, queen bees, bee larvae, diseases, and pests. Visual data have been extensively used in computer science analyses. Each class is classified into an image using deep learning. Object detection algorithms are fast and non-destructive approaches to detect bee mites in beecomb images.

Computer vision is the field of computing that uses image data. Computer vision systems have been widely used in machinery, medicine, and agriculture. In precision agriculture, a weak classifier model has been developed using object detection [7]. In a recent study, a banana disease detection model was built using a neural network and transfer learning [8].

Several efforts have been made to achieve more precise beekeeping using computer vision systems. Ngo et al. developed a monitoring system that possessed the ability to count the number of bees at the entrance [9]. Bjerge et al. constructed a measurement system at the entrance to a hive and attempted to monitor bee mites using near-infrared and deep learning [10].

Artificial intelligence used for object detection learns from object keypoints, which are regarded the most important values during image matching, detection, and tracking. Increasing the number of enhanced keypoints helps improve the detection performance.

The keypoint detection algorithm is primarily affected by the measurement environment, even for the same object. The inference performance was changed using a detector. Thus, a keypoint detector must be selected based on its speed and accuracy [11]. Image matching was based on the keypoints of each object. It can connect to similar keypoints. The matching quality is affected by keypoint frequency and location.

This study aims to develop an image processing method in order to improve the quality of bee mite detection. A beecomb measurement system must be developed for image acquisition. A keypoint detector was used to estimate the keypoints. The frequencies and locations of the keypoints were analyzed using a rational image processing method. The image processing methods implemented are color model conversion, histogram normalization, and equalization. The combination of image processing generated 30 analysis cases.

2. Materials and Methods

2.1. Materials and Location

Eight beehives were used for image data measurements at the apiary of the National Institute of Agricultural Sciences in Jeollabuk-do, Republic of Korea, and at a beekeeping farm in Gangwon-do. The honey bee is a Western honey bee (*Apis mellifera*) that is adaptable to the environment and yields high productivity. In another study, two common species of bee mites, *Varroa jacobsoni* and *Varroa destructor*, were measured, which possess dimensions of 1.0630 × 1.5068 mm and 1.1673 × 1.7089 mm, respectively [12]. The observed bee mite had a size of 1.2 × 1.7 mm. Therefore, it was assumed to be *Varroa destructor* (Figure 1).

2.2. RGB Image Acquisition System and Measurement Method

An image acquisition system must be established to define an optimal image-processing method to detect bees and bee mites. Image data were acquired for bees and bee mites in beecombs in a manner similar to that for human inspection.

The image acquisition system was built using a camera, a laptop, and a beecomb supporter (Figure 1). The supporter can directly control this angle. A CMOS-type Blackfly-SGigE camera (FLIR, Wilsonville, OR, USA), with a resolution of 2048 × 1536 pixels, was used in this study.

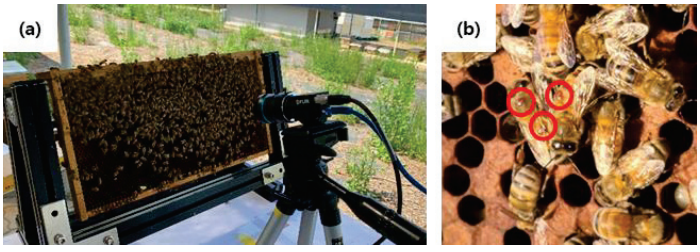


Figure 1. (**a**): image acquisition systems and (**b**): image of bees and bee mites (red circles: bee mites).

The measurement software was developed in Python 3.7. An acquisition area was set for the entire beecomb, which was the same as that visually inspected by humans. The numbers of shots required varied depending on the distance from the camera to the beecomb. The images were measured at five shooting distances at 200-mm intervals from 300 mm to 1100 mm, and the number of measurements per distance is shown in Table 1. The number of image measurements was set to measure one side of the beecomb.

Table 1. Number of image measurements according to imaging distance for measuring the entire beecomb area.

Imaging Distance	300 mm	500 mm	700 mm	900 mm	1100 mm
Number of image measurement (ea)	9	6	4	2	1

Image acquisition of the beecomb with bees and bee mites was performed in apiary. After adjusting the distance between the camera and the beecomb, the angle and position of the camera and supporter were set. The camera and support angle were fixed at 15 degrees in order to prevent light saturation. The aperture and exposure time were manually adjusted in response to changes in environmental factors, such as changing sunlight and weather.

2.3. Bee and Bee Mite Image Dataset

An image was selected and a region of interest was extracted from the measured RGB image data (Figure 2). The selected image contained bee mites, and the region of interest was infested with bees (parasitized by *Varroa destructor*). A total 65 images were extracted at five measurement distances, with 13 images at each level, and 65 images were used in the analysis (Figure 3).

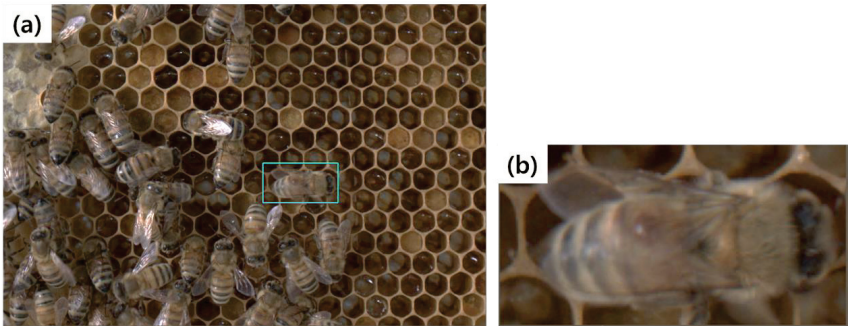


Figure 2. Region of Interest (RoI) cropping: (**a**) original image with the green box meant the extracted area and (**b**) cropped image after the extraction.

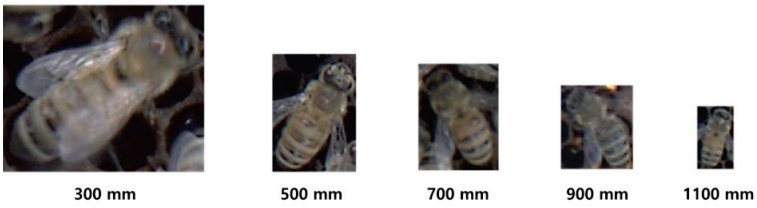


Figure 3. Infested bee with parasitic mite according to measurement distance.

2.4. Optimal Image Processing

To determine the optimal image processing method for detecting bee mites, an analysis based on keypoints and image matching must be performed after image processing. Various image processing methods, such as color model conversion, histogram normalization, and equalization, have been applied to improve the matching rate of bee mites in beekeeping images. There were a total of 30 image processing combinations, and these were applied to the image of the extracted infected bees (i.e., bees parasitized by a bee mite). The image processing methods are shown in Sections 2.4.1 and 2.4.2.

2.4.1. Color Model Conversion

To identify the characteristics of the infected bees that did not appear in the color model of the existing image, five color model conversions were performed. The color models RGB, HSV, Lab, YCrCb, and Gray were representative color models classified according to the mixing method, brightness component, and color-difference component.

According to previous research, RGB is more suitable for neural network learning compared to the one-dimensional value according to the H values of RGB and HSV [13]. A color model refers to 3D array data for expressing colors, and each dimension has a component value for implementing the color. The three dimensions of the RGB model represent the components red, green, and blue, and the HSV model represents the components hue, saturation, and brightness. The YCrCb model consists of brightness and color difference information (Cr and Cb), and the Lab model consists of brightness, red-green, and yellow-blue components. The HSV, YCrCb, and Lab had a common component that represented brightness. The gray color model represents one-dimensional array data. Therefore, gray represents only the intensity of a pixel.

The cvtColor function of OpenCV was used for the color model conversion. The color model conversion equations are as follows (OpenCV, 2022): color model conversion was performed based on a floating-point number with a value between 0 and 1, substituted from the RGB model data. After the model-change formula was applied, it was redefined as 8-bit data, with values ranging from 0 to 255. In the case of HSV, YCrCb, and Gray, they were converted at once in a specific way corresponding to the coefficients defined, as in Equations (1), (3) and (4). The Lab case was converted to the color model XYZ, as shown in Equation (2):

$$V = \max(R, G, B)\, S = \begin{Bmatrix} \frac{V-\min(R,G,B)}{V} \\ 0 \end{Bmatrix} \begin{matrix} \text{if } V \neq 0 \\ \text{otherwise} \end{matrix} \quad (1)$$

$$H = \begin{Bmatrix} 60(G-B) \,/\, (V-\min(R,G,B)) \\ 120 + 60(B-R)/(V-\min(R,G,B)) \\ 240 + 60(R-B)/(V-\min(R,G,B)) \\ 0 \end{Bmatrix} \begin{matrix} \text{if } V = R \\ \text{if } V = G \\ \text{if } V = B \\ \text{if } R = G = B \end{matrix}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

$$X = \frac{X}{X_n}, \text{ where } X_n = 0.950456$$

$$Z = \frac{Z}{Z_n}, \text{ where } Z_n = 1.088754$$

$$L = \begin{Bmatrix} 116 * Y^{\frac{1}{3}} - 16 \\ 903.3 * Y \end{Bmatrix} \begin{matrix} \text{for } Y > 0.008856 \\ \text{for } Y \leq 0.008856 \end{matrix}$$

$$a = 500(f(X) - f(Y)) + 128$$

$$b = 200(f(X) - f(Z)) + 128$$

$$f(t) = \{ \begin{matrix} t^{1/3} \\ 7.787t + 16/116 \end{matrix} \} \begin{matrix} \text{for } t > 0.008856 \\ \text{for } t \leq 0.008856 \end{matrix}$$

$$Y = 0.299R + 0.587G + 0.114B \quad (3)$$

$$Cr = (R - Y)0.713 + 128$$

$$Cb = (B - Y)0.564 + 128$$

$$Gray = 0.299R + 0.587G + 0.114B \quad (4)$$

$$R = \text{PixelofintensityRchannel}$$

$$G = \text{PixelofintensityGchannel}$$

$$B = \text{PixelofintensityBchannel}$$

2.4.2. Histogram Normalization and Equalization

An image acquisition experiment was performed outdoors according to the same condition as that of a visual inspection by a beekeeper. In outdoor image acquisition experiments, the intensity of sunlight changed depending on factors such as measurement time, clouds, and weather. Sunlight variations caused deviations in the measurement data. In other words, measurement errors, such as sunlight, deviation of appropriate exposure time, and aperture value, may occur. Histogram calibration may reduce further deviations due to changes in light intensity. Histogram normalization and equalization was one of the methods used to calibrate the intensity of each component. Both histogram correction methods could normalize data and enhance contours and contrast.

The minimum–max normalization was calculated using Equation (5). In the Equalization method, there were various algorithms, such as Global Histogram Equalization (GHE), Local Histogram Equalization (LHE), and Dynamic Histogram Equalization (DHE) [14]. Histogram equalization was performed using the cumulative distribution in Equation (6). Global Histogram Equalization and Contrast-Limited Adaptive Histogram Equalization (CLAHE), which are calculated by dividing the image into a grid, were selected for the equalization method.

In this study, normalization (not applied, applied) and equalization (not applied, GHE, and CLAHE) were applied to color-converted images for a detailed comparison of the effects of histogram correction. The color image consisted of three channels. The normalized channels differed for each color model. In the HSV, YCrCb, and Lab color models, normalization was applied to brightness components, and the RGB and the Gray color models were applied to all components:

$$I_{normalization} = \frac{\left(I_{original} - Min_{original}\right) * 255}{\left(Max_{original} - Min_{original}\right)} \quad (5)$$

$I_{normalization}$: Normalized image
$I_{original}$: Original image
$Max_{original}$: Maximum pixel value of original image
$Min_{original}$: Minimum pixel value of original image

$$H'(v) = \text{round}\left(\frac{cdf(v) - cdf_{min}}{(M * N) - cdf_{min}} * (L - 1)\right) \quad (6)$$

$H'(v)$: Equalized Histogram
v: Value of pixel
round(v): Rounds Function
cdf(v): Histogram cumulative function
cdf_{min}: Minimum cumulative value, usually 1
$M * N$: Resolusion of image,(M: Width, N: Height)
L: Range of pixel value, 256

2.5. Keypoint Detection Algorithm of Bees and Bee Pests

A keypoint is the point at which an object can be distinguished locally. This is used as a matching point for object matching, detection, and tracking. In addition, as an essential factor, the keypoint must be derived in order to recognize an object or structure using a computer. Therefore, as recognition points for objects such as honeybees and bee mites, the frequency and the location accuracy of keypoints can be used to evaluate the quality of the images to which image processing was applied.

The keypoint detection algorithm should be selected according to the data characteristics, and both its speed and its accuracy may vary depending on the analysis hardware [11]. There were research to identify bee pollen with RGB image and the vector of locally aggregated descriptors encoded by the Scale-Invariant Feature Transform (SIFT) keypoint detection algorithm [15]. Oriented FAST and Rotated BRIEF (ORB) are keypoint detection algorithms based on the Features from Accelerated Segment Test (FAST) that is applied to real-time systems and the Binary Robust Independent Elementary Features (BRIEF) with rotation invariance [16]. The keypoints of each patch were detected using FAST, and efficient points were calculated among the detected keypoints based on the BRIEF descriptor. Among four keypoint detection algorithms (BRISK, SIFT, SURF, and ORB), the ORB algorithm showed the best performance in terms of evaluation of feature point frequency, calculation efficiency, matching efficiency, and detection speed [17]. As the distortion of an image varies depending on the type of camera or lens, distortion correction is necessary. A comparison of the detection and matching performance of SIFT, SURF, and ORB for distortion based on data with 30% salt-and-pepper noise compared to the original showed that the ORB algorithm was the best [18]. Thus, the ORB algorithm was applied to data with minimal image-warping distortion.

2.6. Performance of Keypoint

This study aimed to investigate the optimal measurement distance and image-processing method for honeybee and bee mite recognition. The keypoint detection performance was based on frequency and location accuracy analysis for each image processing step.

2.6.1. Analysis of Keypoint Location and Frequency

The keypoints detected through the ORB are composed of an object that stores keypoint information and an object that stores descriptor information. The stored keypoint information was stored, and it had the following values: pt, size, angle, response, octave, and class_id. (here, pt denotes the location of a feature point). Therefore, by contrasting the values of pt, the regions of bees, and bee mites, it is possible to determine the frequency of keypoints that would actually be used for object matching.

The location information of the bee mites in the images is labeled in a boxed JSON format. Bee mite region information can also be used to extract the mite area within an image.

The pt component of the keypoint was obtained from both the original and each processed image. The pt components of the extracted keypoints were compared with the coordinates of the bee mites, and the number of valid keypoints for bee mite identification for each image processing step was calculated.

2.6.2. Image Matching Algorithm

Image matching algorithms could match detected keypoints in two images. Matching performance was affected by the quality of the keypoints in the image. Image matching was performed to verify the performance of the detected bee and honeybee keypoints and to compare the changes according to the image processing method.

The BFMatcher function in OpenCV was used for image matching. The BFMatcher is an algorithm that uses a brute-force match to compute all matchable keypoints in order to produce good results. The matching parameters for the brute-force operations were NORM_HAMMING, which uses the Hamming distance, and CrossCheck, which determines whether the matching results in both directions are the same. The image matching result is represented as Dmatch with four components: queryIdx, trainIdx, imgIdx, and distance. QueryIdx and trainIdx were the indices of the keypoints that are detected in the images used for matching. The imgIdx is the component that is used when matching multiple images, and the distance is the matching value between the keypoint vectors. A small matching distance indicates a high similarity. The components of the matching results are sorted in order of decreasing distance to select the top-matching objects with high matching similarity.

In this study, image matching was implemented based on the original image, and an image with histogram normalization and equalization. The top ten matching objects were selected based on the distance component to compare the matching performance. The selected matching objects are checked for anomalous matches. An abnormal match is observed when different points on an object are matched.

The overall process of applying image processing, keypoint detection, and image matching to 150 conditions under five measurement distance conditions and 30 image processing combinations is shown in the flowchart in Figure 4.

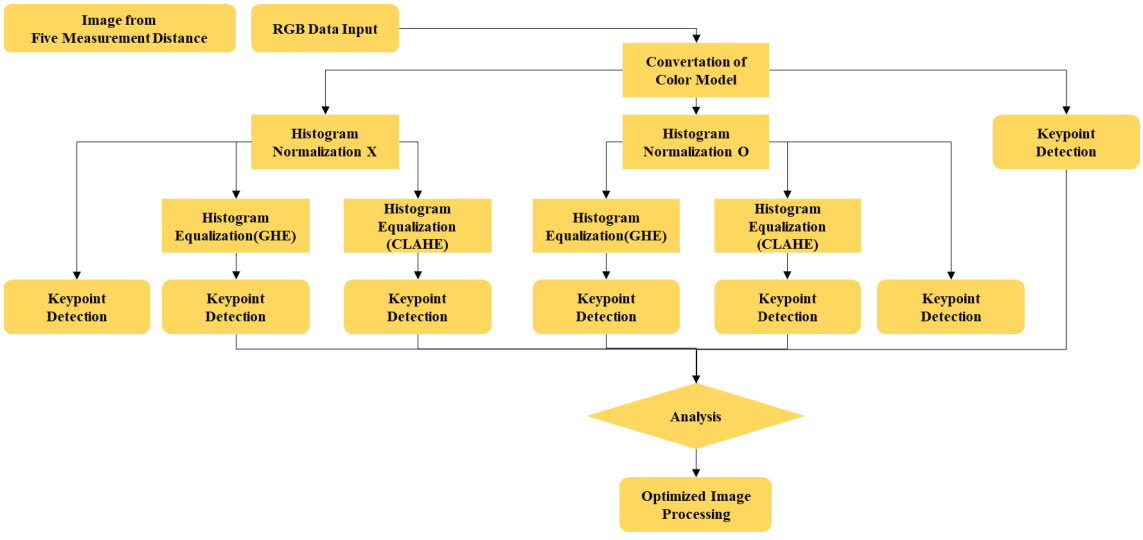


Figure 4. Flowchart of optimal image processing for feature point detection according to color and image correction.

3. Results and Discussion

3.1. Color Model Conversion and Histogram Analysis of Bee and Bee Mite Image

To identify the optimal color model for detecting honeybee and bee mite keypoints, RGB images were converted into four color models (Gray, HSV, Lab, and YCrCb). The image and histogram analyses of the original RGB image and the converted color models are shown in Figure 5. The average intensity of the 13 images for each color model was used for histogram analysis.

The measured image data exhibited values in the range 0–255. The RGB color model analysis showed that values 0–2 were not present in the R and G channels, and values 0–1 were not present in the B channel. In addition, the values 197–255 for G and those for 181–255 for B were not present. In the HSV color model, values 179–255 in the H channel, 224–255 in the S channel, and 0–3 in the V channel were not present.

In the LAB, the distribution was skewed toward values between 120 and 135 in channels A and B, with no values between 0 and 1 and 200 and 255 in channel L; 0 and 105 and 159 and 255 in channel A; and 0 and 78 and 166 and 255 in channel B. YCrCb had a distribution in which the frequencies of the Cr and Cb channels were clustered around values between 120 and 135, with 0–2 and 192–255 for the Y channel; 0–98 and 155–255 for the Cr channel; and 0–99 and 173–255 for the Cb channel. The single-channel color model, gray, had no values between 0 and 2 and 192 and 255.

The distribution of the color values tended to be skewed toward some specific values rather than the full range, and some values were empty. Therefore, normalization was required to ensure that the color component values were evenly distributed over the range of 0–255.

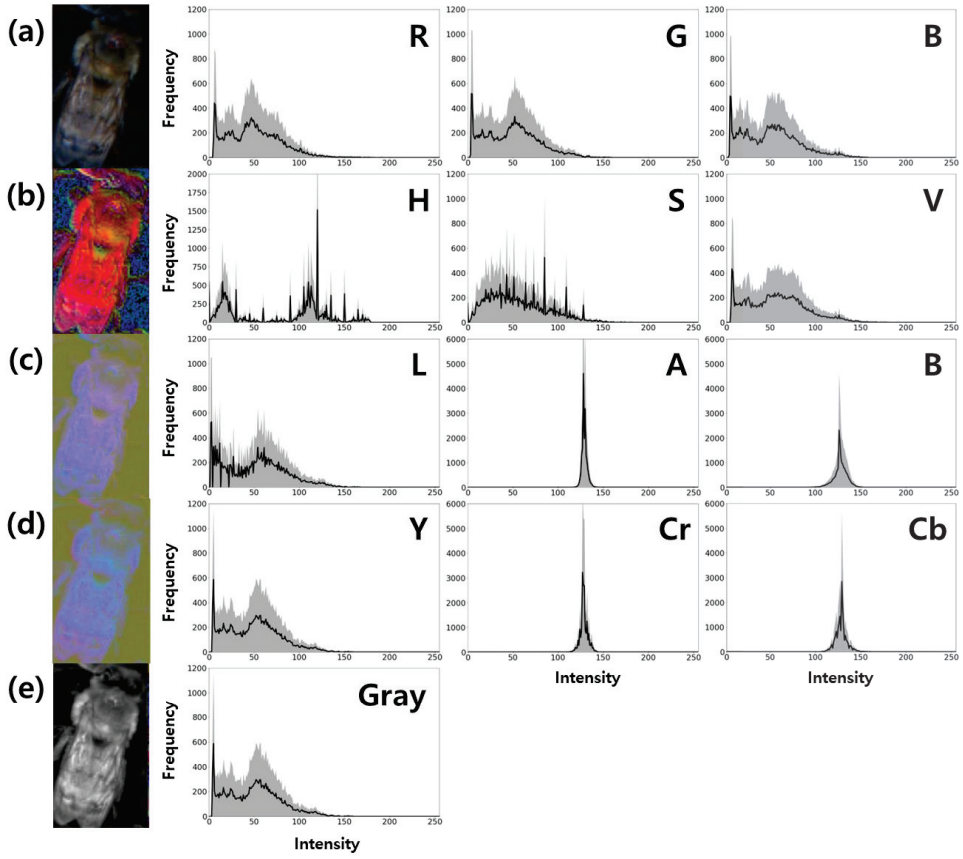


Figure 5. Average histogram of each channel by color models: (**a**) RGB, (**b**) HSV, (**c**) Lab, (**d**) YCrCb, and (**e**) Gray.

3.2. Histogram Normalization and Equalization for Bee and Bee Mite Image

Honeybee and bee mite images were subjected to histogram normalization and equalization. After image processing, each beekeeping image was converted into 30 images, including the original image. As shown in Figure 6, when histogram normalization was applied, the pixel values were distributed in the range 0–255 and the contrast was improved.

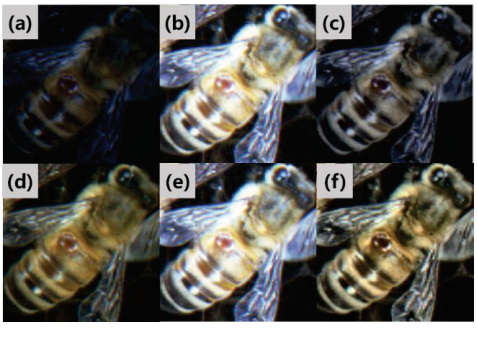


Figure 6. Histogram equalization processing image: (**a**) original, (**b**) GHE, (**c**) CLAHE, (**d**) normalized image, (**e**) normalized GHE, and (**f**) normalized CLAHE.

The normalization algorithm required the maximum and minimum values of the data (Equation (1)). If the measured data possessed values in the range 0–255, the normalization algorithm might not work correctly. Figure 7 shows the normal and abnormal operations of histogram normalization. If the values at either end of the distribution are 0 or 255, the normalization algorithm will not work properly, and contrast improvement cannot be expected.

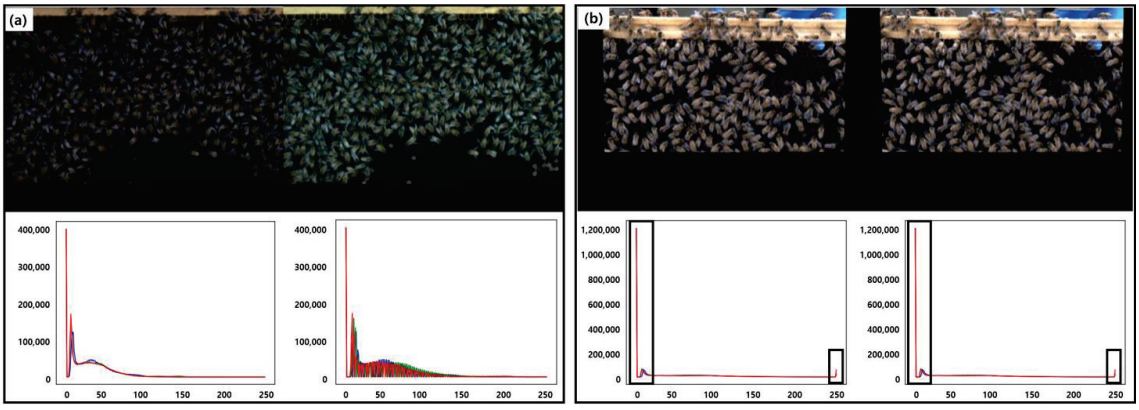


Figure 7. (**a**) Normal operation and (**b**) abnormal operation of histogram normalization processing. When values existed between 0 and 255, as in (**b**), histogram normalization did not work properly.

As shown in Figure 8, histogram equalization resulted in a relative improvement in the contrast compared to the original. The equalization of the entire basis and CLAHE yielded different results. The CLAHE method divides an image into grids and equalizes each grid. This enhances the unique color of bees and cells. However, global equalization is applied according to the entire image, which further improves the overall brightness.

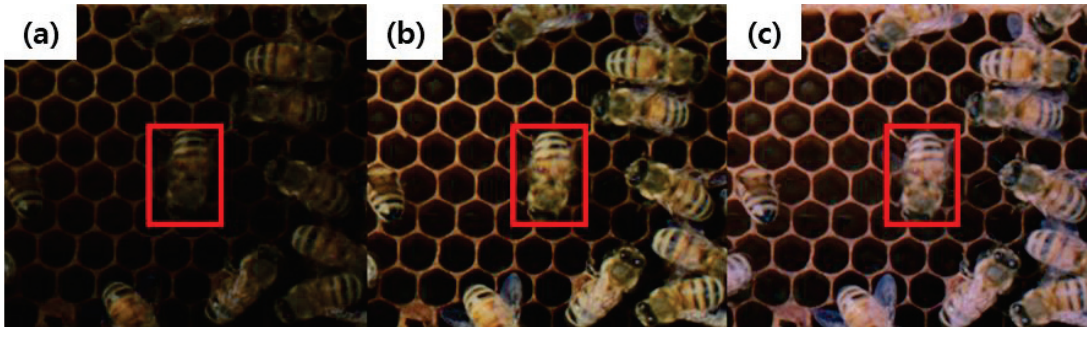


Figure 8. Histogram equalization processing image: (**a**) original, (**b**) CLAHE, and (**c**) GHE. The red boxes in each image represented bees infected with bee mite.

3.3. Detection of Keypoints in Bee and Bee Mite Image

The ORB was applied to the beekeeping images in 150 different cases to detect the keypoints. The results showed that most keypoints were detected at a shooting distance of 300 mm. The number of keypoints tended to decrease as the shooting distance increased (Figure 9). These results suggest that resolution-dependent measurement distances should be considered when recognizing bees using images.

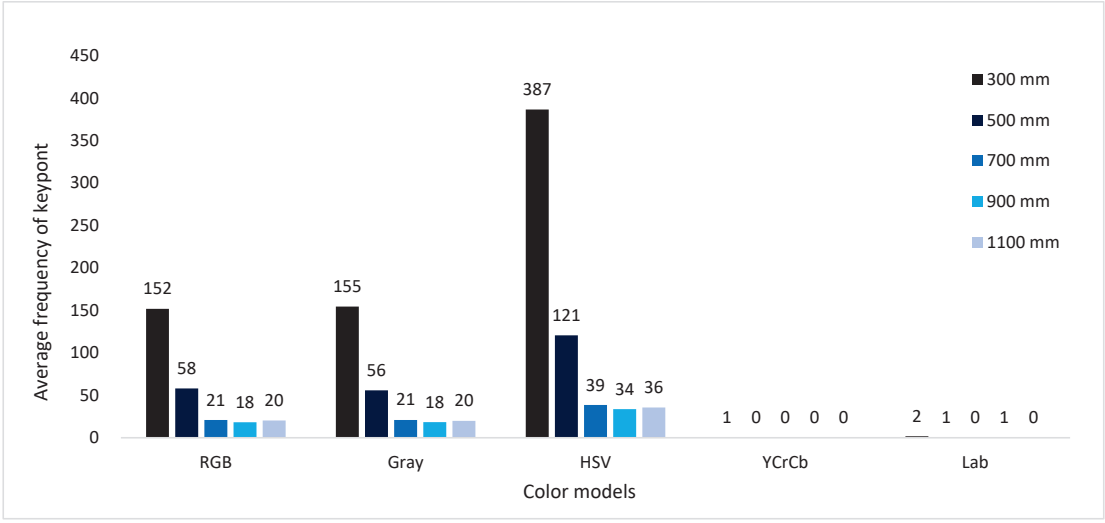


Figure 9. Number of average keypoints of bees for each imaging distance in original and each color model image.

The numbers of keypoints detected were compared using a color model. The average number of detected keypoints in RGB and Gray were 152 and 155, and the mean error rate and standard deviation were 1.03% and 1.39%, respectively. Similar performance results were obtained, but the Gray model required a color model conversion from the original data.

In the case of the YCrCb and Lab color models, up to four features were detected in the images that were measured at a distance of 300 mm, with average detection frequency of one and two. Therefore, it is not ideal to use the color models Lab and YCrCb to analyze bees and bee mites.

The HSV color model detected the most keypoints at all measurement distances. However, for HSV, the detected keypoints were often not located in the bee or bee mite zones (Figure 10b). An increase in non-object keypoints may result in a decrease in the matching rate of the bee mites. The keypoints for object recognition must be used accurately as matching points, otherwise inaccurate recognition may occur. Based on the comparison of the keypoint detection of five different color models (RGB, HSV, Lab, YCrCb, and Gray), we determined that the RGB color model was suitable for beekeeping monitoring.

The average keypoint detection performance increased by 44%, from 278 to 398, using the normalization algorithm (Table 2). Among the GHE and CLAHE methods used for equalization, a higher number of keypoints was detected using GHE. However, the GHE-detected keypoints were not specific to bees and bee mites (Figure 10). Therefore, CLAHE is more suitable than GHE as an image-processing method for bee-monitoring data.

Table 2. Average number of keypoints of the RGB image according to imaging distance and image processing.

	300 mm	**500 mm**	**700 mm**	**900 mm**	**1100 mm**
Original	67	21	2	1	0
Normalization	276	50	30	12	1
GHE	456	93	50	31	1
CLAHE	502	105	34	24	1
Normalization and GHE	456	93	50	31	1
Normalization and CLAHE	758	129	66	37	2

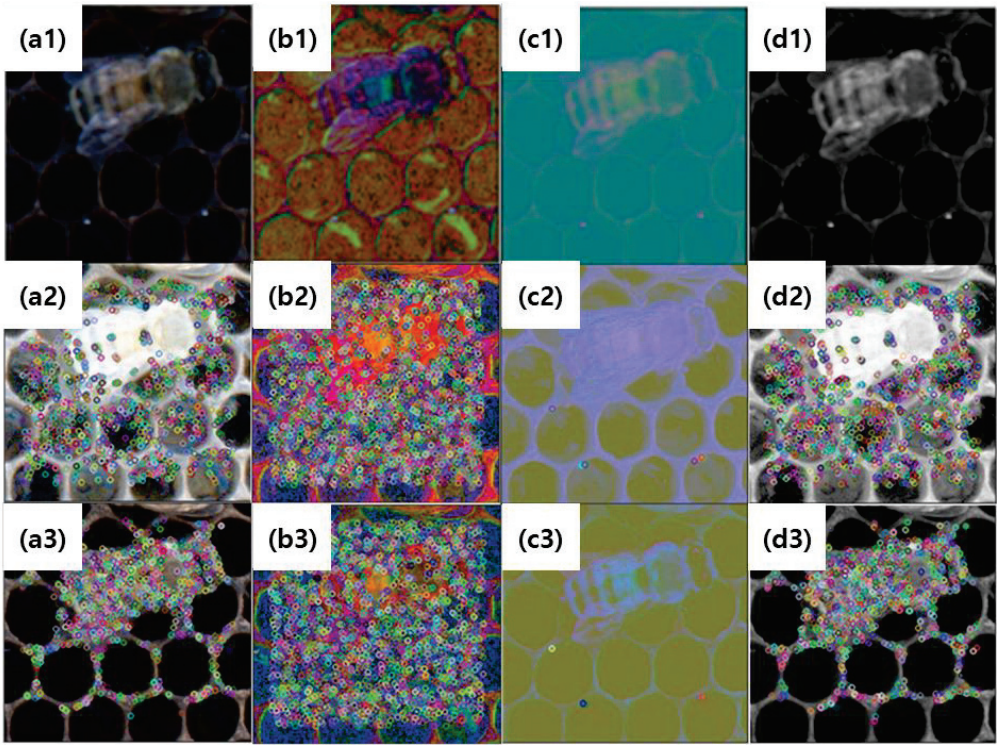


Figure 10. Keypoint detection images for original images of four color models (RGB (**a1**), HSV (**b1**), Lab (**c1**), Gray (**d1**)), GHE processed image (**a2**–**d2**), and CLAHE processed image (**a3**–**d3**).

Based on the original data, an average of two keypoints was detected at a shooting distance of 700 mm, an average of one keypoint at 900 mm, and zero keypoints at 1100 mm. Compared with the 300-mm image data, the number of keypoints detected at a distance of 500 mm was reduced by 69%, from 67 to 21. For images with measurement distances of 700, 900, and 1100 mm, the detection performance decreased by 97%, 99%, and 100%, respectively, compared to the 300-mm image. In particular, the images measured at a distance of 1100 mm with only one or two keypoints were detected even after image processing. Images measured at distances greater than 1100 mm were not available for analysis.

The keypoint detection performances were compared by applying 30 image-processing methods. Histogram normalization and equalization can help improve the image contrast and subsequently increase the number of keypoints. However, normalization may or may not be applicable depending on the histogram distribution, and, thus, equalization should also be applied. Consequently, the optimal image processing conditions were the application of histogram normalization and histogram equalization (CLAHE) to the RGB color model. The RGB color model can be effective for analysis because it can represent the reddish brown of bee mite more effectively than other color models.

3.4. Validation and Histogram Analysis Based on Optimal Image Processing

To verify the performance of the optimal image-processing conditions, the frequency of the keypoints was analyzed. Performance was validated using a bee image that was captured at a distance of 300 mm. When normalization was applied, the distribution of pixels was split between 0 and 255 and the number of keypoints increased by 399% (Figure 11b). Equalization improved the number of keypoints by 269% over the normalized data by spreading out the distributions concentrated on a few values (Figure 11c). By integrating

an RGB color model, normalization, and equalization (CLAHE) to bee and bee mite images, the quality was enhanced considerably. The processed images contained more keypoints. This image-processing method improved the recognition rate of honeybees and mites. Image processing methods that affect an image locally were more effective than methods that affect the entire image. Image processing to homogenize an image to distinguish objects sharpened the image. Objects in images with increased sharpness have more points that are distinct (darker or lighter) from their surroundings, and the frequency of feature points may increase.

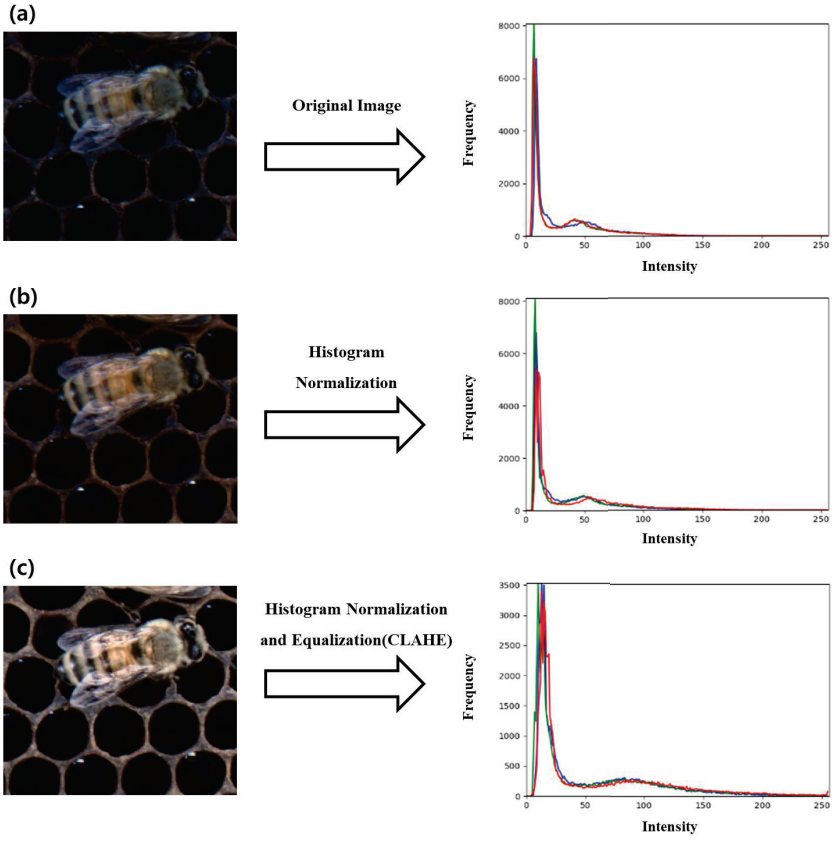


Figure 11. Histogram distribution change original image (**a**), after histogram normalization (**b**), and normalization with CLAHE (**c**). Red, green, and blue lines represent the red, green, and blue conponent of the RGB channels, respectively.

3.5. Analyzing Frequency of Keypoints in Bee Mite

The average values of the numbers of keypoints in the original and processed images are shown in Figure 12. When image processing was applied to the data that were measured at a distance of 300 mm, the number of keypoints was the highest, with an average of 31 in the bee mite area. This was approximately 340% higher than the frequency before image processing was applied. Through optimal image processing, 500 mm, 700 mm, and 900 mm data showed an increase of 380%, 1733%, and 2400% in keypoints, respectively, for bee mites. The data measured at a distance of 1100 mm did not detect any keypoints in the mite area. Among beekeeping objects, such as bees, queens, and workers, the bee mite belongs to the small scale. Therefore, the increase in keypoints of bee mites was noteworthy. This may be a clue to solving the problem of simultaneous recognition of small and large objects.

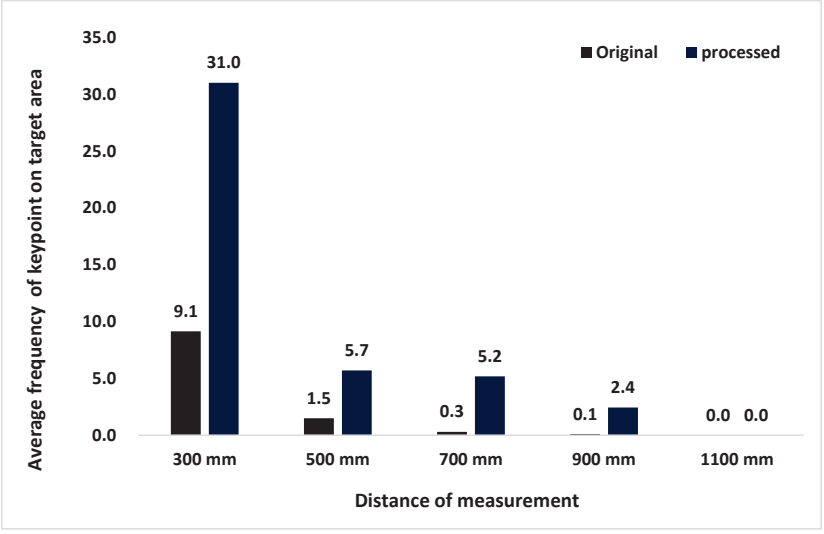


Figure 12. Number of average keypoints of bee mites for each image measurement distance in original image and image processed with histogram normalization and histogram equalization (CLAHE). The numbers above the bars were the average of keypoints.

3.6. Bee Mite Image Matching Results—Comparison of Top Ten Matching Objects

We checked whether the optimal image processing method could improve bee mite detection performance. The image matching with the coordinates of the bee mite was used for verification.

Image matching was used for the bee mite region. If the matching point was not correct, it was judged as an abnormal match (Figure 13).

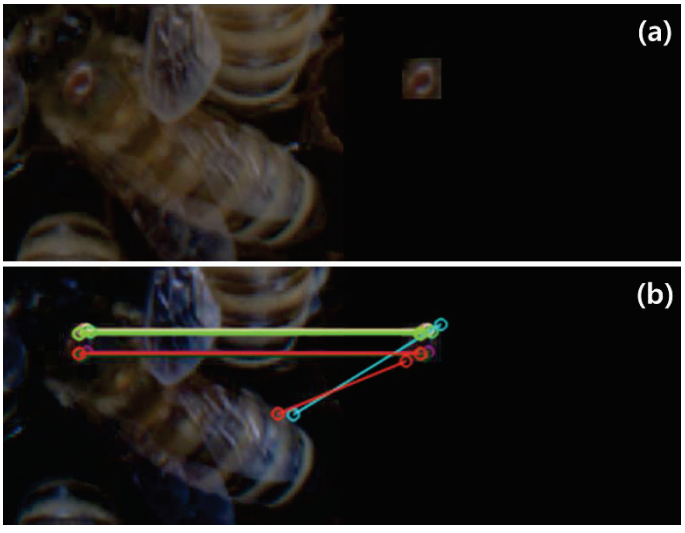


Figure 13. Normal and abnormal matching points in image matching between bee (**left**) and bee mite (**right**): (**a**) source image and (**b**) matching result. Each line connects the matched keypoints. The lines provide a visual representation of the normal and abnormal match between the bee mite on the bee and the bee mite image.

The original data measured at a distance of 300 mm could not generate 10 matching objects, in accordance with the image. Abnormal matches were 3.7 (50%) based on an average of 7.4 matching objects. Given image processing, the top ten matching objects were generated from all images. For the processed images, an average of 3.7 (38%) abnormal matches were obtained in the top ten matching objects. Thus, through optimal image processing, the matching performance could be improved by 12% based on images with a measurement distance of 300 mm (Figure 14).

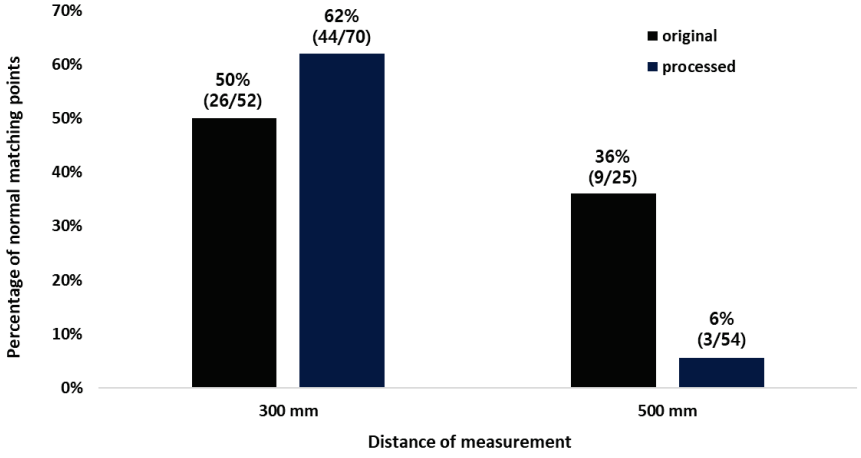


Figure 14. Normal matching result of bee mite of image data measured at distances of 300 and 500 mm.

For the data measured at a distance of 500 mm, cases were present that produced fewer than 10 matching objects for each image. On average, 1.5 (64%) of the abnormal matchings were found in the 2.3 matching objects. Even with image processing, an image with a distance of 500 mm generated fewer than ten matching objects. Image processing increased the average number of matching objects to 4.9 but resulted in abnormal matching of 4.6 (94.4%). In other words, for a beecomb RGB image in a case where the measurement distance is longer than 500 mm, the bee mite-matching performance may be degraded.

For the images that were measured at distances greater than 700 mm, the object matching algorithm did not work regardless of the image processing. Given a camera with a resolution of 2048 $\times$ 1536, bee mite RGB data measured at a distance greater than 700 mm could not be used for image matching.

4. Conclusions

Bee mites cause more economic damage than other honeybee pests and diseases. Bee mites are small and reddish-brown in color, making it difficult to distinguish them from bees when attached to them. This has generated the need for technology that can objectively and quickly test for *Varroa* mite outbreaks. Image-based analytics, such as object detection, possess the potential to recognize bee mites. However, their small size and protective color may be a problem for computer vision systems.

Therefore, in this study, we applied image processing, keypoint detection, and image matching algorithms to images of bees and bee mites to improve the matching rate of bee mites. The frequency and location of the keypoints were analyzed and the quality of the matched objects was evaluated accordingly.

The analysis results for 30 combinations of image processing methods, including color model conversion, histogram normalization, histogram equalization, and five measurement distances, are as follows: applying normalization and equalization (CLAHE) based on the RGB color model to bee and bee mite images resulted in better keypoint detection by

reinforcing the image quality. The effectiveness of the optimal image processing method was observed through the data that were measured at 300 mm of the 300–1100 mm measurement distance, with improved keypoint detection and matching performance. Regardless of image processing, it was difficult to match images to bee mites at the measurement distance of 700 mm or more. At measurement distances of 500 mm, image matching of bee mite images was possible, but with a high mismatch rate.

The improved matching quality can lead to improved detection performance of deep learning-based algorithms. The optimal image processing method and measurement distance for identifying bee mites can be used to simultaneously detect beekeeping objects with different sizes and shapes. The results of this study can be used as basic supporting data for recognizing bee mites, which are small objects, and bees, which are relatively large objects. In future research, we would like to apply this image processing condition to deep learning-based object detection to develop a model for identifying bee mites and beekeeping objects, such as bee, larva, cell, egg.

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

Funding: This study was supported by the Rural Development Administration as "Cooperative Research Program for Agriculture Science and Technology Development [Project Nos. PJ01582601, RS-2023-00232224]".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors have no conflicting financial or other interest.

References

1. Ellis, J. The honey bee crisis. *Outlooks Pest Manag.* **2012**, *23*, 35–40. [CrossRef]
2. Eliash, N.; Mikheyev, A. Varroa mite evolution: A neglected aspect of worldwide bee collapses? *Curr. Opin. Insect Sci.* **2020**, *39*, 21–26. [CrossRef] [PubMed]
3. Boecking, O.; Genersch, E. Varroosis—The ongoing crisis in bee keeping. *J. Verbr. Lebensmittelsicherh.* **2008**, *3*, 221–228. [CrossRef]
4. Sammataro, D.; Gerson, U.; Needham, G. PARASITIC MITES OF HONEY BEES: Life history, implications, and impact. *Annu. Rev. Entomol.* **2000**, *45*, 519–548. [CrossRef] [PubMed]
5. Roth, M.A.; Wilson, J.M.; Tignor, K.R.; Gross, A.D. Biology and management of *Varroa destructor* (Mesostigmata: Varroidae) in Apis mellifera (Hymenoptera: Apidae) colonies. *J. Integr. Pest Manag.* **2020**, *11*, 1. [CrossRef]
6. Jack, C.J.; Ellis, J.D. Integrated pest management control of *Varroa destructor* (Acari: Varroidae), the most damaging pest of (*Apis mellifera* L. (Hymenoptera: Apidae)) colonies. *J. Insect Sci.* **2021**, *21*, 6. [CrossRef]
7. Salazar-Gomez, A.; Darbyshire, M.; Gao, J.; Sklar, E.I.; Parsons, S. Towards Practical Object Detection for Weed Spraying in Precision Agriculture. 2021. Available online: http://arxiv.org/abs/2109.11048 (accessed on 22 September 2021).
8. Selvaraj, M.G.; Vergara, A.; Ruiz, H.; Safari, N.; Elayabalan, S.; Ocimati, W.; Blomme, G. AI-powered banana diseases and pest detection. *Plant Methods* **2019**, *15*, 1–11. [CrossRef]
9. Ngo, T.N.; Wu, K.C.; Yang, E.C.; Lin, T.T. A real-time imaging system for multiple honey bee tracking and activity monitoring. *Comput. Electron. Agric.* **2019**, *163*, 104841. [CrossRef]
10. Bjerge, K.; Frigaard, C.E.; Mikkelsen, P.H.; Nielsen, T.H.; Misbih, M.; Kryger, P. A computer vision system to monitor the infestation level of *Varroa destructor* in a honeybee colony. *Comput. Electron. Agric.* **2019**, *164*, 104898. [CrossRef]
11. Liu, C.; Xu, J.; Wang, F. A review of keypoints' detection and feature description in image registration. In *Scientific Programming*; Hindawi Publishing Limited: London, UK, 2021; Volume 2021, pp. 1–25. [CrossRef]
12. Anderson, D.L.; Trueman, J.W.H. *Varroa jacobsoni* (Acari: Varroidae) is more than one species. *Exp. Appl. Acarol.* **2000**, *24*, 165–189. [CrossRef] [PubMed]
13. Dembski, J.; Szymański, J. Bees detection on images: Study of different color models for neural networks. In *Lecture Notes in Computer Science*; Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics; Speinger: Berlin, Germany, 2019; Volume 11319, pp. 295–308. [CrossRef]

14. Abdullah-Al-Wadud, M.; Hasanul Kabir, M.; Ali Akber Dewan, M.; Chae, O. A Dynamic Histogram Equalization for Image Contrast Enhancement. *IEEE Trans. Consum. Electron.* **2007**, *53*, 593–600. [CrossRef]
15. Stojnić, V.; Risojević, V.; Pilipović, R. Detection of pollen bearing honey bees in hive entrance images. In Proceedings of the 17th International Symposium on INFOTEH-JAHORINA, INFOTEH 2018, Jahorina, Bosnia and Herzegovina, 21–23 March 2018; Volume 2018, pp. 1–4. [CrossRef]
16. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571. [CrossRef]
17. Tareen, S.A.K.; Saleem, Z. *A Comparative Analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK*; IEEE Publications: Sukkur, Pakistan, 2018. [CrossRef]
18. Karami, E.; Prasad, S.; Shehata, M. Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images. *arXiv* **2017**, arXiv:1710.02726.

agriculture

MDPI

Article

Soybean-MVS: Annotated Three-Dimensional Model Dataset of Whole Growth Period Soybeans for 3D Plant Organ Segmentation

Yongzhe Sun [1], Zhixin Zhang [1], Kai Sun [1], Shuai Li [1], Jianglin Yu [1], Linxiao Miao [1], Zhanguo Zhang [2], Yang Li [2], Hongjie Zhao [2], Zhenbang Hu [3], Dawei Xin [3], Qingshan Chen [3] and Rongsheng Zhu [2,*]

[1] College of Engineering, Northeast Agricultural University, Harbin 150030, China; 18545155636@163.com (Y.S.); s210702060@neau.edu.cn (Z.Z.); kaisunneau@163.com (K.S.); zz-3280@163.com (S.L.); jianglinyuneau@163.com (J.Y.); s220701045@neau.edu.cn (L.M.)

[2] College of Arts and Sciences, Northeast Agricultural University, Harbin 150030, China; neauzzg@neau.edu.cn (Z.Z.); code77@163.com (Y.L.); zhaohongjie77@neau.edu.cn (H.Z.)

[3] College of Agriculture, Northeast Agricultural University, Harbin 150030, China; zbhu@neau.edu.cn (Z.H.); dawxin@neau.edu.cn (D.X.); qschen@neau.edu.cn (Q.C.)

* Correspondence: rshzhu@126.com; Tel.: +86-133-9451-6944

Abstract: The study of plant phenotypes based on 3D models has become an important research direction for automatic plant phenotype acquisition. Building a labeled three-dimensional dataset of the whole growth period can help the development of 3D crop plant models in point cloud segmentation. Therefore, the demand for 3D whole plant growth period model datasets with organ-level markers is growing rapidly. In this study, five different soybean varieties were selected, and three-dimensional reconstruction was carried out for the whole growth period (13 stages) of soybean using multiple-view stereo technology (MVS). Leaves, main stems, and stems of the obtained three-dimensional model were manually labeled. Finally, two-point cloud semantic segmentation models, RandLA-Net and BAAF-Net, were used for training. In this paper, 102 soybean stereoscopic plant models were obtained. A dataset with original point clouds was constructed and the subsequent analysis confirmed that the number of plant point clouds was consistent with corresponding real plant development. At the same time, a 3D dataset named Soybean-MVS with labels for the whole soybean growth period was constructed. The test result of mAccs at 88.52% and 87.45% verified the availability of this dataset. In order to further promote the study of point cloud segmentation and phenotype acquisition of soybean plants, this paper proposed an annotated three-dimensional model dataset for the whole growth period of soybean for 3D plant organ segmentation. The release of the dataset can provide an important basis for proposing an updated, highly accurate, and efficient 3D crop model segmentation algorithm. In the future, this dataset will provide important and usable basic data support for the development of three-dimensional point cloud segmentation and phenotype automatic acquisition technology of soybeans.

Keywords: 3D reconstruction; the whole growth period; soybean; point cloud segmentation; dataset

Citation: Sun, Y.; Zhang, Z.; Sun, K.; Li, S.; Yu, J.; Miao, L.; Zhang, Z.; Li, Y.; Zhao, H.; Hu, Z.; et al. Soybean-MVS: Annotated Three-Dimensional Model Dataset of Whole Growth Period Soybeans for 3D Plant Organ Segmentation. *Agriculture* **2023**, *13*, 1321. https://doi.org/10.3390/agriculture13071321

Academic Editors: Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 7 May 2023
Revised: 23 June 2023
Accepted: 24 June 2023
Published: 28 June 2023

1. Introduction

With the continuous development of plant phenomics, three-dimensional plant phenotypic analysis has become a challenging research topic. Using deep learning for point cloud segmentation is the foundation of crop phenotype measurement and breeding. The common point cloud datasets used for training are scarce and difficult to obtain, and there is no commonly used basic data for organ instance segmentation for phenotype extraction. In addition, due to the complex structure of plants, the data annotation work needs considerable manual processing. A well-labeled dataset is essential for the segmentation of plant point clouds using deep learning. In order to obtain a well-labeled dataset, it should have

the following characteristics: complete plant structure, high precision, and the ability to cover multiple varieties and growth periods. Consequently, building a labeled crop plant point cloud dataset of the entire growth period is a key step toward achieving accurate crop point cloud segmentation using deep learning.

Although the lack of well-labeled 3D plant datasets limits the further progress of plant point cloud segmentation [1], many scholars have made significant advancements in building plant point cloud segmentation datasets in recent years. Zhou et al. [2] manually segmented the 3D point cloud data of soybean plants and gave each point a real label. This was used as the training set for point cloud segmentation and real ground data for evaluating segmentation accuracy using machine learning methods. Li et al. [3] used the MVS-Pheno platform to obtain multi-view images and point clouds of corn plants in the study of organ-level point cloud automatic segmentation of corn branches based on high-throughput data acquisition and deep learning. At the same time, the research team developed a data annotation tool kit specifically for corn plants, called Label3DMatch, and annotated the data to ultimately build a training dataset. Conn et al. [4] planted tomatoes, tobacco, and sorghum under the five growth conditions of ambient light, shade, high temperature, strong light, and drought, and performed 3D laser scanning (311 tomato scans, 105 tobacco scans, and 141 sorghum scans) on the plant stem structure during 20–30 days' development. A 3D plant dataset was constructed after summarizing the species, conditions, and time points. Li et al. [5] used this original dataset and manually marked the semantic labels belonging to stems and leaves using the semantic segmentation editor (SSE) tool and established a well-labeled point cloud dataset for plant stem leaf semantic segmentation and leaf instance segmentation. Hideaki et al. [6] proposed a 3D phenotype platform that can measure plant growth and environmental information in a small indoor environment to obtain plant image datasets. In addition, annotation tools were introduced, which can manually, but effectively, create leaf tags in plant images on a pixel-by-pixel basis. Barth et al. [7] rendered a composite dataset containing 10,500 images through Blender. The scene used had 42 program-generated plant models and random plant parameters. These parameters were based on 21 empirically measured plant characteristics at 115 locations on 15 plant stems. The fruit model was obtained through 3D scanning and the plant part textures were collected through photos as a reference dataset for modeling and evaluating the segmentation performance. David et al. [8] established a large, diverse, and well-labeled wheat image dataset, called the Global Wheat Head Detection (GWHD) dataset. It contained 4700 high-resolution RGB images from multiple countries and 190,000 wheat head markers at different growth stages, with a wide range of genotypes. Wang et al. [9] constructed a lettuce point cloud dataset consisting of 620 real and synthetic point clouds fused together for 3D instance segmentation network training. Lai et al. [10] first used the SfM-MVS method to obtain point clouds of these plant population scenes, which were then annotated similarly to the S3DIS dataset to obtain data that could be trained and tested. In order to provide important and available basic data support for the development of three-dimensional point cloud segmentation and phenotype automatic acquisition technology of soybeans, this study uses the multiple-view stereo technology to construct 102 soybean three-dimensional plant models by taking advantage of its low cost, fast speed and high precision. At the same time, it is manually labeled to construct the dataset for point cloud segmentation. Compared with other datasets, this dataset contains three-dimensional information on soybean plants during the whole growth period, which has certain advantages in model accuracy and quantity.

There are several key binocular stereovision spatial positioning technologies involving image acquisition, camera calibration, image preprocessing, edge feature extraction, and stereo matching. Multi-vision is based on binocular vision, adding one or more cameras as a measuring assistant so that multiple pairs of images from different angles of the same object can be obtained. For the 3D reconstruction of a single plant, this method is more suitable for low sunlight conditions in the laboratory (Duan et al. [11]; Hui et al. [12]). This method can also be used for 3D reconstruction in the field such as studying overall crop

canopy volumes (Biskup et al. [13]; Shafiekhani et al. [14]). Compared with other methods, the multiple-view stereo method requires relatively simple equipment, and the model can be established quickly and effectively, with minimum human-computer interaction required. Although the reconstruction speed is average and the requirements for the reconstruction of environmental factors are high, the reconstruction accuracy is high, it is easy to use, and the required equipment price is relatively low. Zhu et al. [15] built a soybean digital image acquisition platform based on the principle of constructing a multi-perspective stereovision system with digital cameras covering different angles, effectively improving the problem of mutual occlusion between soybean leaves. The morphological sequence images of target plants for 3D reconstruction were then obtained. Nguyen et al. [16] described a field 3D reconstruction system for plant phenotype acquisition. The system used synchronous, multi-view, high-resolution color digital images to create real 3D crop reconstructions and successfully obtained the plant canopy geometric characteristic parameters. Lu et al. [17] developed an MCP-based SfM system using multiple-view stereo technology and studied the appropriate 3D reconstruction method and the optimal shooting angle range. Choudhury et al. [18] devised the 3DPhenoMV method. Plant images captured from multiple side views were used as the algorithm input, and a 3D model of the plant was reconstructed using multiple side views and camera parameters. Miller et al. [19] used low-cost hand-held cameras and SfM-MVS to reconstruct a spatially accurate 3D model of a single tree. Shi et al. [20] adopted the multi-view method, allowing information from two-dimensional (2D) images to be integrated into the three-dimensional (3D) plant point cloud model, and evaluated the performance of 2D and multi-view methods on tomato seedlings. Lee et al. [21] proposed an image-based 3D plant reconstruction system based on multiple UAVs to simultaneously obtain two images from different views of plants during growth and reconstruct 3D crop models with moving structures, based on multiple view stereo algorithms and metric structures. Sunvittayakul et al. [22] developed a platform for acquiring 3D cassava root crown (CRC) models using close-range photogrammetry for phenotypic analysis. This novel method is low cost, and it is easy to set up the 3D acquisition requiring only a background sheet, a reference object, and a camera and is suitable for field experiments in remote areas. Wu et al. [23] developed a small branch phenotype analysis platform, MVS-Pheno V2, based on multi-view 3D reconstruction, which focused on low plant branches and realized high-throughput 3D data collection.

In this study, the multiple view stereo method (MVS) was used to reconstruct soybean plants. A soybean image acquisition platform was constructed to obtain multi-angle images of soybean plants at different growth stages. Based on the silhouette contour principle, the model was established by contour approximation, vertex analysis, and triangulation, and 3D point cloud and original soybean datasets were constructed. Meanwhile, the obtained 3D models of soybean were manually labeled using CloudCompare v2.6.3 software. An annotated 3D dataset called Soybean-MVS, including 102 models, was established. Due to the inherent changes in the appearance and shape of natural objects, the segmentation of plant parts was a challenge. In this paper, to verify the availability of this dataset, RandLA-Net and BAAF-Net point cloud semantic segmentation networks were used to train and test the Soybean-MVS dataset.

2. Materials and Methods

2.1. Method Process

In 2018 and 2019, we cultivated high-quality soybean plants including DN251, DN252, DN253, HN48, and HN51 varieties. An original 3D soybean dataset and labeled 3D soybean plant dataset were constructed for the whole soybean growth period, consisting of the first trifoliolate stage (V1), second trifoliolate stage (V2), third trifoliolate stage (V3), fourth trifoliolate stage (V4), fifth trifoliolate stage (V5), initial flowering stage (R1), full bloom stage (R2), initial pod stage (R3), full pod stage (R4) initial seed stage (R5), full seed stage (R6), initial maturity stage (R7), and full maturity stage (R8). Among them, V represents the vegetative growth stage and R represents the reproductive growth stage. Table 1

shows the basic characteristics of experimental soybean materials, including soybean varieties, growing days, planting methods, and active accumulated temperature greater than 10 °C. The research process of this paper mainly involved 3D reconstructions based on the multiple view stereo method, manually labeling data to build datasets, and training and evaluating datasets through point cloud segmentation. Figure 1 details the overall process of building a soybean 3D dataset for point cloud segmentation.

Table 1. Basic characteristics of soybean materials. This shows the basic attribute information of soybean materials selected for this experiment, including soybean varieties, childbearing days, accumulated temperature and planting methods.

Variety	Childbearing Days	>10 °C Accumulated Temperature	Planting Method
DN 251	125	2600 °C	potted planting
DN 252	124	2500 °C	potted planting
DN 253	115	2350 °C	potted planting
HN 48	118	2350 °C	potted planting
HN 51	126	2600 °C	potted planting

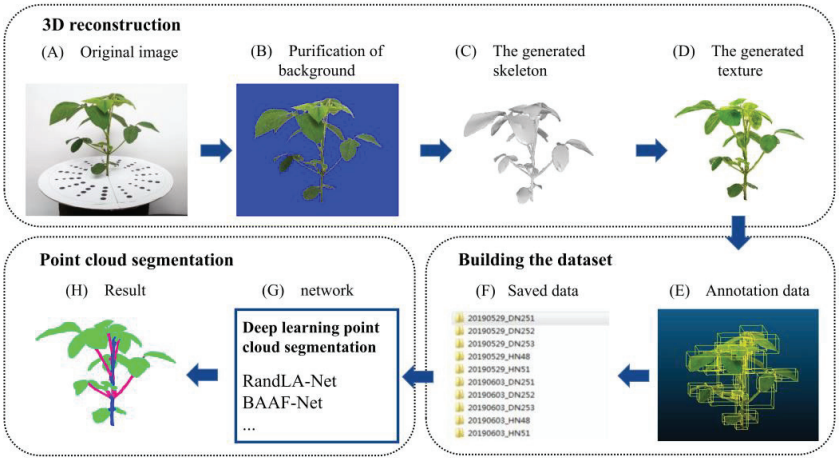


Figure 1. The process of building a soybean 3D dataset for point cloud segmentation. The process mainly includes three parts: 3D reconstruction, building the dataset, and point cloud segmentation. 3D reconstruction includes: (**A**) original image acquisition; (**B**) image preprocessing; (**C**) generation of 3D model skeleton; (**D**) generation of 3D model texture. Building the dataset includes: (**E**) data annotation; (**F**) construction of annotated dataset. Point cloud segmentation includes: (**G**) point cloud segmentation network selection; (**H**) result of point cloud segmentation.

2.2. Image Acquisition

This study prepared the image acquisition of 3D reconstruction in the room. The tools used to collect plant images included: (1) photo studio, (2) Canon EOS 600D SLR (Canon (China) Co. Ltd., Beijing, China) digital camera and camera rack, (3) rotary table, (4) calibration pad, and (5) white light absorbing cloth. A light source was added around the plant to guarantee the required basic environment needed for 3D reconstruction, based on the multiple view stereo method. The pot was about 90 cm from the camera. During the image acquisition for each pot of plants, we placed the plant pots on the rotary table, positioned a dot calibration pad at the plant roots, lowered the camera height, manually operated the rotary table, took a photo every 10°~25° (this study determined 24° according to the black dot on the calibration pad), and collected 15 photos after a circle of rotation. Then, according to the height of the plant, we adjusted the camera height three times on average, from low to high, and repeated the process. Finally, 60 photos were obtained by

taking four sets of circular rotation shots at different angles. According to the soybean growth, image acquisition was conducted at each growth stage (Figure 2). The final number of images of different varieties of soybean plants is shown in Appendix A Table A1.

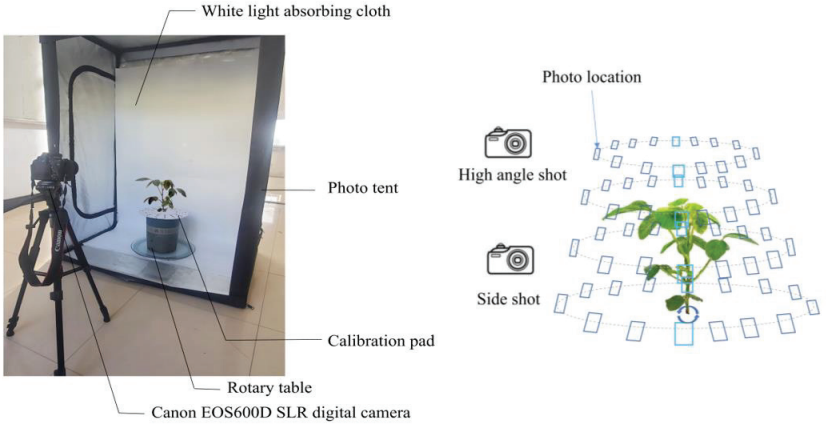


Figure 2. Soybean 3D reconstruction image acquisition. (**a**) Soybean image acquisition platform. (**b**) Schematic diagram of soybean plant 3D reconstruction image acquisition. The 3D reconstruction was carried out in a laboratory with no wind and sufficient light, using multiple-view stereo technology (MVS).

2.3. Three-Dimensional Reconstruction

This study obtained a large number of corresponding soybean plant images (about 60) from multiple perspectives. In addition, this study preprocessed basic image operations such as noise removal and distortion correction based on Python. At the same time, in the process of three-dimensional modeling, it is necessary to connect and combine images from different directions. Therefore, the relationship between the spatial positions of various images is particularly important. This study adopted the auxiliary camera calibration method of the calibration device, using a calibration pad to determine the problem of image overlap, and to determine the shooting direction of various multi-angle images. The model was established using the "contour extraction", "vertex calculation", and "visual shell generation" steps of the silhouette contour method. Silhouette contour is the contour line of the image projected on the imaging plane, which is an important clue to understanding the geometric shape of the object. When a space object is observed from multiple perspectives by perspective projection, a silhouette line of the object can be obtained in the corresponding screen of each perspective. Here the silhouette line and the corresponding perspective projection center together determine a cone of general shape in three-dimensional space, and the object to be observed is located inside this cone. By analogy, increasing the number of viewing angles of the target object from different directions can make the shape of each corresponding cone approach the surface of the object, so as to carry out three-dimensional visualization of the shape features of the target object.

Firstly, we masked the multi-angle images, selected the position of the soybean plants in each image, and purified all the background and calibration pad areas unrelated to the soybean plants, leaving only the complete soybean plant information. Then, according to the partial information of the target object in each multi-angle image, we obtained several approximate polygonal contours, numbered each approximate contour, calculated three vertices from the polygon contour, and recorded the information of each vertex. A triangular grid was used to divide the complete surface to outline the surface fine joints. The above is the realization of the "contour extraction" and "vertex calculation and visual shell generation" steps of the silhouette contour method. At that point, only the soybean

plant skeleton had been generated. In addition, further optimization operations such as volume optimization and surface refinement were required to obtain the final soybean plant surface morphology model. Finally, according to the corresponding orientation information characteristics of the three-dimensional surface contour soybean plant model obtained above, combined with the orientation information of different multi-angle images, texture mapping of its surface was performed, so that the model had more visual features and better described the characteristics of actual objects. Following three-dimensional reconstruction, 102 original models were obtained and named according to the year, date, and variety.

2.4. Data Annotation

The data annotation work in this study was completed using the open-source software CloudCompare v2.6.3. The acquired soybean 3D plant model (.obj format file) was imported into CloudCompare software, the leaves, main stems, and stems were manually segmented and marked on the soybean plants, and each point cloud was given a real label. At the same time, each segmented and marked organ was sampled points on a mesh. The number of sampling points was fixed at 50,000. The labeled point cloud information included xyzRGB information and was stored in .txt format. The soybean plant leaves, main stems, and stems were marked, as shown in Figure 3 (using 20180612_HN48 as an example). Finally, a labeled soybean 3D point cloud dataset named Soybean-MVS was constructed, including 102 3D models, of which 89 models were used as the training set and 13 models were used as the test set.

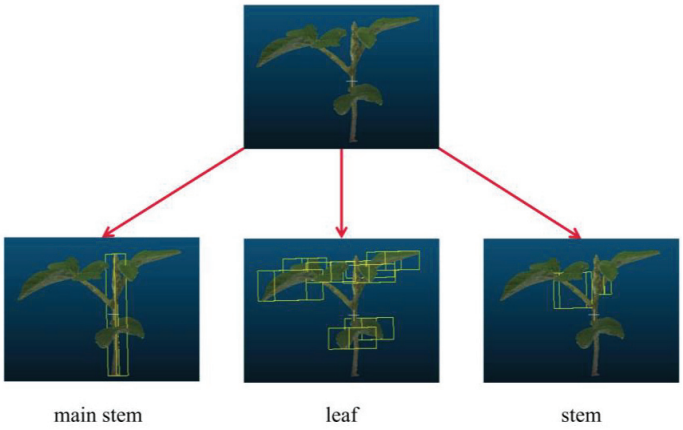


Figure 3. Manually mark leaves, main stems, and stems of soybean plants. The organs of the soybean plants were manually labeled.

2.5. Point Cloud Segmentation Network

For the semantic segmentation of the soybean-MVS 3D point cloud dataset, this study selected two deep learning-based point cloud segmentation network architectures, (1) RandLA-Net [24]; (2) BAAF-Net [25] to test its availability. Appendix A Table A2 shows the hardware, software, and super parameter configuration of the deep learning model. Figure 4 shows the architecture of the two-point cloud segmentation semantic models. We have already submitted the data and computer programs used for the analysis, which will allow the results of our experiments to be reproduced by anyone. The link addresses are https://github.com/18545155636/BAAF-Net.git (accessed on 1 January 2023) and https://github.com/18545155636/randla-net.git (accessed on 1 January 2023). The following briefly describes the key methods of these architectures for encoding 3D point cloud local geometry. Please refer to the original text for the default structure and other details of the architecture.

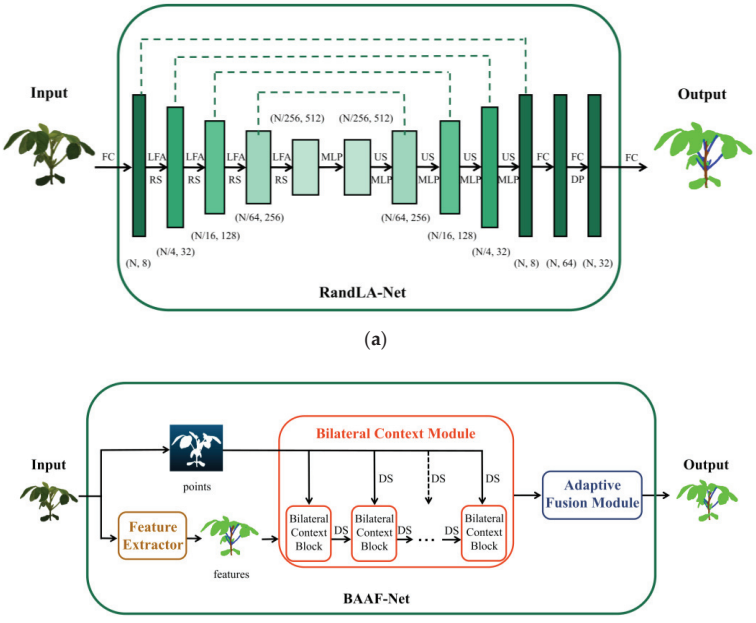


Figure 4. Point cloud semantic segmentation architecture. (**a**) RandLA-Net semantic segmentation architecture diagram. (**b**) BAAF-Net semantic segmentation architecture diagram. The dataset was trained and tested on two networks.

2.5.1. RandLA-Net

RandLA-Net is an effective and portable network that can identify the semantics of each point and apply it to large-scale point clouds. It uses the local feature aggregation module (LFA) to gradually improve the receptivity of each 3D point, which can effectively save the geometric details of the point cloud. The local feature aggregation module involves three main steps:

The first step is local spatial encoding (LocSE). The coordinates and features of a point (center point) in the point cloud P and K points adjacent to the point are taken as input. It consists of three parts: (1) Finding neighboring points, (2) relative point position encoding, and (3) point feature augmentation. A new adjacent feature of the center point is output, which encodes the local geometric feature of the center point. This module can significantly learn the local geometric features of point clouds, which will eventually play a beneficial role in learning the complex local structure information of the entire network. The second step is known as attention pooling. The LocSE output is used as the input of this step. This includes two parts: (1) computing attention scores and (2) weighted summation. Then, the feature vectors generated by the center point aggregated local features are output. The third step is called the divided residential block. It consists of multiple LocSE and attention pooling layers plus a skip connection.

RandLA-Net regards each point as the center point and each point aggregates the information of the surrounding points to itself. According to the principle that the points sampled in the whole point cloud by random sampling should conform to a normal distribution, random sampling is directly adopted. By employing this, the sampling speed can be greatly accelerated.

2.5.2. BAAF-Net

BAAF-Net uses a bilateral structure to increase the local context information of a point, while adaptively fusing multi-resolution features, to propose a new point cloud semantic segmentation network, involving the following two steps:

The first step is the bilateral context module. This consists of multiple bilateral context blocks (BCBs). A BCB is composed of bilateral augmentation and mixed local aggregation. During bilateral augmentation, the neighborhood information is aggregated around a point to the point to obtain the local context information in the geometric and feature spaces, but this is insufficient to express the domain information. Then, the local geometric context information is adjusted through the local semantic context information, which in turn is adjusted through the enhanced local geometric context information. Finally, MLP is used to further process the enhanced local geometric, and local semantic, context information and stack them together to obtain the enhanced local context information. The mixed local aggregation process uses the maximum pooling method, that is, the maximum K values of each feature are calculated as the value of the feature of point i. Then, the mean point of the local neighborhood of the point is learned through MLP, and the feature of the point is taken as the feature of point i. Lastly, the above two aggregated features are spliced to obtain the final feature of point i. The bilateral context module is used to combine bilateral context modules and continuously output the downsampled points to BCB, which is also the corresponding encoder part. The second step is the Adaptive Fusion Module. This part corresponds to the decoder. The encoder will output feature maps with different resolutions. The output of each layer is gradually upsampled to obtain full-size feature maps. The previous layer's feature maps need to be fused each time upsampling is performed. Then, the full-size feature maps sampled on these multiple scales need to be fused. To obtain different-sized important information, the full-size feature map is inputted into MLP to obtain the point level information, which is then normalized using Sofmax. Finally, the integrated feature map for semantic segmentation is obtained by fusing the normalized point level information and the full-size feature map after upsampling.

BAAF-Net enhances its local context by making full use of geometric and semantic features in bilateral structures. It fully explains the uniqueness of points from multiple resolutions and represents feature maps at the point level according to adaptive fusion methods for accurate semantic segmentation.

2.6. Evaluation Index

In this study, the average value of the *IoU* scores of three categories (*mIoU*) and the average accuracy (*mAcc*) were used to evaluate the success of each architecture. The number of true positives, true negatives, false positives, and false negatives in each category were expressed as *TP*, *TN*, *FP*, and *FN*, respectively. Then, the intersection over union (*IoU*) of each semantic class, the total accuracy (*Acc*) of each plant, the mean score of *IoU* (*mIoU*), and the mean accuracy (*mAcc*) were defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (1)$$

$$IoU = \frac{TP}{TP + FP + FN}, \quad (2)$$

$$mAcc = \frac{1}{n}\sum_{i=1}^{n} Acc, \quad (3)$$

$$mIoU = \frac{1}{k}\sum_{i=1}^{k} IoU, \quad (4)$$

where n represents the total number of datasets in the test set (13 data) and k represents the total number of categories.

3. Results

3.1. Soybean-MVS Dataset

3.1.1. Original 3D Dataset

This paper tracked and recorded the entire growth period of five varieties of soybean and created a 3D reconstruction of the soybean plants during each period. A total of 102 3D virtual soybean plants were obtained and a 3D point cloud dataset of original soybean plants was constructed. Appendix A Table A3 details the point cloud of the original soybean 3D plant dataset. Figure 5 shows the point cloud information map of the original soybean three-dimensional plant dataset. Figure 5a displays the comparison results of the total point cloud cover of stage V and stage R using a *t*-test. It can be seen that there was a significant difference between the point cloud covers of stage R and stage V, with the stage R point cloud cover being significantly larger than that of stage V. Figure 5b shows the comparison results of the reconstructed point cloud cover in 2018 and 2019 using a *t*-test. It can be seen that the reconstructed model had almost the same point cloud cover over two years. Figure 5c is the comparison map of the point cloud cover of soybean plants at different development stages following an ANOVA variance test, among which the point cloud cover of soybean plants at the R5 stage is the greatest, indicating that soybean plants grow the most vigorously during the R5 stage and reach the peak stage of their development. The two control graphs show that the more complex the soybean plant, the greater the model point cloud cover. Figure 5d is the comparison map of point cloud cover of different soybean varieties after an ANOVA variance test, and the difference in point cloud cover among different varieties is not found to be significant.

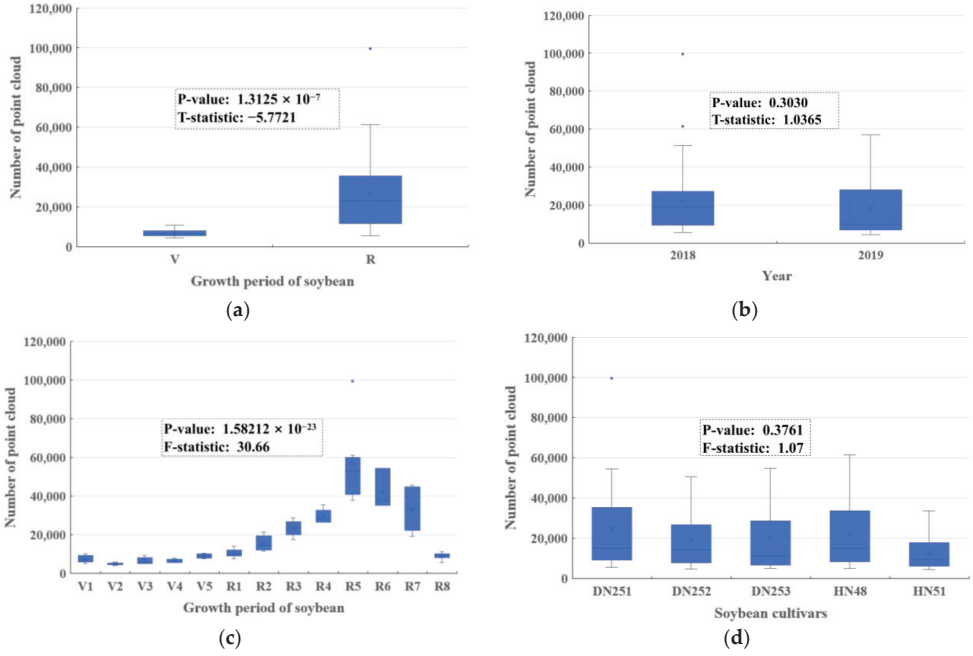


Figure 5. Point cloud information map of original soybean 3D plant dataset. (**a**) Comparison chart of total point cloud amount of stage V and stage R. (**b**) Comparison chart of reconstructed point cloud amount in 2018 and 2019. (**c**) Comparison chart of point cloud amount in different development stages. (**d**) Comparison chart of point cloud amounts of various varieties.

3.1.2. Labeled 3D Dataset

This study annotated the original dataset. In order to homogenize the point cloud, this study conducted network point collection for each labeled organ, and the number of sampled point clouds was controlled at 50,000. A labeled soybean 3D point cloud dataset was constructed. Figure 6 compares the point amount of the original 3D dataset and the sampled point cloud amount of the labeled 3D dataset, taking the DN252 soybean plant as an example. The leaves, main stems, and stems of three soybean plant organs were manually marked. Table 2 shows the number of organs of different types of soybean plants after labeling.

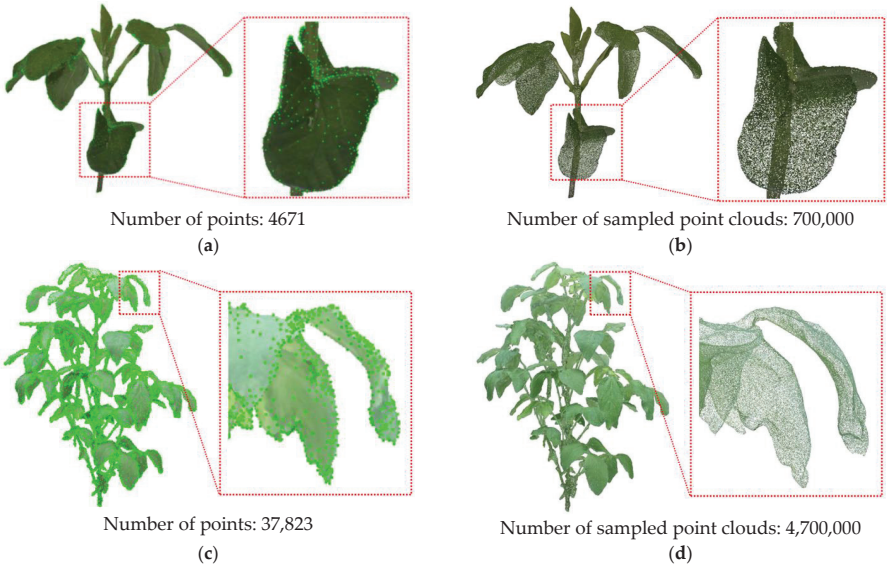


Figure 6. Comparison between the amount of points in the original 3D dataset and the amount of sampled point clouds in the labeled 3D dataset. (**a**,**c**) Point volume of the original dataset. (**b**,**d**) Sampled point cloud volume of labeled dataset.

Table 2. Number of organ markers in different soybean plants. The number of leaves, the number of main stems, and the number of stems were compared by counting the organs of labeled soybean plants.

	Leaf	Main Stem	Stem
DN251	756	22	182
DN252	813	22	188
DN253	718	20	165
HN48	649	21	161
HN51	437	17	125

Finally, 89 labeled models were divided into a training set, and 13 labeled models were divided into a test set. The point cloud amount distribution of each organ in the training set and test set is shown in Table 3.

Table 3. Point cloud amount distribution of each organ in the training set and test set (%). The proportion of cloud cover of different organ points in the training set and the test set was calculated.

	Leaf	Main Stem	Stem
Soybean-MVS training models	78.08	2.72	19.20
Soybean-MVS test models	79.13	2.36	18.51

3.2. Point Cloud Segmentation

The test results of 20 models in the Soybean-MVS dataset on the RandLA-Net and BAAF-Net models are shown in Table 4.

Table 4. Point cloud segmentation test results (%). The results of the dataset on two models, including IoU, mIoU, and mAcc.

		RandLA-Net	BAAF-Net
IoU	leaf	88.58	88.83
	main stem	57.03	27.25
	stem	45.54	48.23
mIoU		63.72	54.77
mAcc		88.52	87.45

Figure 7 shows the Acc of the same soybean plant (DN251) at different growth stages after RandLA-Net and BAAF-Net network tests. Overall, the mAcc tested by the two networks was high. For the different complex stages of soybean plant growth, the segmentation accuracy was high and there was no significant difference. Among them, the Acc value in the R5 period was the highest, which may be because the soybean plants are the most vigorous and the leaves are the most luxuriant during the R5 period. The effect of the two networks on the leaf segmentation was better than on the main stems and stems. At the R8 stage, because the soybean plant was leafless, the Acc value was lowest.

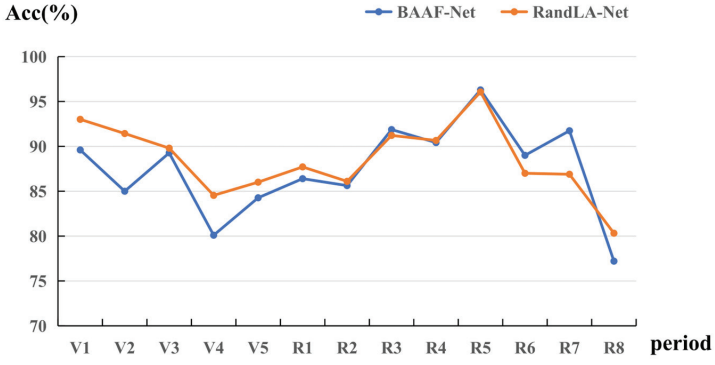


Figure 7. Acc results of soybean plants tested in the RandLA-Net network and BAAF-Net network during the whole growth period. This shows a comparison of the Acc results of the test set on the two models.

Figure 8 shows the label data, label data visualization results, RandLA-Net test visualization results, and BAAF-Net test visualization results of the DN251 soybean plants. From the results, both networks separated soybean plant leaves, main stems, and stems, but there were still identification errors in some details. Figure 9 highlights an example of a false prediction with a red ellipse. In terms of leaves, both networks performed well, which may be due to the regular leaf shape and a large amount of training, and they were all segmented. However, Figure 9a,b show that the two networks recognized stems as leaves when recognizing the petiole. In terms of the main stem, BAAF-Net performed worse than RandLA-Net. Figure 9c,d show that some main stem components were identified as stems. This may be due to the small amount of main stem training and the similar morphology of main stems and stems. In terms of the stem, Figure 9e,f show that both network test results identified the stems as part as leaves. In addition, Figure 9g,h show that RandLA-Net identified the connection between main stems and stems as a leaf, while the BAAF-Net performed well.

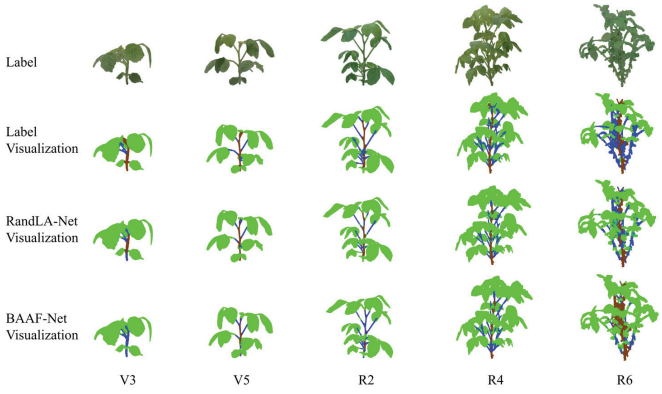


Figure 8. Soybean plant annotation data, RandLA Net, and BAAF Net visualization results in different stages. By contrast, this shows the overall segmentation effect of the two models.

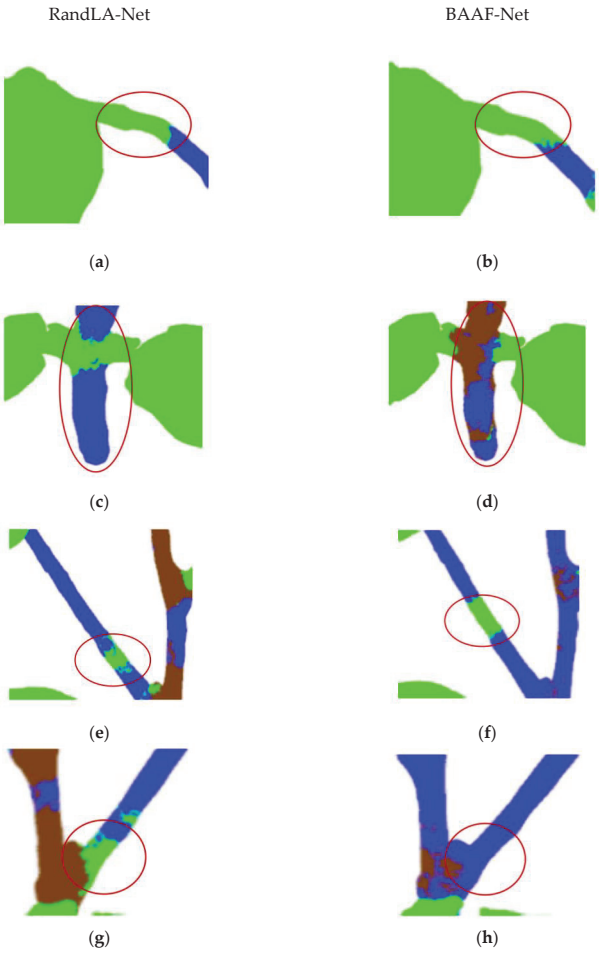


Figure 9. Example of error prediction. (**a**,**b**) Examples of false prediction of the petiole. (**c**,**d**) Examples of main stem error prediction. (**e**,**f**) Examples of stem error prediction. (**g**,**h**) Examples of error prediction at

the connection of main stem and stem. (**a**,**c**,**e**,**g**) The RandLA-Net test results. (**b**,**d**,**f**,**h**) BAAF-Net test results. By contrast, this shows the local segmentation difference between the two models.

4. Discussion

This paper explored the growth of soybean plants based on 3D reconstruction technology. Figure 10 shows the full soybean plant growth period, using the three-dimensional model of DN251 soybean plants constructed in this study as an example. The original three-dimensional soybean plant whole growth period dataset and the labeled three-dimensional plant soybean whole growth period dataset constructed in this study can provide an important basis for solving and tackling issues raised by breeders, producers, and consumers. For example, research on crop phenotypic measurement and other issues requires the effective phenotypic analysis of plant growth and morphological changes throughout the growth period. Considering this, we propose the use of point cloud segmentation.

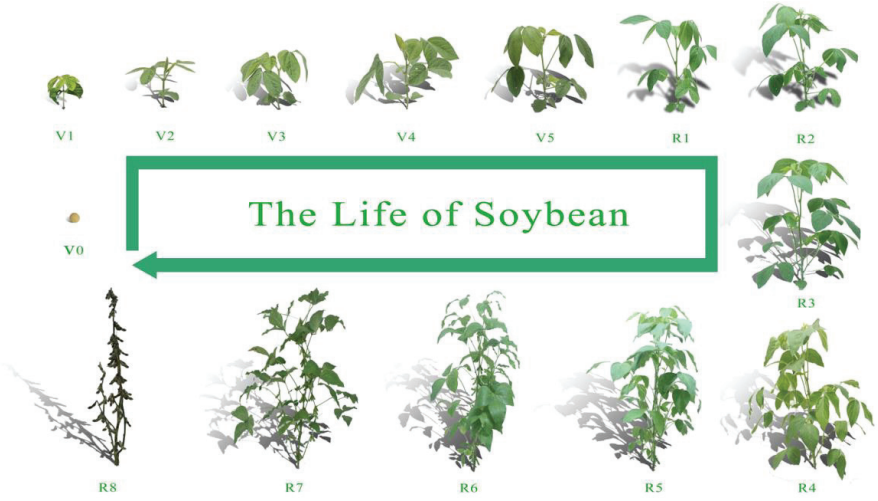


Figure 10. The life of soybean.

First of all, this paper chose the multiple-view stereo method to reconstruct the entire growth period of soybean plants. This method obtains detailed information about plants through crop images and extracts the phenotypic parameters of crops through related algorithms. Cao et al. [26] developed a 3D imaging acquisition system to collect plant images from different angles to reconstruct 3D plant models. However, only 20 images were collected in that study to meet the minimum image overlap requirements for 3D model reconstruction. In our study, 60 soybean plant images from different perspectives were collected at four different heights during image acquisition, so the 3D model obtained after 3D reconstruction was more accurate. At the same time, a three-dimensional dataset of the whole growth period of the original soybean was established. By comparing the original point cloud amount of the V and R stages, the relationship between the point cloud amount of the three-dimensional soybean plant model and the growth period was analyzed, which confirmed that the number of plant point clouds was consistent with corresponding real plant development. This provides an important basis for more accurate three-dimensional reconstruction of crops in the whole growth period in the future.

Secondly, training point cloud segmentation models usually require a large amount of tag data, the cost of which is very high, particularly in intensive prediction tasks such as semantic segmentation. In addition, the plant phenotype dataset also faces the additional challenges of severe occlusion and different lighting conditions, which makes obtaining annotations more time-consuming (Rawat et al. [27]). Gong et al. [28] used a structured light

3D scanning platform, based on a special turntable, to obtain the 3D point cloud data of rice panicles, and then used the open-source software LabelMe to mark point by point and create a rice panicle point cloud dataset. Boogaard et al. [29] manually marked cucumber plants twice with CloudCompare and constructed annotated dataset A and annotated dataset B. Dutagaci et al. [30] obtained 11 3D point cloud models of Rosa through X-ray tomography and manually annotated them, creating a labeled dataset to evaluate 3D plant organ segmentation methods, called the ROSE-X dataset. However, these datasets do not emphasize the importance of three-dimensional data of the entire growth period of plants and the amount of data is relatively small, which lacks integrity for subsequent studies such as the phenotypic measurement of whole plant growth periods. In our study, Soybean-MVS, a labeled three-dimensional dataset of the whole growth period of soybean, was constructed, which fully meets the data volume requirements of in-depth learning point cloud segmentation training and evaluation and ensures the integrity of the dataset used for the point cloud segmentation research. This not only provides a basis for measuring plant phenotype, bionic species, and other issues, but may also provide a basis for exploring the natural laws of plant growth.

Thirdly, in the process of labeling the dataset in our paper, since the soybean plant main stem and stem information are relatively similar, and a soybean plant only has one main stem, the number is much lower than leaf and stem, leading to a low segmentation accuracy of the main stems. There is a situation where the points on the petiole were classified as leaves. However, the visualization results show that each point cloud segmentation network model still segmented most of the points on the main stems. Therefore, the Soybean-MVS dataset can ensure the effectiveness of the point cloud segmentation task.

Finally, the Soybean-MVS dataset is universal. The universality of datasets is crucial to empirical research evaluation for at least three reasons: (1) providing a basis for measuring progress by copying and comparing results; (2) revealing the shortcomings of the latest technology, thus paving the way for novel methods and research directions; (3) the method can be developed without first collecting and tagging data (Schunck et al. [31]). Furthermore, data with high universality can meet the requirements of different point cloud segmentation models and obtain a highly reliable segmentation model. Turgut et al. [32] evaluated their performance on real rose shrubs based on the ROSE-X and synthetic model datasets and adjusted six-point cloud-based deep learning architectures (PointNet, etc.) to subdivide the structure of a rosebush model. In our paper, RandLA-Net and BAAF-Net were used for testing (also applicable to other 3D point cloud classification and segmentation models based on depth learning). In the future, we will continue to expand and adjust the Soybean-MVS dataset and apply it to other point cloud segmentation network models, to further improve it.

5. Conclusions

In order to provide important and usable basic data support for the development of three-dimensional point cloud segmentation and phenotype automatic acquisition technology of soybeans, this paper adopted the multiple-view stereo technology and obtained 60 photos in each group through four different height circular rotation shots. Three-dimensional plant reconstruction was carried out using the profile contour method to construct the original three-dimensional soybean plant dataset of the whole growth period. It was concluded that the number of point clouds was consistent with the actual plant development. The leaf, mainstem and stem in the obtained data and sample points were manually annotated on a mesh. A soybean three-dimensional plant dataset named Soybean-MVS was constructed for point cloud semantic segmentation. Finally, RandLA-Net and BAAF-Net models were used to evaluate the dataset, and the mAcc of the test results were 88.52% and 87.45%, respectively. The usability of the Soybean-MVS labeled 3D plant dataset was verified. The publication of this dataset provides an important basis for proposing an updated, high-precision, and efficient 3D crop model segmentation algorithm. In the future, we will constantly update and supplement the dataset, and apply it to more point cloud

segmentation models to make it more universal. At the same time, the automatic acquisition and breeding of soybean phenotype will be further explored on the basis of this dataset.

Author Contributions: Y.S.: formal analysis, investigation, methodology, image acquisition, three-dimensional reconstruction, annotation of data and writing—original draft. Z.Z. (Zhixin Zhang): supervision and validation. K.S. and J.Y.: image acquisition and three-dimensional reconstruction. S.L. and L.M.: annotation of data. Y.L., Z.H., Z.Z. (Zhanguo Zhang) and H.Z.: project administration and resources. D.X.: writing—review and editing and funding acquisition. Q.C.: writing—review and editing, funding acquisition, and resources. R.Z.: designed the research of the article, conceptualization, data curation, funding acquisition, resources, and writing—review and editing. All authors agreed to be accountable for all aspects of their work to ensure that the questions related to the accuracy or integrity of any part is appropriately investigated and resolved. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of Heilongjiang Province of China (LH2021C021).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Original models are available in a publicly accessible repository: The original contributions presented in the study are publicly available. These data can be found here: https://www.kaggle.com/datasets/soberguo/soybean-original-model (accessed on 1 January 2023). The soybean-MVS dataset is available in a publicly accessible repository: Publicly available datasets were analyzed in this study. These data can be found here: https://www.kaggle.com/datasets/soberguo/soybeanmvs (accessed on 1 January 2023).

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

Appendix A

Table A1. Image collection quantity of soybean plants of different varieties in different stages.

	V1		V2		V3		V4		V5		R1		R2		R3		R4		R5		R6		R7		R8	
	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019	2018	2019
DN251	0	60	0	60	60	60	60	60	0	60	60	60	60	60	60	60	0	60	60	60	60	60	60	60	60	60
DN252	0	60	0	60	60	60	60	60	0	60	60	60	60	60	60	60	0	60	60	60	60	60	60	60	60	60
DN253	0	60	0	60	60	60	60	60	0	60	60	60	60	60	60	60	0	60	60	60	60	60	60	60	60	60
HN48	0	60	0	60	60	60	60	60	0	60	60	60	60	60	60	60	0	60	60	60	60	60	60	60	60	60
HN51	0	60	0	60	60	60	60	60	0	60	60	60	60	60	60	60	0	60	60	60	60	60	60	60	60	60

Notes: In this study, five kinds of soybeans, DN251, DN252, DN253, HN48 and HN51, were planted in the pot farm of Northeast Agricultural University in 2018 and 2019, and images were collected during the whole growth period of soybeans. Table 1 shows the specific number of images collected.

Table A2. Hardware, software, and hyperparameter configuration of deep learning models.

Catalogue	Content
CPU	Core i9-12900kf
RAM	64 GB
GPU	NVIDIA 3090 (24 GB)
operating system	Ubuntu 18.04
Cuda	11.3
Cudnn	8.4
Data Annotation	CloudCompare
Deep learning framework	Tensorflow 2.6.0
Anaconda	Anaconda 5.2
Momentum	0.9
threshold	0.5

Table A3. Original information of 3D soybean plant model.

Variety	Date of Reconstruction	Stage	Points
DN251	12 June 2018	V3	66,528
DN252	12 June 2018	V3	85,871
DN253	12 June 2018	V3	5164
HN48	12 June 2018	V3	63,915
HN51	12 June 2018	V3	5390
DN251	19 June 2018	V4	78,211
DN252	19 June 2018	V4	7482
DN253	19 June 2018	V4	6581
HN48	19 June 2018	V4	5776
HN51	19 June 2018	V4	6734
DN251	26 June 2018	R1	10,752
DN252	26 June 2018	R1	140,986
DN253	26 June 2018	R1	11,535
HN48	26 June 2018	R1	9371
DN251	4 July 2018	R2	14,842
DN252	4 July 2018	R2	21,367
HN48	4 July 2018	R2	18,757
HN51	4 July 2018	R2	12,300
DN251	11 July 2018	R3	25,306
DN252	11 July 2018	R3	24,316
DN253	11 July 2018	R3	26,733
HN48	11 July 2018	R3	22,995
HN51	11 July 2018	R3	271,221
DN251	26 July 2018	R5	99,451
DN252	26 July 2018	R5	37,704
DN253	26 July 2018	R5	51,456
HN48	26 July 2018	R5	61,301
HN51	26 July 2018	R5	808,638
DN251	17 August 2018	R6	35,193
DN252	17 August 2018	R6	37,896
DN251	8 September 2018	R7	24,864
DN252	8 September 2018	R7	19,805
DN253	8 September 2018	R7	19,145
HN48	8 September 2018	R7	35,983
HN51	8 September 2018	R7	33,647
DN251	3 October 2018	R8	5574
DN252	3 October 2018	R8	8662
DN253	3 October 2018	R8	11,313
HN48	3 October 2018	R8	11,220
HN51	3 October 2018	R8	9366
DN251	29 May 2019	V1	9415
DN252	29 May 2019	V1	10,233
DN253	29 May 2019	V1	7014
HN48	29 May 2019	V1	8766
HN51	29 May 2019	V1	6541
DN251	3 June 2019	V2	6113
DN252	3 June 2019	V2	4671
DN253	3 June 2019	V2	4860
HN48	3 June 2019	V2	4947
HN51	3 June 2019	V2	4269
DN251	8 June 2019	V3	8322
DN252	8 June 2019	V3	5228
DN253	8 June 2019	V3	5161
HN48	8 June 2019	V3	7974
HN51	8 June 2019	V3	5777
DN251	12 June 2019	V4	7890
DN252	12 June 2019	V4	5612
DN253	12 June 2019	V4	88,756

Table A3. *Cont.*

Variety	Date of Reconstruction	Stage	Points
HN48	12 June 2019	V4	113,444
HN51	12 June 2019	V4	5956
DN251	18 June 2019	V5	9132
DN252	18 June 2019	V5	7669
DN253	18 June 2019	V5	9416
HN48	18 June 2019	V5	10,604
HN51	18 June 2019	V5	100,902
DN251	24 June 2019	R1	149,372
DN252	24 June 2019	R1	9728
DN253	24 June 2019	R1	135,007
HN48	24 June 2019	R1	160,789
HN51	24 June 2019	R1	7672
DN251	27 June 2019	R2	13,951
DN252	27 June 2019	R2	171,706
DN253	27 June 2019	R2	176,975
HN48	27 June 2019	R2	242,936
HN51	27 June 2019	R2	11,597
DN251	5 July 2019	R3	19,569
DN252	5 July 2019	R3	20,336
DN253	5 July 2019	R3	286,872
HN48	5 July 2019	R3	22,544
HN51	5 July 2019	R3	17,661
DN251	13 July 2019	R4	29,729
DN252	13 July 2019	R4	26,609
DN253	13 July 2019	R4	28,611
HN48	13 July 2019	R4	35,583
HN51	13 July 2019	R4	26,426
DN251	22 July 2019	R5	37,823
DN252	22 July 2019	R5	50,636
DN253	22 July 2019	R5	54,806
HN48	22 July 2019	R5	56,830
DN251	6 August 2019	R6	54,325
DN252	6 August 2019	R6	712,682
DN253	6 August 2019	R6	632,552
HN48	6 August 2019	R6	603,497
DN251	26 August 2019	R7	45,556
DN252	26 August 2019	R7	45,332
DN253	26 August 2019	R7	44,100
HN48	26 August 2019	R7	27,986
DN251	21 September 2019	R8	9990
DN252	21 September 2019	R8	8426
DN253	21 September 2019	R8	9317
HN48	21 September 2019	R8	7229
HN51	21 September 2019	R8	9964

References

1. Li, D.; Shi, G.; Li, J.; Chen, Y.; Zhang, S.; Xiang, S.; Jin, S. PlantNet: A dual-function point cloud segmentation network for multiple plant species. *ISPRS J. Photogramm. Remote Sens.* **2022**, *184*, 243–263. [CrossRef]
2. Zhou, J.; Fu, X.; Zhou, S.; Zhou, J.; Ye, H.; Nguyen, H.T. Automated segmentation of soybean plants from 3D point cloud using machine learning. *Comput. Electron. Agric.* **2019**, *162*, 143–153. [CrossRef]
3. Li, Y.; Wen, W.; Miao, T.; Wu, S.; Yu, Z.; Wang, X.; Guo, X.; Zhao, C. Automatic organ-level point cloud segmentation of maize shoots by integrating high-throughput data acquisition and deep learning. *Comput. Electron. Agric.* **2022**, *193*, 106702. [CrossRef]
4. Conn, A.; Pedmale, U.V.; Chory, J.; Navlakha, S. High-Resolution Laser Scanning Reveals Plant Architectures that Reflect Universal Network Design Principles. *Cell Syst.* **2017**, *5*, 53–62.e3. [CrossRef] [PubMed]
5. Li, D.; Li, J.; Xiang, S.; Pan, A. PSegNet: Simultaneous Semantic and Instance Segmentation for Point Clouds of Plants. *Plant Phenomics* **2022**, *2022*, 9787643. [CrossRef]

6. Uchiyama, H.; Sakurai, S.; Mishima, M.; Arita, D.; Okayasu, T.; Shimada, A.; Taniguchi, R.I. An easy-to-setup 3D phenotyping platform for KOMATSUNA dataset. In Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 2038–2045.
7. Barth, R.; Ijsselmuiden, J.; Hemming, J.; Henten, E.J.V. Data synthesis methods for semantic segmentation in agriculture: A Capsicum annuum dataset. *Comput. Electron. Agric.* **2018**, *144*, 284–296. [CrossRef]
8. David, E.; Madec, S.; Sadeghi-Tehran, P.; Aasen, H.; Zheng, B.; Liu, S.; Kirchgessner, N.; Ishikawa, G.; Nagasawa, K.; Badhon, M.A.; et al. Global Wheat Head Detection (GWHD) Dataset: A Large and Diverse Dataset of High-Resolution RGB-Labelled Images to Develop and Benchmark Wheat Head Detection Methods. *Plant Phenomics* **2020**, *2020*, 3521852. [CrossRef]
9. Wang, L.; Zheng, L.; Wang, M. 3D Point Cloud Instance Segmentation of Lettuce Based on PartNet. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Period, New Orleans, LA, USA, 19–23 June 2022; pp. 1647–1655.
10. Lai, Y.; Lu, S.; Qian, T.; Chen, M.; Zhen, S.; Guo, L. Segmentation of Plant Point Cloud based on Deep Learning Method. *Comput. Aided Des. Appl.* **2022**, *19*, 1117–1129. [CrossRef]
11. Duan, T.; Chapman, S.C.; Holland, E.; Rebetzke, G.J.; Guo, Y.; Zheng, B. Dynamic quantification of canopy structure to characterize early plant vigour in wheat genotypes. *J. Exp. Bot.* **2016**, *67*, 4523–4534. [CrossRef]
12. Hui, F.; Zhu, J.; Hu, P.; Meng, L.; Zhu, B.; Guo, Y.; Li, B.; Ma, Y. Image-based dynamic quantification and high-accuracy 3D evaluation of canopy structure of plant populations. *Ann. Bot.* **2018**, *121*, 1079–1088. [CrossRef]
13. Biskup, B.; Scharr, H.; Schurr, U.; Rascher, U. A stereo imaging system for measuring structural parameters of plant canopies. *Plant Cell Environ.* **2007**, *30*, 1299–1308. [CrossRef]
14. Shafiekhani, A.; Kadam, S.; Fritschi, F.B.; Desouza, G.N. Vinobot and Vinoculer: Two Robotic Platforms for High-Throughput Field Phenotyping. *Sensors* **2017**, *17*, 214. [CrossRef]
15. Zhu, R.; Sun, K.; Yan, Z.; Yan, X.; Yu, J.; Shi, J.; Hu, Z.; Jiang, H.; Xin, D.; Zhang, Z.; et al. Analysing the phenotype development of soybean plants using low-cost 3D reconstruction. *Sci. Rep.* **2020**, *10*, 7055. [CrossRef]
16. Nguyen, T.T.; Slaughter, D.C.; Townsley, B.; Carriedo, L.; Sinha, N. Comparison of Structure-from-Motion and Stereo Vision Techniques for Full In-Field 3D Reconstruction and Phenotyping of Plants: An Investigation in Sunflower. In Proceedings of the Asabe International Meeting, Orlando, FL, USA, 17–20 July 2016.
17. Lu, X.; Ono, E.; Lu, S.; Zhang, Y.; Teng, P.; Aono, M.; Shimizu, Y.; Hosoi, F.; Omasa, K. Reconstruction method and optimum range of camera-shooting angle for 3D plant modeling using a multi-camera photography system. *Plant Methods* **2020**, *16*, 118. [CrossRef]
18. Das Choudhury, S.; Maturu, S.; Samal, A.; Stoerger, V.; Awada, T. Leveraging Image Analysis to Compute 3D Plant Phenotypes Based on Voxel-Grid Plant Reconstruction. *Front Plant Sci.* **2020**, *11*, 521431. [CrossRef]
19. Miller, J.; Morgenroth, J.; Gomez, C. 3D modelling of individual trees using a handheld camera: Accuracy of height, diameter and volume estimates. *Urban For. Urban Green.* **2015**, *14*, 932–940. [CrossRef]
20. Shi, W.; Van De Zedde, R.; Jiang, H.; Kootstra, G. Plant-part segmentation using deep learning and multi-view vision. *Biosyst. Eng.* **2019**, *187*, 81–95. [CrossRef]
21. Lee, H.-S.; Thomasson, J.A.; Han, X. Improvement of field phenotyping from synchronized multi-camera image collection based on multiple UAVs collaborative operation systems. In Proceedings of the 2022 ASABE Annual International Meeting, Houston, TX, USA, 17–20 July 2022.
22. Sunvittayakul, P.; Kittipadakul, P.; Wonnapinij, P.; Chanchay, P.; Wannitikul, P.; Sathitnaitham, S.; Phanthanong, P.; Changwitchukarn, K.; Suttangkakul, A.; Ceballos, H.; et al. Cassava root crown phenotyping using three-dimension (3D) multi-view stereo reconstruction. *Sci. Rep.* **2022**, *12*, 10030. [CrossRef]
23. Wu, S.; Wen, W.; Gou, W.; Lu, X.; Zhang, W.; Zheng, C.; Xiang, Z.; Chen, L.; Guo, X. A miniaturized phenotyping platform for individual plants using multi-view stereo 3D reconstruction. *Front. Plant Sci.* **2022**, *13*, 897746. [CrossRef]
24. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Randla-net: Efficient semantic segmentation of large-scale point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11108–11117.
25. Qiu, S.; Anwar, S.; Barnes, N. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 1757–1767.
26. Cao, W.; Zhou, J.; Yuan, Y.; Ye, H.; Nguyen, H.T.; Chen, J.; Zhou, J. Quantifying Variation in Soybean Due to Flood Using a Low-Cost 3D Imaging System. *Sensors* **2019**, *19*, 2682. [CrossRef]
27. Rawat, S.; Chandra, A.L.; Desai, S.V.; Balasubramanian, V.N.; Ninomiya, S.; Guo, W. How Useful Is Image-Based Active Learning for Plant Organ Segmentation? *Plant Phenomics* **2022**, *2022*, 9795275. [CrossRef] [PubMed]
28. Gong, L.; Du, X.; Zhu, K.; Lin, K.; Lou, Q.; Yuan, Z.; Huang, G.; Liu, C. Panicle-3D: Efficient Phenotyping Tool for Precise Semantic Segmentation of Rice Panicle Point Cloud. *Plant Phenomics* **2021**, *2021*, 9838929. [CrossRef] [PubMed]
29. Boogaard, F.P.; Van Henten, E.J.; Kootstra, G. Boosting plant-part segmentation of cucumber plants by enriching incomplete 3D point clouds with spectral data. *Biosyst. Eng.* **2021**, *211*, 167–182. [CrossRef]
30. Dutagaci, H.; Rasti, P.; Galopin, G.; Rousseau, D. ROSE-X: An annotated dataset for evaluation of 3D plant organ segmentation methods. *Plant Methods* **2020**, *16*, 28. [CrossRef]

31. Schunck, D.; Magistri, F.; Rosu, R.A.; Cornelissen, A.; Chebrolu, N.; Paulus, S.; Leon, J.; Behnke, S.; Stachniss, C.; Kuhlmann, H.; et al. Pheno4D: A spatio-temporal dataset of maize and tomato plant point clouds for phenotyping and advanced plant analysis. *PLoS ONE* **2021**, *16*, e0256340. [CrossRef]
32. Turgut, K.; Dutagaci, H.; Galopin, G.; Rousseau, D. Segmentation of structural parts of rosebush plants with 3D point-based deep learning methods. *Plant Methods* **2022**, *18*, 20. [CrossRef]

agriculture

MDPI

Article

Method for Classifying Apple Leaf Diseases Based on Dual Attention and Multi-Scale Feature Extraction

Jie Ding [1,2], Cheng Zhang [1,2], Xi Cheng [3], Yi Yue [1,2], Guohua Fan [1,2], Yunzhi Wu [1,2,*] and Youhua Zhang [1,2]

[1] Anhui Provincial Engineering Laboratory for Beidou Precision Agriculture Information, Hefei 230036, China
[2] School of Information and Computer, Anhui Agricultural University, Hefei 230036, China
[3] School of Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
* Correspondence: wuyzh@ahau.edu.cn

Abstract: Image datasets acquired from orchards are commonly characterized by intricate backgrounds and an imbalanced distribution of disease categories, resulting in suboptimal recognition outcomes when attempting to identify apple leaf diseases. In this regard, we propose a novel apple leaf disease recognition model, named RFCA ResNet, equipped with a dual attention mechanism and multi-scale feature extraction capacity, to more effectively tackle these issues. The dual attention mechanism incorporated into RFCA ResNet is a potent tool for mitigating the detrimental effects of complex backdrops on recognition outcomes. Additionally, by utilizing the class balance technique in conjunction with focal loss, the adverse effects of an unbalanced dataset on classification accuracy can be effectively minimized. The RFB module enables us to expand the receptive field and achieve multi-scale feature extraction, both of which are critical for the superior performance of RFCA ResNet. Experimental results demonstrate that RFCA ResNet significantly outperforms the standard CNN network model, exhibiting marked improvements of 89.61%, 56.66%, 72.76%, and 58.77% in terms of accuracy rate, precision rate, recall rate, and F1 score, respectively. It is better than other approaches, performs well in generalization, and has some theoretical relevance and practical value.

Keywords: dual attention mechanism; multi-scale feature extraction; RFCA ResNet; classification

Citation: Ding, J.; Zhang, C.; Cheng, X.; Yue, Y.; Fan, G.; Wu, Y.; Zhang, Y. Method for Classifying Apple Leaf Diseases Based on Dual Attention and Multi-Scale Feature Extraction. *Agriculture* **2023**, *13*, 940. https://doi.org/10.3390/agriculture13050940

Academic Editors: Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 6 March 2023
Revised: 21 April 2023
Accepted: 22 April 2023
Published: 25 April 2023

1. Introduction

China is the world's leading apple grower and occupies a significant position in the global apple market [1]. However, apple production is vulnerable to climate, pests, and diseases, which can cause negative impacts on both the quantity and quality of the fruit, as well as substantial financial losses [2]. In the early stages of apple disease, most affected areas appear on the leaves, and visual observation is the primary method used to identify these diseases. However, identifying the specific type of disease is challenging, and misdiagnosis is common. Therefore, it is crucial to swiftly and accurately recognize the various types and complexities of apple diseases.

In agriculture, computer vision has been widely utilized [3–8], particularly in the field of plant disease detection [9]. This technology is a critical factor in productive agriculture and economic growth. With advancements in machine learning, image processing techniques can now be used to solve problems using morphological features such as color, intensity, and size. Zhang Chuanlei et al. [10] initially employed image processing to convert the color space of images, conduct background removal, and employ the region-growing algorithm to segregate lesions. The evolutionary algorithm and correlation feature selection method were then utilized to screen essential features, to improve the model's accuracy. Finally, the support vector machine (SVM) was used for automatic identification, and the method accurately identified apple mosaic, rust, and other diseases with an accuracy rate of over 90%. Nuruzzaman et al. [11] compared the results of machine learning algorithms such as the random forest classifier, support vector machine, and

logistic regression on 1200 potato images. Ultimately, the logistic regression algorithm produced the best result. Similarly, Chakraborty et al. [12] employed the Otsu threshold technique and histogram equalization to segregate diseased apple leaf sections, and then utilized a multiclass SVM to detect these sections, with an accuracy rate of 96%.

The application of machine learning technologies in practical agricultural settings has been challenging due to various constraints, such as the requirement for high-precision image acquisition equipment, homogeneous illumination, and simple image backgrounds. Recently, convolutional neural networks (CNNs) have emerged as a promising technique for directly learning important features from data, with good performance on large datasets and high adaptability. Consequently, CNNs have increasingly been applied to plant disease recognition and identification with impressive results [13–18].

To overcome the issue of overfitting, Jiang Peng et al. [19] constructed datasets for five common leaf diseases, including apple brown spots, by enhancing and annotating the data. They utilized the VGG [20] network as the basic framework and introduced the Inception module to extract multi-scale lesions, along with the feature pyramid's context and fusion features to enhance recognition performance. The model obtained a recognition accuracy of 78.8% mAP. Similarly, Liu Aoyu et al. [21] addressed the inadequacies of manual diagnosis of corn diseases by constructing and training the ResNet50 network on the PlantVillage dataset. They added data augmentation operations to the collected corn dataset and incorporated the focal loss function to handle difficult-to-classify samples, resulting in an average accuracy of 98.60%. Thapa Ranjita et al. [22] introduced the Plant Pathology 2021 Challenge dataset, which comprised images captured from various distances, angles, and lighting conditions, to represent real-world scenarios of disease symptoms on cultivated apple leaves. The dataset featured a complex background and an uneven distribution of categories. The authors performed a multiclass classification task using ResNet34, and the experimental results revealed that the performance was poor for the combination of diseases such as apple scab and frog eye leaf spot, while the combination of snow apple rust and gray spot, as well as the combination of snow apple rust and other diseases, exhibited high accuracy. The corresponding rate scores were all above 0.75. Yan Qian et al. [23] replaced the fully connected layer with the batch norm layer and the global average pooling layer, and pre-trained the VGG16 network to recognize three apple leaf diseases: scab, frost spot, and cedar rust. The model's overall accuracy was 99.01%. Sardogan et al. [24] employed Inceptionv2 to differentiate between healthy and diseased apple leaves in images with complex backgrounds. They first used the Faster R-CNN method to locate and mark various items and regions on the image and then achieved a typical accuracy rate of 84.5%. Finally, Li Xiaopeng et al. [25] combined convolution and transformer to extract both global and local disease features. They utilized the self-attention mechanism and visual transformer to direct the convolutional network to focus on effective features and applied separable convolution and global average pooling operations to reduce model complexity. Their approach achieved equivalent identification accuracy to the Swin Tiny [26] model, while being lighter in weight.

In real-world scenarios, the datasets collected for plant disease classification are often imbalanced due to a low incidence rate of a specific disease or the presence of multiple diseases simultaneously. However, using the conventional approach of classifying plant diseases as mainstream, through a convolutional neural network and cross-entropy loss function, does not yield satisfactory results on such datasets. In this research, we aim to enhance the detection ability of convolutional neural networks on an unbalanced plant disease dataset with complex backgrounds. Our primary contributions are:

- Extraction of multi-scale lesion features based on the RFB module and adjusting the convolution kernel size to improve recognition accuracy.
- Construction of the RFCA ResNet network, which utilizes ResNet18 as the backbone network, using focal loss in combination with the class balance approach to enhance the detection performance on the imbalanced dataset.

- Building a dual attention mechanism that incorporates both the coordinate attention mechanism and the frequency attention mechanism to improve lesion feature extraction capabilities.
- Comparison and evaluation of our proposed approach with the conventional cross-entropy loss function-based classification method, which has theoretical importance and practical relevance in real-world applications.

The remainder of this research paper is structured as follows. Section 2 provides a detailed description of the network structure and loss function. In Section 3, we introduce the dataset source, preprocessing method, experimental apparatus, experimental design, and evaluation indexes. The experimental results are presented and analyzed in Section 4. In Section 5, we discuss and evaluate our work. Finally, we conclude the research in Section 6 and provide directions for future work.

2. Methods

2.1. RFCA ResNet Design

The apple leaf disease dataset used in this research has a complex visual background, which was collected under different lighting conditions and at different times. Due to the dispersed and varying sizes of the disease spots and the uneven number of photos in each category, model identification is challenging. Therefore, the aim of this research is to design a model, with relatively low computational complexity, that can accurately classify datasets with an uneven number of categories. To achieve this, we designed a convolutional neural network model based on a dual attention mechanism, utilizing the ResNet topology model. To limit computation and network complexity, we chose an 18-layer ResNet as the fundamental network. As using a single-sized convolution kernel may result in the loss of extracted feature information, we replaced the first convolutional layer in the ResNet with the RFB module, which can improve the recognition of lesions of various sizes on leaves by adjusting the receptive field's size using parallel expansion convolution kernels of various sizes. The attention mechanism helps the model focus on relevant information while ignoring irrelevant information. Therefore, to enhance the ability to retrieve lesion features, we included the intended attention module in each residual structure. The precise structure of the RFCA ResNet model is shown in Figure 1. The model mainly comprises the FCCA attention mechanism module and the enhanced ResNet18, designed to accurately classify complex datasets with an uneven number of categories while having relatively low computational complexity.

In the task of identifying plant diseases, some categories of images may have a very low probability of occurrence, or there may be multiple diseases coexisting on the leaves, resulting in certain categories having a significantly higher number of images than others. This can lead to overfitting of the network during training, where the model becomes biased towards the categories with a higher number of images. To address this issue, we employ focal loss in combination with the class balance approach in our model, to update the network parameters and mitigate the effects of the imbalanced dataset. The following are the specific steps in the implementation:

First, the probability of predicting each category is calculated:

$$p_i = o(z_i = \frac{1}{1 + e^{z_i}}) \tag{1}$$

where z_i denotes the predicted output of the i category, and o represents the sigmoid function.

Next, the loss function is computed using focal loss in combination with the class balance approach. This is achieved by adjusting the standard focal loss function to include a weight factor for each category based on its proportion in the training dataset. The class balance loss function can be expressed as:

$$L_{CBFL} = -\frac{1}{N} \sum_{i=1}^{N} \alpha_i (1 - p_i)^{\gamma} \log(p_i) \tag{2}$$

where N is the number of samples in the batch, α_i is the weight factor for the i-th category, calculated using the class balance approach, p_i is the predicted probability of the i-th category, and γ is the focusing parameter. The class balance weight factor for each category is computed as the inverse of its frequency in the training dataset, raised to a power β. Thus, categories with low frequency will have a higher weight factor to balance their influence on the training process.

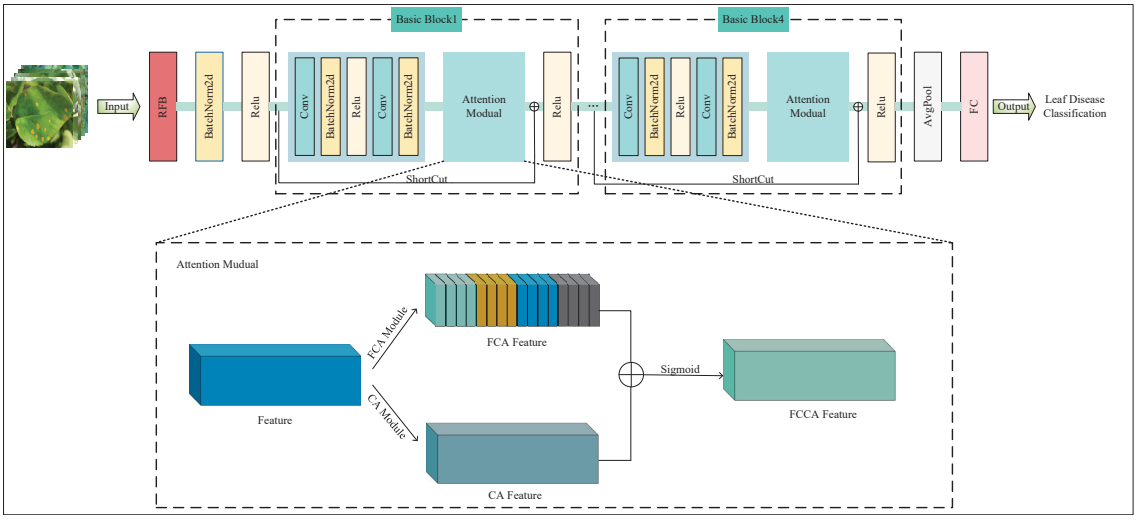


Figure 1. Overall framework of RFCA ResNet.

Incorporating class balance with focal loss helps to mitigate the negative effects of imbalanced categories during training and improves the model's ability to accurately classify plant disease images.

2.2. Topology Fusion

As the number of layers in deep convolutional neural networks increases, the problem of gradient vanishing becomes more pronounced, leading to a decrease in network performance. The ResNet series of networks address this issue by utilizing residual structures that enable the stacking of layers without a loss in performance. The ResNet architecture is widely used in classification tasks due to its effectiveness.

The residual structure adds the input to the output of a layer through a shortcut connection, resulting in a straightforward addition operation that speeds up training without increasing model complexity or the number of required parameters. The precise calculation procedure for the residual is shown in Equation (3):

$$x_{i+1} = x_i + H(x_i, \omega_i) \tag{3}$$

where x_i represents the input of the i-th layer, ω_i represents the parameters of the i-th layer, $H(x_i, \omega_i)$ represents the output of the i-th layer convolution operation, and x_{i+1} represents the residual mapping of the input.

By stacking residual structures, ResNet increases the effectiveness of network training without degradation. To improve the network's ability to extract feature information and enhance the receptive field, the RFB module borrows the structure of the Inception module and adds dilated convolution to the original foundation. The RFB module can extract feature information of different scales by using convolution kernels of different sizes in parallel, making it suitable for the characteristics of lesion features in this experimental dataset. The RFB module's structure is illustrated in Figure 2.

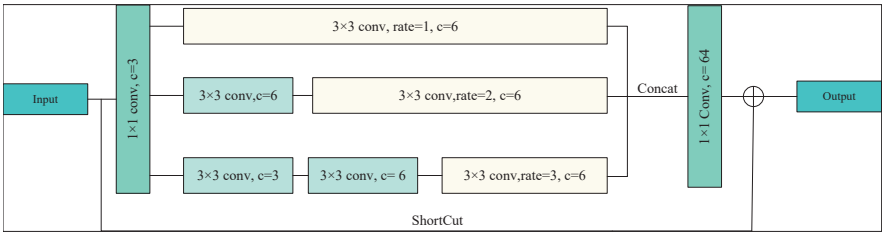


Figure 2. Adjusted RFB module structure.

We present the mathematical reasoning for the receptive field block (RFB) module as follows: Let the input feature map be denoted by x, with dimensions $H \times W \times C$, where H, W, and C represent the height, width, and number of channels, respectively. The output feature map is denoted by Y, with dimensions $H \times W \times N$, where N is the output dimension. The RFB module consists of three parallel branches.

For the first branch, a 3×3 convolution operation, with kernel size $K1$, is performed. The output feature map of this branch, denoted as $F1$, can be expressed as:

$$F1 = Conv(x, K1) \tag{4}$$

where $Conv$ denotes a 3×3 convolution operation.

The second branch includes two consecutive operations: a 3×3 convolution with kernel size $K2$, followed by a 3×3 dilated convolution with kernel size $K2$ and dilation rate of 2, to capture multi-scale contextual information. The output feature map of this branch, denoted as $F2$, can be expressed as:

$$F2 = Conv(Conv(x, K2), K2) \tag{5}$$

where $Conv$ denotes the convolution operation.

In the third branch, three consecutive operations are performed: two successive 3×3 convolutions with kernel size $K3$, followed by a 3×3 dilated convolution with kernel size $K3$ and dilation rate of 3, to capture multi-scale context information. The output feature map of this branch, denoted as $F3$, can be expressed as:

$$F3 = Conv(Conv(Conv(x, K3), K3), K3) \tag{6}$$

where $Conv$ denotes the convolution operation.

After computing the feature maps for all three branches, a 1×1 convolution is applied to adjust the number of channels. The feature maps are then concatenated along the channel dimension to obtain the final output feature map Y:

$$Y = Concat(F1, F2, F3) \tag{7}$$

where $Concat$ represents the concatenation operation along the channel dimension.

To leverage the benefits of each module, we replace the ResNet's convolutional layer with the RFB convolution module to extract low-level feature information. This replacement allows our fused network to accomplish multi-scale extraction of image feature information more effectively than ResNet. As a result, our model's generalization performance and feature discriminability are significantly improved.

2.3. FCCA Attention Module

Images of apple leaf diseases captured in natural settings often feature non-uniformly arranged leaves and complex backgrounds. Accurately identifying these diseases requires incorporating coordinated information on apple disease features present in the image. Existing channel attention approaches do not leverage global pooling to express adequate

information. To address this, we propose integrating coordinate attention with frequency attention to creating a dual attention mechanism, as shown in Figure 3.

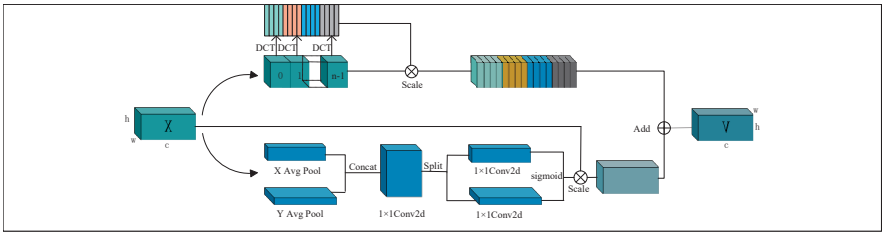


Figure 3. Module structure of FCCA attention mechanism.

The FCCA attention module utilizes input feature maps to simultaneously compute frequency and coordinate attention, employing two softmax multiplications and one additional operation. The mathematical operation of this module can be expressed as shown in Equation (4):

$$FCCA(x) = CA(x) + FA(x) \tag{8}$$

where $FCCA(x)$ denotes the feature map obtained through the dual attention module, $FA(x)$ denotes the frequency feature map, and $CA(x)$ represents the coordinate position feature map.

The FCCA attention module expands the amount of feature information introduced through channels and captures feature information across channels, effectively enhancing the attention of feature channel and position information. This results in increased accuracy in identifying apple leaf diseases.

2.3.1. Coordinate Attention Module

In recent times, several researchers have utilized the SE module proposed by Hu, Jie et al. [27] in their research. This module initially employs global pooling to compress the global spatial information before learning the significance of each channel in the channel dimension. However, it overlooks the importance of position pairs in creating a spatial map. CBAM [28] attempts to incorporate location information using global pooling, but it only considers local range information and cannot establish long-distance relationships. On the other hand, the coordinate attention module provided by Hou Q et al. [29] is a lightweight and effective method, that enhances the expressiveness of learned features by integrating spatial coordinate information into attention maps and capturing the long-distance dependencies of input feature maps.

As illustrated in Figure 4, the process of generating coordinate attention involves two crucial steps: embedding coordinate information and generating attention.

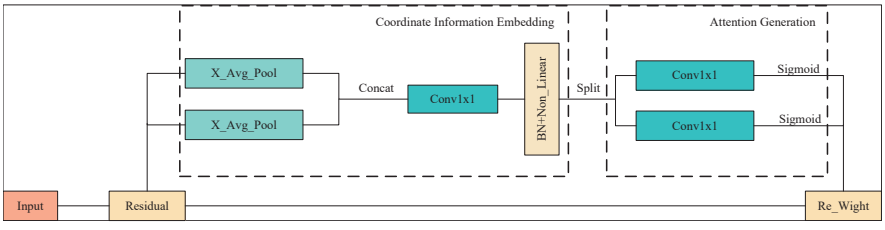


Figure 4. Structure of the coordinate attention mechanism module.

To generate coordinated attention, one-dimensional average pooling is utilized to encode position information in the horizontal and vertical spatial directions and to generate

long-distance dependencies, as global pooling can result in the loss of position information. Specifically, size average pooling kernels of sizes (H, 1) and (1, W) are employed to encode the channels in the two directions, respectively. Thus, the output feature map for the c-th channel and height h is given by:

$$z_c^h(h) = \frac{1}{W}\sum_{i=0}^{W} x_c(h,i), \tag{9}$$

where z_c^h represents the output of the c-th channel in the overall height directions, x_c represents the input of the c-th channel, and W represents the width of the c-th channel input.

The output feature map for the c-th channel and width w is given by:

$$z_c^w(w) = \frac{1}{H}\sum_{i=0}^{H} x_c(i,w), \tag{10}$$

where z_c^w represents the output of the c-th channel in the overall width directions, x_c represents the input of the c-th channel, and H represents the height of the c-th channel input.

To create an attention map, the horizontal and vertical feature maps are transformed using a shared 1×1 convolution kernel. The resulting attention map is then split along the spatial axis and the number of channels is adjusted to match the number of input channels using two 1×1 convolutions. The sigmoid function is applied to normalize the weight, and the coordinated attention module (CA) is expressed as:

$$f = \delta(F(\left[z^h, z^w\right])), \tag{11}$$

$$g^h = o(F_h(f^h)), \tag{12}$$

$$g^w = o(F_w(f^w)), \tag{13}$$

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j), \tag{14}$$

where $[\cdot,\cdot]$ denotes the concatenation operation along the spatial dimension, δ is a non-linear activation function, $f^h \in \mathbb{R}^{C/r\times H}$ and $f^w \in \mathbb{R}^{C/r\times W}$, o is the sigmoid function, and $x_c(i,j)$ represents the output of the c-th channel at position (i,j) in the input image.

2.3.2. Frequency Attention Module

In order to enhance the feature representation ability, the channel attention module is utilized to focus on channels that contain important information by assigning weights to each channel. Typically, the channel relationship is extracted using global average pooling, and the weighted attention map is obtained by applying a fully connected layer and a sigmoid function, which can be expressed as:

$$Attn_{channel} = o(fc(gap(X))) \tag{15}$$

where o is the sigmoid function, fc denotes a fully connected layer, and gap is global average pooling.

Qin Z et al. [30] demonstrated that global average pooling is a special case of discrete cosine transform (DCT), which can result in limited diversity in the features obtained and insufficient representation of information between different channels. To address this issue, they proposed a multi-spectral channel attention (MSCA) module, which first divides the input feature map into multiple groups and applies a two-dimensional DCT (2DDCT)

operation to each group. The resulting frequency feature set is then weighted and fused using a fully connected layer and a sigmoid function, as follows:

$$X = [X_0, X_1, \cdots, X_{n-1}] \tag{16}$$

$$Freq = cat([2DDCT(X_0), 2DDCT(X_1), \cdots, 2DDCT(X_{n-1})]) \tag{17}$$

$$Attn_{fca} = o(fc(Freq)) \cdot X \tag{18}$$

where $X \in \mathbb{R}^{C \times H \times W}$, o is the sigmoid function, fc denotes a fully connected layer, $Freq$ represents the frequency feature set of input features after 2DDCT operation, and n is a constant indicating that the input features are divided into several parts.

3. Experiments

3.1. Dataset Source

This research employed a publicly available dataset, plant-pathology-fpgv8 [22], sourced from the Kaggle website. The dataset comprises 18,632 high-quality photographs classified into 12 categories based on the complexity and diversity of the leaf diseases. Figure 5 depicts the twelve categories in the dataset, and their corresponding names and counts are presented in Table 1.

Table 1. Category name and quantity of apple leaf disease dataset.

Categories	Number of Original Pictures	Number of Pictures after Enhancement
Complex	1441	8356
Frog eye leaf spot	2862	16,794
Frog eye leaf spot complex	148	864
Healthy	4161	23,938
Powdery mildew	1065	6142
Powdery mildew complex	78	446
Rust	1674	9660
Rust complex	87	488
Rust frog eye leaf spot	108	626
Scab	4343	25,136
Scab frog eye leaf spot	617	3606
Scab frog eye leaf spot complex	180	1006

The dataset used in this research exhibits a background of complex disease leaves, a high number of images depicting a single disease, a limited number of images displaying multiple diseases, and an unequal distribution of categories, as illustrated in Figure 5 and Table 1. These characteristics pose significant challenges to accurate disease identification and increase the likelihood of model overfitting.

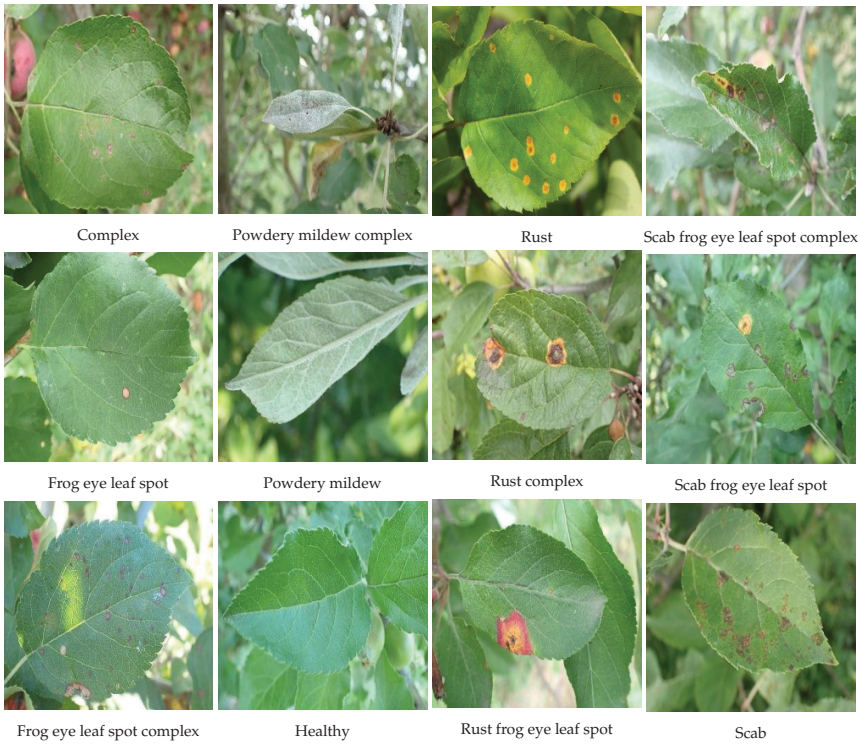


Figure 5. Apple leaf disease dataset category.

3.2. Image Preprocessing and Enhancement

If the pixels of the dataset's images are excessively large and the number of samples is small, both the training infrastructure and the classification network will face significant challenges. The original dataset contains images with resolutions of 4000 × 2760 and 4000 × 3000 pixels. To increase the training set's size and diversity, we cropped the images to 512 × 512 and performed the following three operations: (1) applied color dithering to the image to change saturation, brightness, contrast, and sharpness; (2) randomly rotated the image angle; and (3) added Gaussian noise. Figure 6 shows the exact results of these operations. On the one hand, this process can expand the dataset's diversity and the model's ability to generalize. On the other hand, changing the image's saturation can help to emphasize the lesion. Table 1 shows the number of distinct categories in the dataset.

Figure 6. Image display after data enhancement.

3.3. Equipment

All experiments were conducted on a host CPU with 10 cores, to ensure fairness. Table 2 presents the network model's architecture and other configuration options.

Table 2. Training environment parameter configuration.

Hardware	Software
CPU: NVIDIA GeForce RTX 3060	Windows 11
RAM: 16GB DDR5	Cuda11.1 + Cudnn
CPU:12th Gen Intel Core i5-12600KF	Pytorch1.8.1 + Python 3.8

3.4. Experiment Settings

In this experiment, the original dataset was partitioned into three sets, namely the training set, the validation set, and the test set, using a Python script. The training and validation sets were divided in an 8:1 ratio, with the test set alone consisting of 1868 images.

For the purpose of network training and validation, the images were cropped to 224 × 224 pixels using the center cropping approach, while images of size 512 × 512 pixels were used for testing during the testing phase. To facilitate training, all image data were standardized using Equation (15):

$$X_{out} = \frac{X_{in} - \bar{x}}{\sigma} \tag{19}$$

where X_{out} represents the normalized output result, X_{in} represents the original image input data, x represents the mean values of X_{in}, which are (0.485, 0.456, 0.406), and σ represents the standard deviations, which are (0.229, 0.224, 0.225).

During training, the focal loss function, in combination with the class balance approach, was employed, and the network parameters were optimized using the AdamW optimizer. A batch size of eight was used, with an initial learning rate set at $3 \times e^{-4}$. The cosine annealing strategy was utilized, and the model was trained for 100 epochs. Finally, the predictions were tested, and the optimal training parameters were recorded. It is worth noting that all experiments were executed on a host CPU containing 10 cores, to ensure fairness, and the framework of the network model and other configuration options can be found in Table 2.

3.5. Evaluation Indexes

In evaluating the performance of the classification model on the apple leaf diseases dataset, it is important to note that the dataset has imbalanced data, rendering the accuracy performance index insufficient. To address this limitation, this study employs additional evaluation metrics, such as precision rate, recall rate, and F1 score. The precision rate measures the proportion of properly predicted samples, while the recall rate relates to the proportion of projected positive samples among real positive samples. The F1 score considers both the recall and precision rates, thereby achieving a balanced and optimal outcome. Prior to computing these metrics, one must understand the concept of a confusion matrix, which consists of four components: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), as depicted in Figure 7.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{20}$$

$$Precision = \frac{TP}{TP + FP} \tag{21}$$

$$Recall = \frac{TP}{TP + FN} \tag{22}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{23}$$

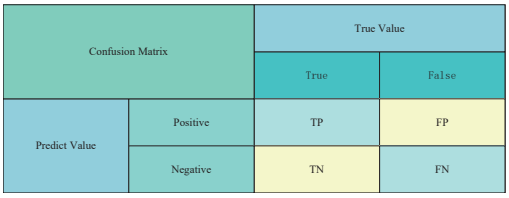


Figure 7. Confusion matrix.

4. Results

The experimental design comprises four distinct sections. Firstly, the impact of varying learning rates on the network model's accuracy is compared. Secondly, the effectiveness of employing data augmentation techniques in identifying apple leaf diseases is evaluated. Thirdly, a comparison is made between the performance of the proposed network model and classical network models. Lastly, an ablation experiment is conducted utilizing the RFCA ResNet network model.

4.1. Comparative Experiments with Different Learning Rates

To investigate the effect of learning rates on image recognition, we employ the control variable method. The initial learning rate is set to 0.01, 0.001, 0.0001, 0.0002, 0.0003, and 0.0004, in order to ensure experiment comparability and increase recognition accuracy. The experiment uses the RFCA ResNet model and trains and tests the original dataset. The learning rate decay strategy, batch size, and training epoch all follow the same guidelines. Table 3 presents the specific training parameters and test results. The highest test accuracy, of 89.61%, is achieved when the learning rate is set to $3 \times e^{-4}$.

Table 3. Parameter configuration and test accuracy of different learning rates.

Learning Rate	Batch Size	Epoch	Training Time	Test Accuracy
0.01	8	100	7 h 57 m 30 s	88.49%
0.001	8	100	8 h 1 m 12 s	89.08%
0.0001	8	100	7 h 52 m 18 s	89.03%
0.0002	8	100	7 h 29 m 5 s	89.13%
0.0003	8	100	7 h 28 m 36 s	89.61%
0.0004	8	100	7 h 56 m 50 s	89.45%

The graph of the model accuracy corresponding to the test learning rate is presented in Figure 8. The results indicate that the model converges slowly and the curve is volatile when the learning rate is high. When the learning rate is set to the e^{-4} level, the curve has iterated nearly 30 epochs and stabilized at an accuracy rate of about 85%.

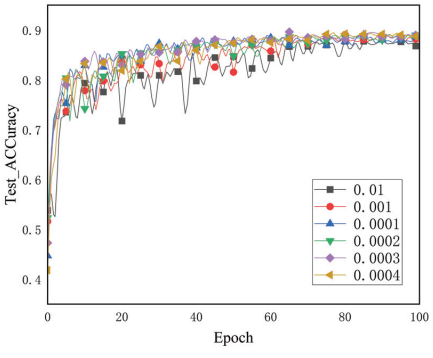


Figure 8. Accuracy of test set under different learning rates.

4.2. Impact of Data Augmentation Methods on Models

A comparative experiment was conducted on the RFCA ResNet network model to verify the effectiveness of the data augmentation strategy in improving the model's accuracy. The recognition accuracy and overall training time on the apple disease test set are presented in Table 4. The results indicate that the model can converge faster and achieve better recognition performance during the same training epoch when the data augmentation strategy is employed, as illustrated in Figure 9.

Table 4. Performance index results without and after enhancement.

Strategy	Accuracy	Precision	Recall	F1 Score
Without enhancement	89.61%	56.66%	72.76%	58.77%
Enhanced	90.58%	55.75%	67.23%	59.44%

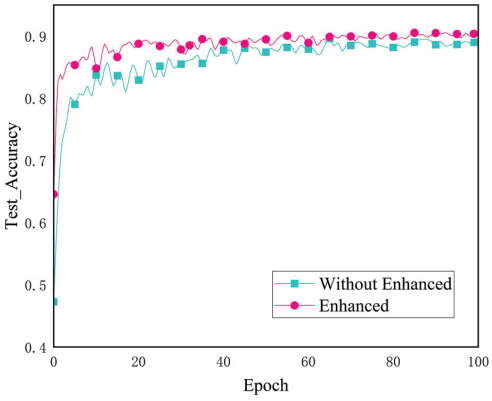


Figure 9. Accuracy of the test set without enhancement and after enhancement.

According to the results presented in Table 4, we observe an improvement of 0.97 and 0.67 percentage points in precision and F1 score, although this requires sacrificing precision and recall for each class. In addition, as shown in Figure 9, the performance of the model converges faster after adopting the data augmentation strategy.

First, the improvements in accuracy and F1 score mean that our method outperforms the state of the art in overall performance. However, sacrificing precision and recall may result in less accurate predictions for some classes. These differences may originate from the imbalance of the dataset and the effect of the adopted data augmentation strategies on different classes to different degrees.

Second, the fast convergence of the model performance demonstrates that our method can utilize limited training data more efficiently, by using data augmentation strategies. This is of great significance for improving model performance under limited resources, especially in large-scale datasets or real-time application scenarios.

4.3. Comparative Experiments of Different Network Models

To demonstrate the superior performance of the RFCA ResNet network model, we conducted a comparative experiment with a commonly used CNN model. As presented in Table 5, RFCA ResNet achieved an average classification accuracy rate, precision rate, recall rate, and F1 score of 89.61%, 56.66%, 72.76%, and 58.77%, respectively, outperforming other CNN methods. Moreover, as depicted in Figure 10, while the loss of the Res2Net network remained nearly constant, the proposed RFCA ResNet achieved faster convergence, indicating that the Res2Net network may not be suitable for this dataset, and the method

suggested in our research is more generalizable. The Densenet121 method connects all channels for feature reuse, which impacts the model's classification accuracy, making it simpler for the model to maintain background noise information in complex contexts. In contrast, the Shufflenet, RegNet, Res2Net, and ConvNeXt neural network architectures may face difficulties in capturing fine-grained details in complex images, potentially hindering their ability to learn sufficiently representative features for all possible variations in apple leaves and complex backgrounds. Therefore, without additional modifications or preprocessing techniques, these networks may not be the most suitable choice for recognizing apple leaves with complex backgrounds.

Table 5. Evaluation index results of different network model training test sets.

Model	Batch Size	Epoch	Params Size	Accuracy	Precision	Recall	F1 score
ResNet34 [31]	8	100	81.22 M	87.58%	49.79%	55.32%	50.22%
ResNet50 [31]	8	100	89.77 M	88.38%	50.54%	60.28%	51.74%
MobilNetV3L [32]	8	100	16.09 M	86.94%	48.29%	52.21%	48.76%
MobilNetV3S [32]	8	100	5.84 M	86.83%	52.90%	66.49%	54.64%
DenseNet121 [33]	8	100	6.54 M	87.79%	49.90%	60.94%	51.04%
RegNet [34]	8	100	8.85 M	83.62%	43.68%	51.24%	44.06%
ShuffleNet [35]	8	100	4.83 M	86.35%	46.13%	48.80%	44.81%
Res2Net [36]	8	100	33.92 M	81.58%	40.93%	40.32%	40.29%
ConvNeXt [37]	8	100	334.02 M	85.70%	45.13%	52.88%	44.69%
RFCA ResNet	8	100	46.38 M	89.61%	56.66%	72.76%	58.77%

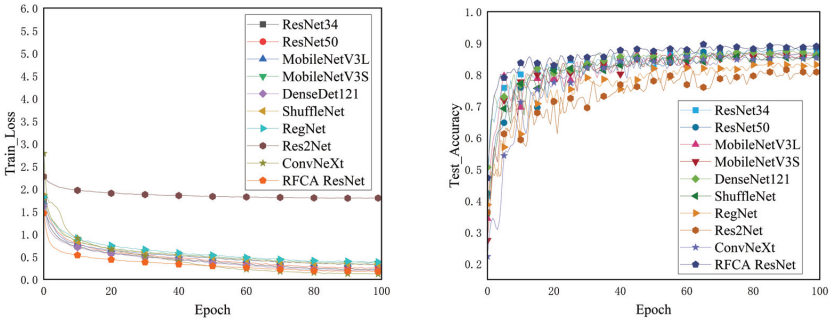


Figure 10. Loss curve of the test set on the training set, and accuracy curve of the test set.

To compare the performance of different models, we plotted the precision–recall (P–R) curve for each model. This approach provides a better evaluation of the performance of each model. Figure 11 shows the P–R curves for each model, represented by different colors. The area covered by the blue curve is the largest, indicating the model has the best classification performance.

4.4. Ablation Experiment

To assess the effectiveness of various modifications made to the ResNet18 network model in improving its performance, we utilized accuracy, precision, recall, and F1 score as evaluation metrics, and the original dataset was used for training and testing. In particular, we replaced the first convolutional layer in the original network when using only the RFB module, incorporated it into the residual module when using only the attention method, and changed the cross-entropy loss function when replacing it with the focal loss in combination with the class balance approach alone. Table 6 presents a comparison of the performance evaluation indicators of the network when adding RFB, class balance with focal loss, embedding the attention module, not adding any module, and the RFCA ResNet model. Additionally, Figure 12 displays the loss changes of the RFCA ResNet network on the training set and the test accuracy change curve of the set.

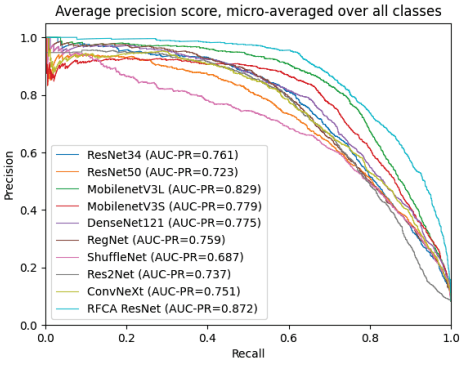


Figure 11. P–R curves of different models.

4.5. Ablation Experiment

To assess the effectiveness of various modifications made to the ResNet18 network model in improving its performance, we utilized accuracy, precision, recall, and F1 score as evaluation metrics, and the original dataset was used for training and testing. In particular, we replaced the first convolutional layer in the original network when using only the RFB module, incorporated it into the residual module when using only the attention method, and changed the cross-entropy loss function when replacing it with the focal loss in combination with the class balance approach alone. Table 6 presents a comparison of the performance evaluation indicators of the network when adding RFB, class balance with focal loss, embedding the attention module, not adding any module, and the RFCA ResNet model. Additionally, Figure 12 displays the loss changes of the RFCA ResNet network on the training set and the test accuracy change curve of the set.

Table 6. Results of ablation experiment performance evaluation index.

Model	RFB	Attention	Class Balance Loss	Accuracy	Precision	Recall	F1 Score
ResNet18	-	-	-	87.95%	40.82%	55.83%	48.60%
	✓	-	-	89.07%	51.67%	56.18%	50.59%
	-	✓	-	88.65%	50.99%	63.64%	52.21%
	-	-	✓	88.44%	52.70%	62.64%	54.40%
RFCA ResNet	✓	✓	✓	89.61%	56.66%	72.76%	58.77%

- indicates absence of the template, while ✓ indicates presence of the template.

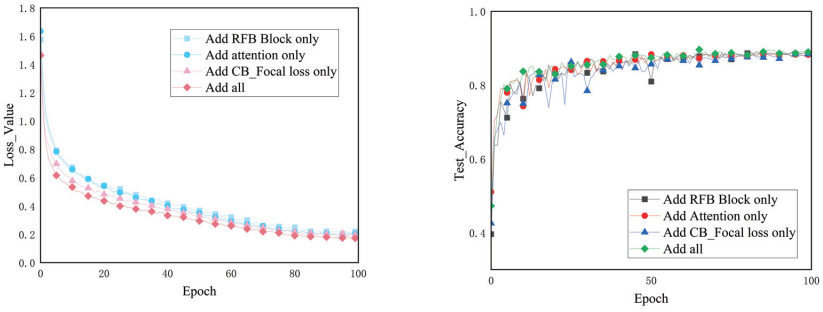


Figure 12. Loss curve of test set on training set, and accuracy curve of test set.

To improve the performance of the ResNet18 network model, a series of enhancements were incorporated. Table 6 presents the evaluation indicators of accuracy, precision, recall,

and F1 score, using the original dataset for both training and testing. The first convolutional layer in the original network was replaced with the RFB module when only the RFB module was used, while the FCCA module was incorporated into the residual module when only the attention method was utilized. When focal loss was used in combination with the class balance approach alone, the cross-entropy loss function was modified. The results showed that adding the attention mechanism to ResNet18 increased the accuracy, precision, recall, and F1 score by 0.7, 10.17, 7.81, and 3.61 percentage points, respectively. The RFB structure was found to broaden the model's receptive field, extract feature information at various scales, and improve its capacity for information representation. This resulted in an increase in accuracy of 1.12 percentage points and in F1 score of 1.99 percentage points. Altering the loss function to account for the effect of an unbalanced dataset on model performance improved all parts of the model's performance evaluation indicators. Finally, the accuracy and F1 score of the RFFA ResNet model on the apple leaf disease dataset were found to be 89.16% and 58.77%, respectively, which were 1.83 and 9.31 percentage points higher than ResNet18. This was achieved by replacing the RFB module, embedding the attention mechanism, and using focal loss in combination with the class balance strategy. Figure 12 shows that the recognition accuracy of the model on the test set was initially unstable but tended to stabilize and perform well afterward, when the focal loss in combination with the class balance approach was employed as the loss function. The training loss value decreased as more iterations were completed after adding the aforementioned modules to ResNet18, and the accuracy rate on the test set increased. This demonstrated the enhanced model's strong generalization capabilities and the value of the several enhancements made to ResNet18 in this research.

5. Discussion

In our research, we propose a novel method for the classification and identification of apple leaf diseases based on a dual attention mechanism and multi-scale feature extraction. Our method is evaluated on a dataset that exhibits common challenges in plant disease classification, including complex backgrounds and class imbalance.

Owing to their effectiveness, attention mechanisms have been widely employed in the field of plant disease recognition. Zhu et al. [38] combined the convolutional block attention module (CBAM) and EfficientNet-B4 to construct the EfficientNet-B4-CBAM model, which improved the ability to express regional information of camellia oleifera fruit and achieved a final model accuracy rate of 97.02%. Lin et al. [39] employed a naive metric few-shot learning network as a baseline learning method, and embedded attention modules of channel, space, and mixed attention types. The experimental results revealed that the incorporation of these attention modules led to varying degrees of improvement in accuracy. In this research, we introduced the FCCA module and evaluated its impact on the baseline accuracy through ablation experiments (refer to Table 6). Our findings indicated that the inclusion of the FCCA module enhanced the baseline accuracy by 0.7%. However, as disease complexity increases, the limitations of attention mechanisms can hinder their effectiveness, necessitating the exploration of models with enhanced feature extraction capabilities. To address this issue, we adopted a multi-scale feature extraction approach inspired by GoogleNet and ResNet, replacing the low-level feature extraction module of ResNet with the RFB module. Our experimental results demonstrated that our approach improved accuracy by 1.12%, making it an innovative and superior method for feature extraction.

In addition, in existing studies on plant disease identification, datasets almost always exhibit a balanced distribution, while studies on datasets exhibiting long-tailed distributions are rare. To address this problem, Hsiao et al. [40] proposed the MTSbag method, which combines MTS with a bagging-based ensemble learning method to enhance the ability of traditional MTS to deal with imbalanced data. Min et al. [41] developed a data augmentation technique that utilizes an image-to-image translation model to address the issue of category number bias by generating additional diseased leaf images to supplement

the insufficient dataset. In this research, we adopted focal loss and class balancing strategies to optimize the model's handling of imbalanced data. With these optimization strategies, our method exhibits significant advantages in handling imbalanced data. Data augmentation is a common method to improve the generalization ability of models. In this paper, we improved the model's accuracy and F1 score by adding Gaussian noise and random rotation, but at the expense of precision and recall. We believe that these evaluation metrics may not fully reflect the model's performance in real-world applications. Future research can design more comprehensive evaluation methods to explore the model's performance in different scenarios and further optimize the model and data augmentation strategies.

In summary, our proposed method for crop disease recognition in complex backgrounds has significant advantages. We adopt a multi-scale feature extraction and attention mechanism, as well as focal loss and class balancing methods to deal with unbalanced data, achieve significant performance improvement, and provide a new approach and method for plant disease recognition.

6. Conclusions

We proposed a novel apple leaf disease classification and recognition method based on multi-scale feature extraction and a dual attention mechanism. Current apple orchard disease diagnosis relies heavily on manual inspection, which consumes significant human and material resources. These factors inspired us to explore deep learning methods for the classification of apple leaf diseases. In our experiments, we evaluated various metrics, including accuracy, precision, F1 score, and recall, and analyzed the following four aspects:

First, we investigated the impact of different learning rates on the network model's accuracy. We found that the highest accuracy, reaching 89.61%, was achieved when the learning rate was 0.0003. In contrast, the accuracy decreased to 88.49% when the learning rate was 0.01. This highlights the importance of selecting an appropriate learning rate during model training.

Second, we studied the effects of data augmentation. By applying random rotation, color balance, and Gaussian noise to the training data, we found that data augmentation could improve the model's performance in terms of accuracy and F1 score by 0.97% and 0.67%, respectively. However, the performance in precision and recall dropped by 0.91% and 5.53%, respectively. Specifically, before using data augmentation techniques, the model's accuracy, precision, recall, and F1 score were 89.61%, 56.66%, 72.76%, and 58.77%, respectively. After applying data augmentation, these metrics changed to 90.58%, 55.75%, 67.23%, and 59.44%, respectively.

Third, we compared the performance of our proposed network model with traditional network models. We found that our model outperformed conventional convolutional neural networks in all considered metrics, including accuracy, precision, F1 score, and recall.

Last, we conducted ablation experiments using the RFCA ResNet network model. We found that each component in our proposed method played a crucial role in the model's performance. Specifically, employing multi-scale feature extraction modules and dual attention mechanisms improved the model's performance, while using the focal loss function and class balancing methods addressed imbalanced data issues. Moreover, the RFCA ResNet network model enhanced the model's robustness.

In summary, we have proposed a method that incorporates multi-scale feature extraction modules and dual attention mechanisms, and applied the focal loss function and class balancing methods to handle imbalanced data for diagnosing apple leaf diseases. Our experimental results have demonstrated that this approach significantly improves the model's performance, outperforming traditional convolutional neural networks. Our research findings have important implications for apple leaf disease diagnosis. However, the model's parameters and computational complexity currently prevent it from being deployed on mobile devices. In the future, we plan to adopt lightweight methods, such as knowledge distillation, to reduce the model's parameter size and computational complexity

while considering resource limitations and processing capabilities on mobile devices, in order to achieve better performance and user experience in mobile deployment.

Author Contributions: Conceptualization, J.D. and Y.W.; methodology, J.D. and Y.Y.; formal analysis, C.Z.; resources, J.D.; data curation, C.Z.; writing—original draft preparation, J.D.; writing—review and editing, J.D. and X.C.; supervision, Y.W.; project administration, Y.W. and Y.Z.; funding acquisition, Y.W. and G.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Anhui Provincial Engineering Laboratory for Beidou Precision Agriculture Information Open Fund Project (BDSYS2021003), the Special Fund for Anhui Characteristic Agriculture Industry Technology System (2021–2025), and the Anhui High School Natural Science Research Project (KJ2019A0211).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We thank Anhui Provincial Engineering Laboratory for Beidou Precision Agriculture Information for supporting our work and providing funding to us. We also thank all of the authors of the primary studies includes in this article. We also wish to thank the anonymous reviewers for their kind advice.

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

References

1. Wu, Z.; Pan, C. State analysis of apple industry in China. In *Proceedings of the IOP Conference Series: Earth and Environmental Science*; IOP Publishing: Bristol, UK, 2021; Volume 831, p. 012067.
2. Mupambi, G.; Anthony, B.M.; Layne, D.R.; Musacchi, S.; Serra, S.; Schmidt, T.; Kalcsits, L.A. The influence of protective netting on tree physiology and fruit quality of apple: A review. *Sci. Hortic.* **2018**, *236*, 60–72. [CrossRef]
3. Duong, L.T.; Nguyen, P.T.; Di Sipio, C.; Di Ruscio, D. Automated fruit recognition using EfficientNet and MixNet. *Comput. Electron. Agric.* **2020**, *171*, 105326. [CrossRef]
4. Gadade, H.D.; Kirange, D. Tomato leaf disease diagnosis and severity measurement. In Proceedings of the 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), London, UK, 27-28 July 2020; pp. 318–323.
5. Habib, M.T.; Mia, M.J.; Uddin, M.S.; Ahmed, F. An in-depth exploration of automated jackfruit disease recognition. *J. King Saud-Univ.-Comput. Inf. Sci.* **2022**, *34*, 1200–1209. [CrossRef]
6. Rozario, L.J.; Rahman, T.; Uddin, M.S. Segmentation of the region of defects in fruits and vegetables. *Int. J. Comput. Sci. Inf. Secur.* **2016**, *14*, 399.
7. Xie, W.; Wang, F.; Yang, D. Research on carrot grading based on machine vision feature parameters. *IFAC-PapersOnLine* **2019**, *52*, 30–35. [CrossRef]
8. Jitanan, S. ; Chimlek, P. Quality grading of soybean seeds using image analysis. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 3495–3503. [CrossRef]
9. Wani, J.A.; Sharma, S.; Muzamil, M.; Ahmed, S.; Sharma, S.; Singh, S. Machine learning and deep learning based computational techniques in automatic agricultural diseases detection: Methodologies, applications, and challenges. *Arch. Comput. Methods Eng.* **2022**, *29*, 641–677. [CrossRef]
10. Zhang, C.; Zhang, S.; Yang, J.; Shi, Y.; Chen, J. Apple leaf disease identification using genetic algorithm and correlation based feature selection method. *Int. J. Agric. Biol. Eng.* **2017**, *10*, 74–83.
11. Nuruzzaman, M.; Hossain, M.S.; Rahman, M.M.; Shoumik, A.S.H.C.; Khan, M.A.A.; Habib, M.T. Machine vision based potato species recognition. In Proceedings of the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 6–8 May 2021; pp. 1–8.
12. Chakraborty, S.; Paul, S.; Rahat-uz Zaman, M. Prediction of apple leaf diseases using multiclass support vector machine. In Proceedings of the 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), Dhaka, Bangladesh, 5–7 January 2021; pp. 147–151.
13. Jia, X.; Song, S.; He, W.; Wang, Y.; Rong, H.; Zhou, F.; Xie, L.; Guo, Z.; Yang, Y.; Yu, L.; et al. Highly scalable deep learning training system with mixed-precision: Training imagenet in four minutes. *arXiv* **2018**. arXiv:1807.11205.
14. Hara, K.; Kataoka, H.; Satoh, Y. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6546–6555.
15. Heusel, M.; Clevert, D.A.; Klambauer, G.; Mayr, A.; Schwarzbauer, K.; Unterthiner, T.; Hochreiter, S. ELU-networks: Fast and accurate CNN learning on imagenet. *NiN* **2015**, *8*, 35–68.

16. Tugrul, B.; Elfatimi, E.; Eryigit, R. Convolutional neural networks in detection of plant leaf diseases: A review. *Agriculture* **2022**, *12*, 1192. [CrossRef]
17. Ramesh, S.; Hebbar, R.; Niveditha, M.; Pooja, R.; Shashank, N.; Vinod, P.; et al. Plant disease detection using machine learning. In Proceedings of the 2018 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C), Bangalore, India, 25–28 April 2018; pp. 41–45.
18. Mohameth, F.; Bingcai, C.; Sada, K.A. Plant disease detection with deep learning and feature extraction using plant village. *J. Comput. Commun.* **2020**, *8*, 10–22. [CrossRef]
19. Jiang, P.; Chen, Y.; Liu, B.; He, D.; Liang, C. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access* **2019**, *7*, 59069–59080. [CrossRef]
20. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**. arXiv:1409.1556.
21. Liu A. ; Wu Y. ; Zhu X. ; Fan G.; Le Y. ; Zhang Y. Corn disease recognition based on deep residual network. *Jiangsu J. Agric. Sci.* **2021**, *37*, 8. (Translated from Chinese: *J. Jiangsu Agric. Sci.* **2021**, *37*, 67–74.)
22. Thapa, R.; Zhang, K.; Snavely, N.; Belongie, S.; Khan, A. The Plant Pathology Challenge 2020 data set to classify foliar disease of apples. *Appl. Plant Sci.* **2020**, *8*, e11390. [CrossRef]
23. Yan, Q.; Yang, B.; Wang, W.; Wang, B.; Chen, P.; Zhang, J. Apple leaf diseases recognition based on an improved convolutional neural network. *Sensors* **2020**, *20*, 3535. [CrossRef]
24. SARDOĞAN, M.; Yunus, Ö.; TUNCER, A. Detection of apple leaf diseases using faster R-CNN. *Düzce ÜNiversitesi Bilim Teknol. Derg.* **2020**, *8*, 1110–1117. [CrossRef]
25. Li, X.; Li, S. Transformer Help CNN See Better: A Lightweight Hybrid Apple Disease Identification Model Based on Transformers. *Agriculture* **2022**, *12*, 884. [CrossRef]
26. 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, 10–17 October 2021; pp. 10012–10022.
27. 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.
28. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
29. 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, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
30. Qin, Z.; Zhang, P.; Wu, F.; Li, X. Fcanet: Frequency channel attention networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 783–792.
31. 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, 27–30 June 2016; pp. 770–778.
32. 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/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
33. 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; pp. 4700–4708.
34. Radosavovic, I.; Kosaraju, R.P.; Girshick, R.; He, K.; Dollár, P. Designing network design spaces. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10428–10436.
35. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6848–6856.
36. Gao, S.H.; Cheng, M.M.; Zhao, K.; Zhang, X.Y.; Yang, M.H.; Torr, P. Res2net: A new multi-scale backbone architecture. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 652–662. [CrossRef]
37. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11976–11986.
38. Zhu, X.; Zhang, X.; Sun, Z.; Zheng, Y.; Su, S.; Chen, F. Identification of oil tea (Camellia oleifera C. Abel) cultivars using EfficientNet-B4 CNN model with attention mechanism. *Forests* **2022**, *13*, 1. [CrossRef]
39. Lin, H.; Tse, R.; Tang, S.K.; Qiang, Z.P.; Pau, G. The Positive Effect of Attention Module in Few-Shot Learning for Plant Disease Recognition. In Proceedings of the 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Chengdu, China, 19–21 August 2022; pp. 114–120.

40. Hsiao, Y.H.; Su, C.T.; Fu, P.C. Integrating MTS with bagging strategy for class imbalance problems. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 1217–1230. [CrossRef]
41. Min, B.; Kim, T.; Shin, D.; Shin, D. Data Augmentation Method for Plant Leaf Disease Recognition. *Appl. Sci.* **2023**, *13*, 1465. [CrossRef]

agriculture

MDPI

Article

Research on Provincial-Level Soil Moisture Prediction Based on Extreme Gradient Boosting Model

Yifang Ren [1], Fenghua Ling [2] and Yong Wang [3,*]

[1] Jiangsu Provincial Climate Center, Nanjing 210008, China; renyifang2023@gmail.com
[2] Institute for Climate and Application Research (ICAR)/CIC-FEMD/KLME/ILCEC, School of Atmospheric Sciences, Nanjing University of Information Science and Technology, Nanjing 210044, China; 20211101018@nuist.edu.cn
[3] School of Applied Meteorology, Nanjing University of Information Science and Technology, Nanjing 210044, China
* Correspondence: ywang@nuist.edu.cn

Abstract: As one of the physical quantities concerned in agricultural production, soil moisture can effectively guide field irrigation and evaluate the distribution of water resources for crop growth in various regions. However, the spatial variability of soil moisture is dramatic, and its time series data are highly noisy, nonlinear, and nonstationary, and thus hard to predict accurately. In this study, taking Jiangsu Province in China as an example, the data of 70 meteorological and soil moisture automatic observation stations from 2014 to 2022 were used to establish prediction models of 0–10 cm soil relative humidity (RH_{s10cm}) via the extreme gradient boosting (XGBoost) algorithm. Before constructing the model, according to the measured soil physical characteristics, the soil moisture observation data were divided into three categories: sandy soil, loam soil, and clay soil. Based on the impacts of various factors on the soil water budget balance, 14 predictors were chosen for constructing the model, among which atmospheric and soil factors accounted for 10 and 4, respectively. Considering the differences in soil physical characteristics and the lagged effects of environmental impacts, the best influence times of the predictors for different soil types were determined through correlation analysis to improve the rationality of the model construction. To better evaluate the importance of soil factors, two sets of models ($Model_{\_soil\&atmo}$ and $Model_{\_atmo}$) were designed by taking soil factors as optional predictors put into the XGBoost model. Meanwhile, the contributions of predictors to the prediction results were analyzed with Shapley additive explanation (SHAP). Six prediction effect indicators, as well as a typical drought process that happened in 2022, were analyzed to evaluate the prediction accuracy. The results show that the time with the highest correlations between environmental predictors and RH_{s10cm} varied but was similar between soil types. Among these predictors, the contribution rates of maximum air temperature (T_{amax}), cumulative precipitation (P_{sum}), and air relative humidity (RH_a) in atmospheric factors, which functioned as a critical factor affecting the variation in soil moisture, are relatively high in both models. In addition, adding soil factors could improve the accuracy of soil moisture prediction. To a certain extent, the XGBoost model performed better when compared with artificial neural networks (ANNs), random forests (RFs), and support vector machines (SVMs). The values of the correlation coefficient (R), root mean square error (RMSE), mean absolute error (MAE), mean absolute relative error (MARE), Nash–Sutcliffe efficiency coefficient (NSE), and accuracy (ACC) of $Model_{\_soil\&atmo}$ were 0.69, 11.11, 4.87, 0.12, 0.50, and 88%, respectively. This study verified that the XGBoost model is applicable to the prediction of soil moisture at the provincial level, as it could reasonably predict the development processes of the typical drought event.

Keywords: soil moisture; prediction; XGBoost algorithm; SHAP

Citation: Ren, Y.; Ling, F.; Wang, Y. Research on Provincial-Level Soil Moisture Prediction Based on Extreme Gradient Boosting Model. *Agriculture* **2023**, *13*, 927. https://doi.org/10.3390/agriculture13050927

Academic Editors: Tarendra Lakhankar and Gerard Arbat

Received: 28 February 2023
Revised: 14 April 2023
Accepted: 17 April 2023
Published: 24 April 2023

1. Introduction

Soil moisture is a critical climate variable that regulates climate change by facilitating the exchange and distribution of water and energy in land–air interaction. Additionally,

soil moisture plays a significant role in agricultural production, as deficits or overflows of soil moisture during critical periods can impact crop growth and yields [1]. Integrating information on available soil moisture and crop water demands can help the development of timely and appropriate irrigation schedules [2], which is particularly important in areas with poor water conditions.

The variations and differences in soil moisture across regions are determined by its budget balance, which is influenced by several factors. Soil moisture is sourced from atmospheric precipitation and artificial irrigation, and its expenditure depends on physical processes such as evapotranspiration and runoff, which are influenced by local weather conditions, soil characteristics, land cover, and other factors [3]. Usually, soil moisture can be expressed using physical variables such as relative humidity, weight water content, and volume water content. Among these variables, relative humidity, calculated as the percentage of soil water content and field capacity, can comprehensively reflect the soil moisture status and surface hydrological processes [4,5]. Consequently, soil relative humidity is an essential reference in irrigation, enabling an analysis of soil moisture differences between regions. Soil moisture prediction based on relative humidity can enhance the defense against waterlogging and drought in farmland.

Numerous studies have investigated soil moisture prediction using various methods. Traditional approaches include the water balance method [6–8], statistical empirical formula method [9], time series method [10,11], and physical models based on hydrological processes [12]. These methods typically consider the soil water budget balance principle, relationships between soil water and environmental factors, change characteristics of soil water over time, and land–air interaction. They use model building or time series analysis to forecast soil moisture. With advances in information technology, various applications of machine learning (ML) in agricultural production have been widely developed, including predictions of the crop growth period, yield, and soil moisture [13–16]. ML technologies such as artificial neural networks (ANNs) [17], support vector machines (SVMs) [18], and gradient boosting regression trees (GBRTs) [19] offer a novel perspective for soil moisture prediction due to their advantages of having a low computational cost, strong self-learning ability, high prediction accuracy, and wide suitability [20–22]. For instance, a GA-BP neural network regression model was tested to perform well in predicting the soil moisture of high side slopes [23]. A proposed novel encoder–decoder model with residual learning played an excellent role in solving the nonlinear problem of soil moisture prediction, which was tested using data from 13 FLUXNET sites with varying plant function types and climatic characteristics [24].

In the research of soil moisture prediction based on machine learning, besides finding suitable prediction models [25], selecting the appropriate input factors for the prediction model is crucial. Many studies have selected meteorological factors directly related to soil moisture, such as precipitation, transpiration, sunshine, and surface temperature [26]. For instance, Xu et al. (2010) developed and tested an integrated soil moisture prediction model based on artificial neural networks (ANNs) with meteorological data in the semi-arid region of eastern China, and the model performed well at basin scales [27]. Li et al. (2018) applied the adaptive genetic ANN method to improve the quality of soil moisture prediction using atmospheric forcing data, which include air temperature, relative humidity, wind speed, radiation, and precipitation, as well as soil forcing data, such as soil temperature at 5 cm depth and lagged soil moisture at 0–10 cm [28]. Moreover, with the advancement of remote sensing technology, remote sensing monitoring indexes based on multi-source data, including optical, thermal infrared, microwave, and other data, have also been widely used for soil moisture monitoring and prediction [29–31].

However, current research on soil moisture prediction has some limitations, including discontinuity in remote sensing images, an inadequate use of data from automatic observation stations, and unclear influencing factors of soil moisture [24,32]. Therefore, this study utilized the soil moisture data and corresponding meteorological data from 70 automatic stations in Jiangsu Province, determined the optimal influence times of the input factors

for prediction models using a correlation analysis method, and applied extreme gradient boosting (XGBoost) to establish two sets of soil relative humidity prediction models (i.e., Model$_{_soil\&atmo}$ and Model$_{_atmo}$). To better interpret the influences of the input factors on these two models and evaluate their performance, Shapley additive explanation (SHAP) was applied, and six metrics were utilized as the predicting effect indicators to compare the models' (e.g., ANN, RF, and SVM) prediction accuracy. Furthermore, a typical drought development process in August 2022 in Jiangsu Province was analyzed in depth. This study aimed to establish a provincial-level and understandable soil moisture prediction model by applying a machine learning algorithm, which could provide a case study for other regions.

2. Materials and Methods

2.1. Study Area

Jiangsu Province (see Figure 1) is located on the east coast of China, in the mid-latitude zone, with a geographical location between 30°46′–35°07′ N and 116°22′–121°55′ E. It lies in the climate transition zone between the subtropical and warm temperate zones and belongs to the East Asian monsoon climate zone. The average annual temperature, precipitation, and sunshine hours in Jiangsu Province are between 13.6–16.1 °C, 704–1250 mm, and 1816–2503 h, respectively [33]. The terrain is generally flat, with the Taihu Plain, Yanjiang, and Lixia River areas being low-lying and having dense water networks. The low mountains and hills account for only 14.33% and are mainly distributed in the west and north regions. There are various soil types in Jiangsu, including zonal soils such as cinnamon, brown soil, yellow-brown soil, and yellow soil, and non-zonal soils such as saline soil, meadow soil, and marsh soil. With a long history of agriculture, natural soil in Jiangsu has evolved into various types of farming soil with different soil textures under the influence of different farming systems and utilization methods [34].

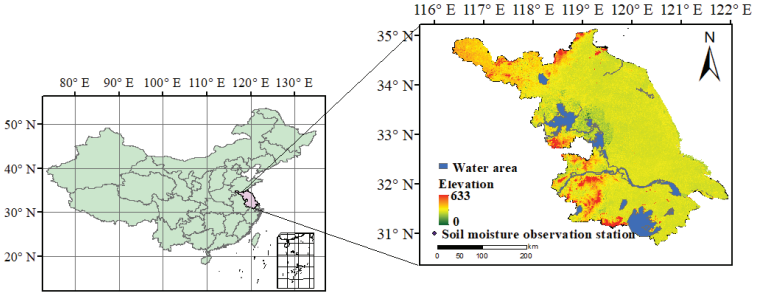


Figure 1. Overview of the study area of Jiangsu Province, China, and its geographical distribution map of soil moisture observation stations.

2.2. Data Source

Automatic moisture observation instruments have been gradually incorporated into the meteorological operational observation system since 2010, resulting in the availability of high regional density and continuous soil moisture observation data across Chinese provinces [35]. Consequently, daily 0–10 cm soil relative humidity data, measured by 70 automatic soil moisture observation stations in Jiangsu Province from 2014 to 2022, along with meteorological data collected by automatic weather stations and soil temperature data measured by soil temperature instruments at the corresponding 70 soil moisture station locations, were used for predicting 0–10 cm soil relative humidity. These atmospheric and soil observation data were obtained from the Jiangsu Meteorological Information Center.

Based on the principle of soil water budget balance and considering the influence of various factors on the 0–10 cm soil relative humidity (RH_{s10cm}, %), the predictive factors were divided into two categories: atmospheric and soil factors. There are ten atmospheric factors, including the mean air temperature (T_a, °C), minimum air temperature (T_{amin}, °C),

maximum air temperature (T_{amax}, °C), air relative humidity (RH_a, %), precipitation (P, mm), sunshine hours (S, h), wind speed (W, ms^{-1}), atmospheric pressure (P_r, hPa), water vapor pressure (e, hPa), and potential evapotranspiration (ET_0, mm). Additionally, there are four soil factors, including the mean surface temperature (T_s, °C), maximum soil surface temperature (T_{smax}, °C), minimum soil surface temperature (T_{smin}, °C), and 0–10 cm soil temperature (T_{s10cm}, °C).

2.3. Data Classification

Soil textures and hydrological constants varied significantly in Jiangsu Province. Even when weather conditions are identical, different regions may exhibit distinct soil water dynamics due to the differences in soil physical properties [36]. Therefore, it is necessary to consider regional soil characteristics and hydrological constants when predicting soil moisture. To this end, according to the soil hydrological and physical characteristics measured by 70 automatic soil moisture observation stations in Jiangsu Province, the soil moisture observation data were classified into three categories: sandy soil, loam soil, and clay soil. The statistics of physical parameters corresponding to the different soil types are shown in Table 1.

Table 1. Classification results and corresponding soil physical characteristics of soil moisture observation data.

Soil Type	Soil Bulk Density ($g \cdot cm^{-3}$)	Field Water Capacity (%)	Withering Humidity (%)	Samples
Sand	1.43	25.46	4.04	40,880
Loam	1.40	26.50	5.29	75,920
Clay	1.36	26.62	5.72	87,600

2.4. Methodology Description

2.4.1. Selection of Predictive Factors

Soil relative humidity changes are mainly affected by previous and current weather conditions and the state of the soil itself. By distinguishing different soil types, we correlated RH_{s10cm} with the averaged or accumulated value (including precipitation and sunshine hours) of 14 predictor factors on the same day as the soil moisture observed, and 1–10 days in the previous period, to determine the maximum impact time of each predictor (see Table 2). We used the time with the largest correlation coefficient of each predictor as its maximum impact time on RH_{s10cm}. The corresponding sample numbers for each soil type used to take correlation analysis are shown in Table 1.

Table 2. List of predictor factors of 0–10 days prior, which are used for correlation analysis with RH_{s10cm}.

Names	Units	Descriptions	Range
Sunshine hours	h	Accumulated sunshine hours	0–128.6
Precipitation	mm	Cumulative precipitation	0–595.4
Evapotranspiration	mm	Averaged potential evapotranspiration	0.1–10.2
Wind speed	ms^{-1}	Averaged wind speed	0–15.9
Relative humidity	%	Averaged mean air relative humidity	19–100
Pressure	hPa	Averaged water vapor pressure	0.6–42.0
		Averaged atmospheric pressure	983.5–1042.4

Table 2. *Cont.*

Names	Units	Descriptions	Range
Temperature	°C	Averaged mean air temperature	−11.1–36.0
		Averaged minimum air temperature	−15.6–31.9
		Averaged maximum air temperature	−7.2–40.9
		Averaged mean soil surface temperature	−7.0–45.8
		Averaged minimum soil surface temperature	−14.7–31.2
		Averaged maximum soil surface temperature	−0.9–70.2
		Averaged 0–10 cm mean soil temperature	−2.7–39.0

2.4.2. XGBoost Model

The XGBoost is an ensemble learning method based on boosting [37]. The boosting technique combines multiple decision trees and aggregates their predictions to obtain a final prediction that is more accurate than any individual tree. XGBoost is designed to prevent over-fitting. The XGBoost model builds multiple trees sequentially, with each subsequent tree intended to reduce the errors of the previous tree. As the training proceeds iteratively, new trees are added to predict the error of the prior tree. Such a fitting process is repeated several times until a stopping criterion is met, such as when the root mean square error (RMSE) reaches an asymptotic value. The ultimate prediction of the model is the sum of the predictions from all of the trees. The formula for the prediction at the step t and site location i can be defined as follows [37]:

$$\hat{y}_i^t = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \quad (1)$$

where $f_t(x_i)$ is the tree model at step t, $\hat{y}_i^t$ and $\hat{y}_i^{(t-1)}$ are the predictions at steps t and $t-1$, and x_i are the predictor variables. The parameters of the model $f(x_i)$ are selected by optimizing the objective function, and the objective function is defined by root mean square error.

Additionally, XGBoost offers several other advanced features [37] that can further enhance the model's performance. For instance, early stopping allows the training process to be stopped early if the performance on a validation set stops improving. This advanced feature prevents the model from overfitting to the training data and can improve its ability to generalize to new data. Cross-validation is another useful technique that can estimate the model's generalization performance and help to select the optimal hyperparameters. By incorporating these and other advanced features, XGBoost has emerged as one of the most popular and influential machine learning models. The flow chart depicting the XGBoost model is presented in Figure 2.

2.4.3. The Key Parameters of XGBoost Model

In this study, we focused on optimizing several crucial parameters of the XGBoost algorithm, including the number of boost rounds, maximum depth, minimum weight in a child, and learning rate. The number of boost rounds determines the maximum number of boosting iterations, while the maximum depth sets the maximum depth of an individual tree. The minimum weight in a child parameter is utilized to prevent overfitting, and the learning rate parameter controls the model's shrinkage at every step (i.e., a lower learning rate indicates more steps used to achieve the optimum) (see Figure 2).

To optimize these parameters, we applied a tuning technique called grid search [38]. This approach computes the optimal values of hyperparameters by exhaustively searching over a range of possible parameter values. We utilized third-fold cross-validation [39] to evaluate the performance of different parameter combinations. In total, we searched through 1500 combinations of parameter values. Ultimately, our XGBoost model achieved

the best performance with the maximum depth, minimum weight needed in a child, and learning rate equal to 15, 10, and 0.02, respectively. In addition, we set the maximum number of boosting rounds to 5000 during training and used the early stop technique to stop the training. The final number of iterations was 4218 when the loss on the validation set no longer decreased.

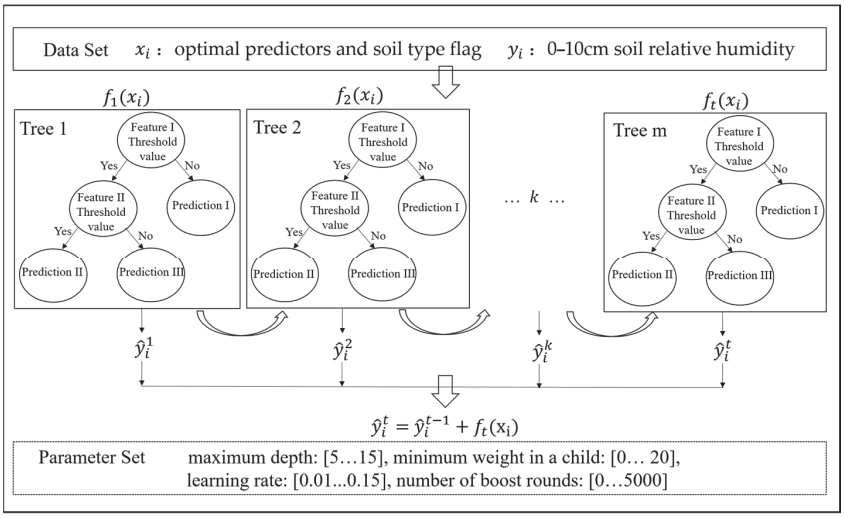


Figure 2. The flowchart of the XGBoost model.

2.4.4. Shapley Additive Explanations (SHAPs)

SHAP is a local attribution method that is based on the use of Shapley values. The Shapley values originate from the field of cooperative game theory and represent each play's average expected marginal contribution in a cooperative game after all possible combinations of players have been considered. It can be formulated as follows [40]:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(F-|S|-1)!}{F!} [f_x(S \cup \{i\}) - f_x(S)] \quad (2)$$

where ϕ_i is the weighted average of all marginal contributions of the predictor i, F is the total number of features, S is the subset of predictors from all predictors except for predictor i, and $\frac{|S|!(F-|S|-1)!}{F!}$ is the weighting factor counting the number of permutations of the subset S. $f_x(S)$ is the expected output given the predictors subset S. $[f_x(S \cup \{i\}) - f_x(S)]$ is the difference made by the predictor i.

2.4.5. Model Construction and Application

This study aimed to develop a soil moisture prediction model for different soil types using relevant atmospheric and soil factors. To achieve this, 14 most related factors were obtained by calculating the correlation. Additionally, to account for the different impacts of soil types, the variable $\mathrm{St_{flag}}$ was included in the model, with values of 1, 2, and 3 representing sandy, loam, and clay soils, respectively.

To further evaluate the importance of soil factors in predicting 0–10 cm soil relative humidity, two sets of data used as the model's independent variables were constructed using 14 optimal predictors (including atmospheric and soil variables) and 10 optimal predictors (including atmospheric variables only) from 70 stations in Jiangsu Province between 2014 and 2021. Before prediction, missing values in these two data sets were completed with the mean values, and the dataset was normalized. A tri-fold cross-validation approach [39] was employed to train, validate, and evaluate the model. The data were

randomly divided into three sets: 80% (163,520 samples) as the model training dataset, 10% (20,440 samples) as the model validation dataset for parameter optimization, and the remaining 10% (20,440 samples) as the model prediction evaluating dataset.

2.4.6. Model Prediction Effect Interpretation and Verification

After building the prediction model, the SHAP method was applied to obtain each predictive factor's positive and negative effects separately for both $\text{Model}_{\text{\_soil\&atmo}}$ and $\text{Model}_{\text{\_atmo}}$. In addition, six metrics were used on the evaluating dataset to evaluate the performance of XGBoost and other state-of-the-art predictive models, including correlation coefficient (R), root mean square error (RMSE), mean absolute error (MAE), mean absolute relative error (MARE), Nash–Sutcliffe efficiency coefficient (NSE), and accuracy (ACC). These indicators are calculated as follows [41]:

$$\mathrm{R} = \frac{\sum_{i=1}^{n}(y_i - \overline{y}_i)(\hat{y}_i - \overline{\hat{y}}_i)}{\sqrt{\sum_{i=1}^{n}(y_i - \overline{y}_i)^2 \sum_{i=1}^{n}(\hat{y}_i - \overline{\hat{y}}_i)^2}} \tag{3}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{4}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|(y_i - \hat{y}_i)| \tag{5}$$

$$\mathrm{MARE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{(y_i - \hat{y}_i)}{y_i}\right| \tag{6}$$

$$\mathrm{NSE} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(\hat{y}_i - \frac{\sum_{i=1}^{n} y_i}{n})^2} \tag{7}$$

$$\mathrm{ACC} = 1 - \frac{1}{n}\sum_{i=1}^{n}\left|\frac{(y_i - \hat{y}_i)}{y_i}\right| * 100\% \tag{8}$$

where y_i is the observed value, $\hat{y}_i$ is the predicted value, n is the number of samples, $\overline{y}_i$ is the mean of observations, and $\overline{\hat{y}}_i$ is the mean of the prediction.

To further verify the prediction capabilities of $\text{Model}_{\text{\_soil\&atmo}}$ and $\text{Model}_{\text{\_atmo}}$ based on XGBoost, we compared these models with three state-of-the-art machine learning models (i.e., ANN [42], RF [43], and SVM [44]) for soil moisture prediction over 70 sites in Jiangsu. The comparison was based on the values of these above metrics and the scatter distributions of predicted and observed soil moisture values. Furthermore, we evaluated the performance of $\text{Model}_{\text{\_soil\&atmo}}$ and $\text{Model}_{\text{\_atmo}}$ during a typical drought in August 2022 in Jiangsu Province. The flow chart depicting the establishment, interpretation, and evaluation of the prediction models for soil moisture is presented in Figure 3.

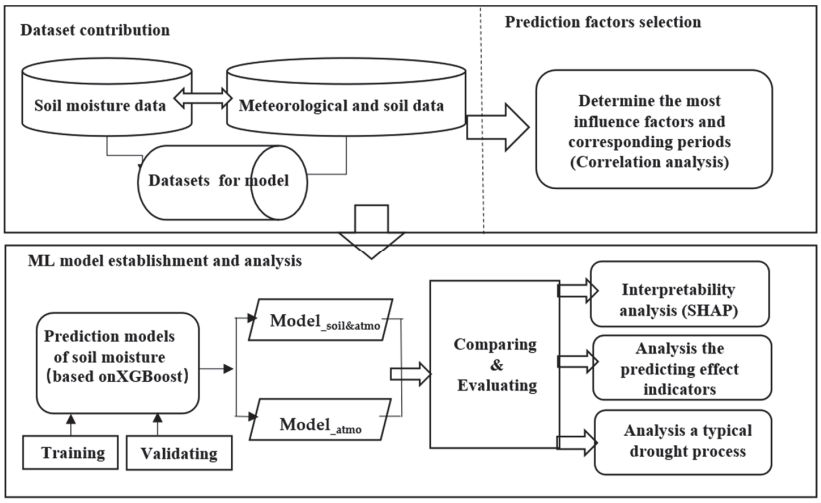


Figure 3. Flow chart of establishing, interpreting, and evaluating soil moisture models.

3. Results

3.1. Correlation Analysis between Soil Moisture and Predictive Factors

After analyzing the correlations between 0–10 cm soil relative humidity (RH_{s10cm}) and various predictors for different soil types with different advance days (See Figure 4), it was observed that, among the atmospheric factors, RH_{s10cm} had a high positive correlation with the mean air relative humidity (RH_a) and cumulative precipitation (P_{sum}). The correlation coefficients were between 0.17–0.33 and 0.13–0.26, respectively, and their absolute values gradually increased with the leading time, peaking 8–10 days prior. Additionally, RH_{s10cm} had a high negative correlation with the mean water vapor pressure (e) and accumulated sunshine hours (S_{sum}). The absolute correlation coefficients were between 0.24–0.33 and 0.15–0.33, respectively. The absolute values also increased with the leading time, reaching the maximum at 8 and 10 days prior, respectively. Among the soil factors, RH_{s10cm} had a high negative correlation with the mean maximum surface temperature (T_{smax}), with its maximum absolute value appearing 4–5 days prior. The correlations between RH_{s10cm} and other factors were relatively low, but all passed the significance test of $p = 0.01$.

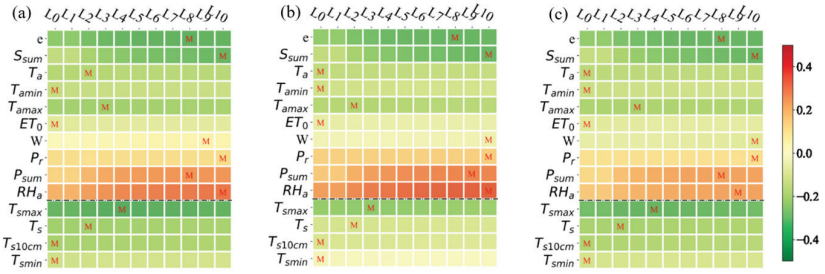


Figure 4. Correlation coefficients between 0–10cm soil relative humidity and various predictive factors of different soil types, which are (**a**) sandy soil, (**b**) loam, and (**c**) clay, respectively.

Overall, the correlations between RH_{s10cm} and various predictor factors, as well as their change rules with the days advanced, were relatively consistent among different soil types, with the times taken to reach the maximum value being similar (see Figure 4a–c). The variabilities of positive–negative correlation with RH_{s10cm} were mainly reflected in the factors of the minimum surface temperature and wind speed. Thus, a fixed optimal impact

time was set for each predictor factor as the model input, and its corresponding differences in the impact times between different soil types were no longer distinguished.

3.2. Interpretability of Model

We analyzed the relationships between the predictor variables and the soil moisture using the XGBoost model and presented the results through SHAP summary plots for each variable. In Figure 3, for each predictor variable displayed on the *y*-axis, each colored point represents a value of this variable in the dataset and the SHAP values displayed on the *x*-axis denoting the contributions of that predictor variable, which can be a positive or negative effect on the prediction of soil moisture. The gradient color of each point indicates the value of the predictor variable, ranging from low (blue) to high (red), providing a visual representation of the relationships between the predictors and soil moisture.

From the SHAP summary chart of Model$_{_soil\&atmo}$ in Figure 5a, we observed that T_{smax}, T_{s10cm}, and T_{amax} had a significant negative contribution to the model prediction, considering both atmospheric and soil variables. Conversely, the effects of other factors on the prediction results were either opposite or insignificant. Among them, P_{sum} had the most considerable positive contribution to the model prediction, followed by RH_a. According to the importance of each predictor, the order of the top five predictors was $T_{smax} > P_{sum} > T_{s10cm} > RH_a > T_s$.

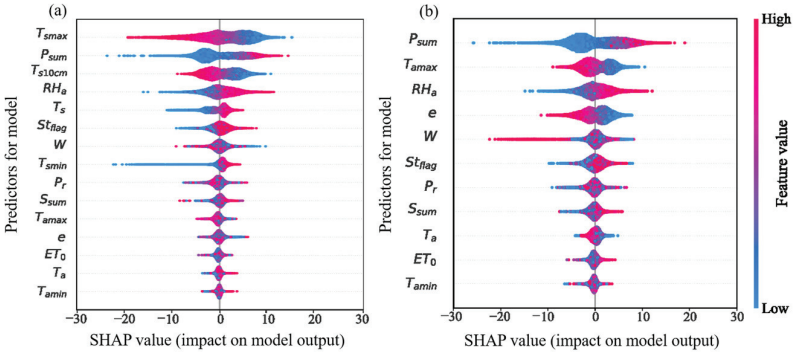


Figure 5. SHAP summary chart of (**a**) Model$_{_soil\&atmo}$ and (**b**) Model$_{_atmo}$.

From the SHAP summary chart of Model$_{_atmo}$ in Figure 5b, we found that the greater value of T_{amax}, e, and W had a greater negative contribution to the model prediction, considering only atmospheric variables. In contrast, other factors have opposite effects on the prediction results, or their positive–negative characteristics were insignificant. Among them, P_{sum} had the most significant positive contribution to the model prediction, followed by RH_a, which was consistent with the results of Model$_{_soil\&atm}$. According to the importance of each predictor, the order of the top five predictors was $P_{sum} > T_{amax} > RH_a > e > W$.

3.3. Model Prediction Evaluation

3.3.1. Analysis of Model Prediction Accuracy

To further verify the prediction capabilities of Model$_{_soil\&atmo}$ and Model$_{_atmo}$ based on XGBoost, we compared them with three other state-of-the-art machine learning models (i.e., ANN, RF, and SVM) based on the scatter distributions of the predicted and observed values of soil moisture, and the values of six metrics (i.e., R, RMSE, MAE, MARE, MSE, and ACC).

The scatter distributions of the model predictions based on XGBoost and the actual observations of the 0–10 cm soil relative humidity are presented in Figure 6a1,a2. Model$_{_soil\&atmo}$ and Model$_{_atmo}$ showed an even distribution of predicted and observed values around the 1:1 diagonal, with Model$_{_soil\&atmo}$ exhibiting a slightly more clustered distribution. The mean and standard deviation of Model$_{_soil\&atmo}$'s predictions (79.28% and

10.32%, respectively) were similar to those of the observations (79.30% and 15.77%, respectively). Model$_{_atmo}$'s prediction results were comparable to those of Model$_{_soil\&atmo}$, with only minor differences. However, overall, the prediction performance of Model$_{_soil\&atmo}$ was slightly better than that of Model$_{_atmo}$.

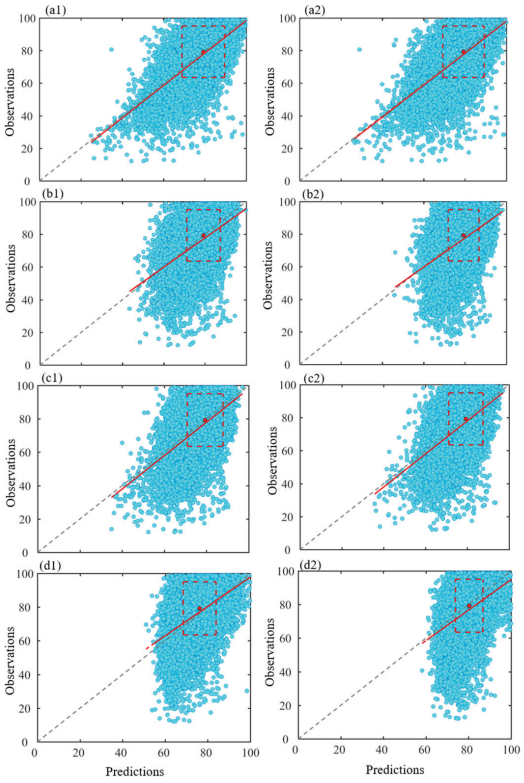


Figure 6. Scatter plot of soil moisture observations and predictions of Model$_{_soil\&atmo}$ and Model$_{_atmo}$ based on (**a1**,**a2**) XGBoost, (**b1**,**b2**) ANN, (**c1**,**c2**) RF, and (**d1**,**d2**) SVM. (The 1:1 diagonal is shown by the gray dashed line, the regression line is shown by the red solid line, and the observed and predicted means and standard deviations are shown by the red dots and dashed boxes, respectively).

After comparing the scatter distributions of observations with model predictions based on XGBoost, ANN, RF, and SVM (see Figure 6), it was observed that the lines between the predicted and observed soil moisture for XGBoost were much closer to the ideal line ($y = x$) than those for the other predictive models. Additionally, the prediction results of the other models presented a relatively smaller standard deviation.

Table 3 shows the comprehensive predictive performances of XGBoost, ANN, RF, and SVM over 70 sites in Jiangsu Province. The values of R, RMSE, MAE, MARE, NSE, and ACC for Model$_{_soil\&atmo}$ and Model$_{_atmo}$ based on XGBoost were 0.69, 11.11, 4.87, 0.12, 0.50, and 88%, as well as 0.66, 11.49, 4.96, 0.14, 0.47, and 86%, respectively. Comparing the values of the six evaluated indexes of other LM models, it was found that models based on XGBoost always had the lowest RMSE, MAE, and MARE, as well as the highest R, NSE, and ACC.

In addition, for XGBoost, compared with Model$_{_atmo}$ having an average prediction accuracy of 86%, Model$_{_soil\&atmo}$ had better precision, with an average accuracy of 88%. Notably, Model$_{_soil\&atmo}$'s prediction effects were always slightly better than those of Model$_{_atmo}$, which was also evident from the prediction results of other models, whether from the scatter charts or metrics.

Table 3. Comparison of XGBoost, ANN, RF, and SVM performances in soil moisture prediction using two data sets as the model's input.

ML	Models	R	RMSE	MAE	MARE	NSE	ACC (%)
XGBoost	$Model_{\_soil\&atmo}$	0.69	11.11	4.87	0.12	0.50	88%
	$Model_{\_atmo}$	0.66	11.49	4.96	0.14	0.47	86%
ANN	$Model_{\_soil\&atmo}$	0.59	12.85	6.55	0.16	0.27	84%
	$Model_{\_atmo}$	0.56	13.19	6.71	0.17	0.23	83%
RF	$Model_{\_soil\&atmo}$	0.64	12.08	6.07	0.15	0.36	85%
	$Model_{\_atmo}$	0.63	12.25	6.19	0.16	0.34	84%
SVM	$Model_{\_soil\&atmo}$	0.54	13.68	7.56	0.17	0.19	83%
	$Model_{\_atmo}$	0.51	13.58	6.86	0.18	0.18	82%

Furthermore, the spatial distribution map of the model evaluation indexes (i.e., R and MAE) showed that both $Model_{\_soil\&atmo}$ and $Model_{\_atmo}$ based on XGBoost had a high accuracy in soil moisture prediction, and their spatial distribution patterns were very similar, with differences only at individual stations (see Figure 7). Stations with relatively small correlation coefficients and large average absolute errors of predictions and observations of both models were mainly concentrated along the northern area of the Yangtze River and in the northeastern area of Jiangsu Province.

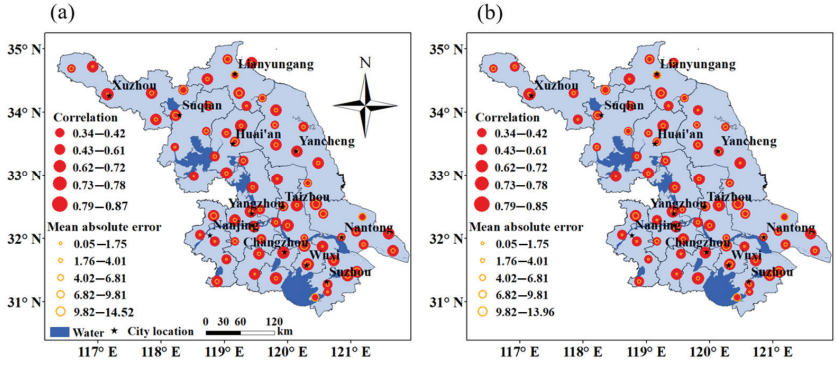


Figure 7. Spatial distribution of prediction accuracy evaluation indicators of (**a**) $Model_{\_soil\&atmo}$ and (**b**) $Model_{\_atmo}$.

In addition, we found that the prediction accuracy of both models varied greatly between sites from the spatial distribution maps. According to the statistical analysis, for $Model_{\_soil\&atmo}$, the R between the predicted and measured values ranged from 0.34 to 0.87, with a mean value of 0.69, and the MAE ranged from 0.12% to 14.52%, with a mean value of 4.87%. The number of sites with $R > 0.60$ reached 58, accounting for more than 82%, and the number of sites with MAE < 5% reached 40, accounting for more than 57%. For $Model_{\_atmo}$, the R between the predicted and measured values ranged from 0.34 to 0.85, with an average value of 0.66, and the MAE ranged from 0.05% to 13.96%, with an average value of 5.04%. The number of sites with $R > 0.60$ reached 53, accounting for more than 75%, and the number of sites with MAE < 5% reached 38, accounting for more than 50%.

3.3.2. Analysis of Typical Drought Process

During 2–23 August 2022, a third round of persistent high temperature occurred in Jiangsu Province, with the first two rounds taking place on 16–22 June and 8–15 July, respectively. The south of Huaihe region experienced 14–19 days of a maximum temperature ≥ 37 °C, with the average temperature between 32–33.7 °C. Compared to the same period in a normal year, the temperature in 2022 was approximately 4 °C higher and the precipitation was less than 90%. In particular, southern Jiangsu faced widespread high

temperatures above 40 °C from 12–15 August, resulting in a rapid expansion of drought across the province. By 15 August, most of the southern Huaihe Basin experienced moderate or above meteorological drought, with some areas suffering from severe drought. However, the high temperature gradually receded from 24 August, and the precipitation gradually increased, mainly in the Huaibei and Sunan areas. As a result, the moisture conditions across the province improved effectively, and the moisture content reached an appropriate level.

According to the distribution of a 0–10 cm soil relative humidity on 1, 15, and 30 August, which was interpolated from the measurement of the automatic soil moisture station (see Figure 8a1–a3), we found on 1 August, affected by antecedent precipitation, the soil moisture in most areas of northern Jiangsu was saturated, and the field humidity was relatively high, while the 0–10 cm soil relative humidity in some areas of southern Jiangsu was less than 60%. By 15th August, there was a severe soil water shortage in most of the southern Huaihe Basin. The 0–10 cm soil relative humidity was only 40% to 50%, which had reached moderate drought, and was even less than 40% in some regions, reaching severe drought. Affected by precipitation, by 30 August, the field soil humidity in some areas of Huaibei was relatively high, and the 0–10 cm soil relative humidity in most southern Huaihe Basin had generally improved to more than 60%, with only sporadic areas still suffering from the drought. Thus, it can be seen that the variation in farmland drought perfectly corresponds with the beginning, aggravation, and extinction of the entire high-temperature process.

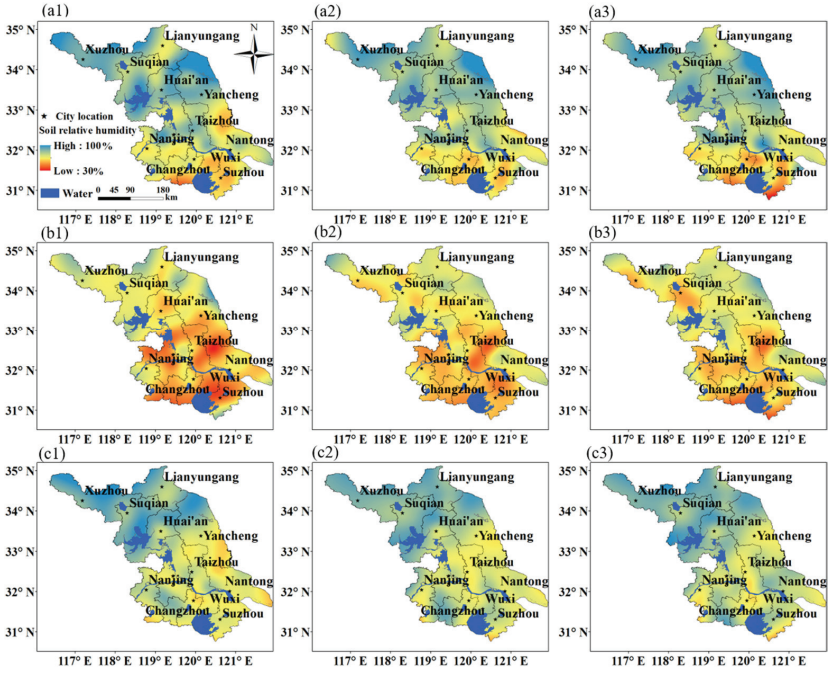


Figure 8. Relative humidity of 10 cm soil relative humidity of (**a1**–**a3**) observations, (**b1**–**b3**) Model_soil&atmo predictions, and (**c1**–**c3**) Model_atmo predictions on 1, 15, and 30 August 2022.

The spatial distribution patterns of the corresponding prediction results of the models agreed with the observation results. The prediction results reflected not only the development process of drought but also the distribution areas of different levels of farmland drought. However, the predicted drought situation was relatively weak compared to the observation results. Overall, the differences in the distribution pattern and numeric

value between the predictions and observations of Model$_{_soil\&atmo}$ were less than those of Model$_{_atmo}$ (see Figure 8b1–b3 and Figure 8c1–c3, respectively).

4. Discussion

Based on the observation, soil types, and meteorological data, this study adopted XGBoost to predict soil moisture variations. Different atmospheric and soil factor combinations were selected as input variables to establish two sets of prediction models (Model$_{_soil\&atmo}$ and Model$_{_atmo}$) for RH_{s10cm}. At the same time, the contributions of the predictive factors were discussed using SHAP. The prediction accuracy was evaluated by comparing six evaluated indexes with other popular ML methods and analyzing a typical drought process in 2022.

The variation in soil moisture is a complex coupling system that exhibits high noise, nonlinearity, and unstable random time series data [22]. Compared to traditional statistical models, machine learning algorithms use multiple processing layers consisting of complex structures or multiple nonlinear transformations to highly abstract data, which could overcome the influence of white noise on the prediction accuracy and effectively improve the simulation accuracy [25]. However, different ML methods have different applicabilities for the same dataset. For example, in a study predicting soil moisture based on three different datasets, machine learning techniques such as multiple linear regression (MLR), support vector regression (SVR), and recurrent neural networks (RNNs) were compared, and MLR was found to have a better performance than the others. Our study used automatic soil moisture observations to compare the prediction accuracies of two models based on XGBoost with ANN, RF, and SVM. It showed that Model$_{_soil\&atm}$ based on XGBoost was superior, providing the lowest RMSE (11.11), MAE (4.87), and MARE (0.12), and highest R (0.69), NSE (0.50), and ACC (88%). Due to different research and application purposes, the dataset applied in soil moisture prediction studies based on machine learning algorithms is varied, including in situ sites [45], remote sensing [46], reanalysis [47], and flux stations [24]. These datasets usually belong to diverse regions with different spatial and temporal resolutions, so it is still challenging to make direct comparisons even if the same method is applied.

The analysis of a typical drought process showed that the XGBoost model based on site data had a good performance and was a feasible method for soil water content prediction, as it could capture a reasonable spatial distribution of the soil moisture. In addition, several advantages were considered for choosing the data observed from the automatic observation stations. Firstly, for a specific site, the data of the automatic observation station have lower errors than the data obtained by remote sensing instruments and reanalysis data, where the problems of insufficient time resolution and delayed acquisition also exist [47]. Hence, we can more accurately explore the relationship between soil moisture and environmental parameters. Secondly, soil moisture and its related meteorological or soil data are commonly available with the exact temporal resolution, so abundant data could be provided for training the predictive model. It is important to note that the predictivity of soil moisture depends on the data's time steps and spatial resolutions due to their different distribution and variation [24,48]. Moreover, the wideness of the application of soil moisture prediction usually depends on its spatial representativeness. Therefore, as more automatic weather stations are installed, the proposed model based on site data could be helpful for the operational studies on soil moisture prediction over larger regions and could provide information for timely and optimal irrigation scheduling. However, considering the spatial variability of soil moisture, in-depth future research is still needed, using situ data, remote sensing, and reanalysis data.

The appropriate selection of model input factors could promote the accuracy of the prediction model [49]. In this research, we correlated the RH_{s10cm} with 14 predictors 1–10 days before to determine each predictor's maximum impact time. The selected predictors were taken as inputs for the model, which would make the model establishment more reasonable, but still needs to be tested in the future. In addition, the contributions of

each predictor on the modeling results of two sets of models were discussed via SHAP. The analysis revealed that soil factors in $Model_{\_soil\&atmo}$ played a positive role in the prediction of soil moisture. Overall, the prediction accuracy of $Model_{\_soil\&atmo}$ was higher than that of $Model_{\_atmo}$. Therefore, introducing soil factors such as T_{smax}, T_s, and T_{s10cm} could improve the prediction accuracy of soil moisture to some extent. For atmospheric factors, T_{amax}, P_{sum}, and RH_a are crucial for improving the soil moisture prediction accuracy. These results are consistent with the view that temperature and precipitation are the main factors affecting the variations in soil moisture by adjusting the water budget balance [50,51].

This study aimed to predict the 0–10cm soil relative humidity, which is a crucial parameter for drought and waterlogging prevention, as well as farmland fertilization and irrigation. Generally, the cultivation layer of crops is 0–20 cm, and the water condition of this layer has a good characterization of crop drought. However, compared with the deep soil layer, the 0–10 cm soil layer is more directly affected by meteorological conditions such as precipitation and temperature. When the temperature is high and the amount of evapotranspiration increases, the lack of moisture in crop fields appears gradually from top to bottom. The moisture deficit in surface soil is easily detected and can serve as the evaluation index for preventing and controlling crop drought. In addition, there is an excellent linear correlation between the soil relative moisture at different levels of depth [52], and hence the surface soil moisture condition is a good indicator of deep soil moisture conditions.

This study deeply integrated the XGBoost with meteorological data to establish a provincial-level soil moisture prediction model, which can provide a reference for soil moisture prediction research in other regions. The model can be used to analyze historical soil water change rules and typical drought and flood cases during the period lacking soil moisture observation while high-density meteorological observation is available (mainly from the 1960s to 2010s). However, there are some deficiencies and uncertainties in this study. For instance, only four frequently used machine learning algorithms were used in the study. In the future, multiple machine learning algorithms or other methods [53–55] could be used to conduct soil moisture prediction research to analyze the advantages and disadvantages of different methods and applicable conditions. Based on the XGBoost algorithm, the positive and negative contributions of most factors in the $Model_{\_soil\&atmo}$ and $Model_{\_atmo}$ for soil moisture prediction analyzed by SHAP were consistent and conformed to the actual physical meaning. However, there were some cases where the same factor had the opposite contribution to the prediction results, which needs further investigation.

5. Conclusions

Soil moisture is the characterization of farmland drought and flood and the basis for irrigation schemes. The prediction of soil relative humidity was achieved based on the XGBoost model using continuous daily atmospheric and soil observation data from automatic stations. The methods of correlation analysis and SHAP were applied to select model predictors and evaluate the contribution of model factors. In addition, six effect indicators and a typical drought process were analyzed to compare the predictive accuracy of the XGBoost model with the other three machine learning models (i.e., ANN, RF, and SVM) to assess the predictive power of the model.

Through correlative analysis, we found that the time with the highest correlations between environmental predictors and RH_{s10cm} varied but was similar between soil types. Among atmospheric factors, the mean RH_a and P_{sum} exhibited strong positive correlations with RH_{s10cm}, with correlation coefficients ranging from 0.17 to 0.33 and 0.13 to 0.26. The correlation gradually increased over time, reaching the maximum 8~10 days ago. On the other hand, the mean e and S_{sum} displayed strong negative correlations with RH_{s10cm}, with correlation coefficients ranging from -0.24 to -0.33 and from -0.15 to -0.33. Their absolute values also gradually increased over time, peaking at the time of 8 days ago and 10 days ago, respectively. Among the soil factors, the mean T_{smax} showed a strong negative correlation with RH_{s10cm}, and its maximum absolute value appeared 4~5 days

ago. Furthermore, via SHAP analysis, it showed that the contributions and impacts of the predictors on the modeling results for Model$_{_soil\&atmo}$ and Model$_{_atmo}$ were different. According to the importance of each predictor, the orders of the top five predictors of these two models were $T_{smax} > P_{sum} > T_{s10cm} > RH_a > T_s$ and $P_{sum} > T_{amax} > RH_a > e > W$, respectively. Overall, among the predictors, the contribution rates of T_{amax}, P_{sum}, and RH_a in atmospheric factors, which functioned as a critical factor affecting the variation in soil moisture, were relatively high in both models.

The overall performances of Model$_{_soil\&atmo}$ and Model$_{_atmo}$ based on XGBoost exhibited lower error values when compared to ANN, RF, and SVM, thereby verifying the prediction capabilities of the XGBoost model. The values of R, RMSE, MAE, MARE, NSE, and ACC for Model$_{_soil\&atmo}$ and Model$_{_atmo}$ based on XGBoost were 0.69, 11.11, 4.87, 0.12, 0.50, and 88%, and 0.66, 11.49, 4.96, 0.14, 0.47, and 86%, respectively. Both Model$_{_soil\&atmo}$ and Model$_{_atmo}$ using XGBoost outperformed the other machine learning models in the scatter distribution of the predicted and measured values. In addition, by integrating the results of SHAP analysis and comparisons of Model$_{_soil\&atmo}$ and Model$_{_atmo}$, it showed that Model$_{_soil\&atmo}$'s prediction effects were always slightly better than those of Model$_{_atmo}$. Hence, it is worth noting that introducing soil factors (e.g., T_{smax}, T_s, and T_{s10cm}) can positively improve the soil moisture prediction accuracy.

Furthermore, the XGBoost model was applicable for provincial-level soil moisture prediction as it captured the spatial distribution characteristics of different levels of drought and effectively predicted the dynamic change process of the "occurrence–development–termination" of a specific drought event. Therefore, the excellent establishment of a soil moisture prediction model based on automatic observation stations, which effectively overcomes the temporary discontinuity of remote sensing inversion and the problem of a low prediction accuracy, could not only effectively guide farmland irrigation but also validly compensate for the insufficient historical observation of soil moisture stations.

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

Funding: This research was funded by the National Natural Science Foundation of China Project (41805049).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The model prediction results presented in this study are available upon request from the corresponding author. The original observations are not publicly available due to the privacy policy.

Acknowledgments: We thank the editors and reviewers for their comments to improve our manuscript.

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

References

1. Ahmad, N.; Malagoli, M.; Wirtz, M.; Hell, R. Drought stress in maize causes differential acclimation responses of glutathione and sulfur metabolism in leaves and roots. *BMC Plant Biol.* **2016**, *16*, 247. [CrossRef] [PubMed]
2. Isabel Ferreira, M.; Valancogne, C. Experimental Study of a Stress Coefficient: Application on a Simple Model for Irrigation Scheduling and Daily Evapotranspiration Estimation. *IFAC Proc. Vol.* **1997**, *30*, 33–38. [CrossRef]
3. Dai, Y.; Zeng, X.; Dickinson, R.E.; Baker, I.; Bonan, G.B.; Bosilovich, M.G.; Denning, A.S.; Dirmeyer, P.A.; Houser, P.R.; Niu, G.; et al. The Common Land Model. *Bull. Am. Meteorol. Soc.* **2003**, *84*, 1013–1024. [CrossRef]
4. Kunstmann, H.; Jung, G.; Wagner, S.; Clottey, H. Integration of atmospheric sciences and hydrology for the development of decision support systems in sustainable water management. *Phys. Chem. Earth Parts A/B/C* **2008**, *33*, 165–174. [CrossRef]
5. Dan, B.; Zheng, X.; Wu, G. Assimilating Shallow Soil Moisture Observations into Land Models with a Water Budget Constraint. *Hydrol. Earth Syst.Sci.* **2020**, *24*, 5187–5201. [CrossRef]
6. Robinson, J.M.; Hubbard, K.G. Soil Water Assessment Model for Several Crops in the High Plains. *Agron. J.* **1990**, *82*, 1141–1148. [CrossRef]

7. Mahmood, R.; Hubbard, K.G. An Analysis of Simulated Long-Term Soil Moisture Data for Three Land Uses under Contrasting Hydroclimatic Conditions in the Northern Great Plains. *J. Hydrometeorol.* **2004**, *5*, 160–179. [CrossRef]
8. Zhang, X.; Ma, Y.H.; Anlauf, R. Forecast and Analysis of Soil Moisture Based on SIMPEL model. *J. Agric. Sci. Technol.* **2013**, *14*, 490–493.
9. Holland, J.E.; Biswas, A. Predicting the mobile water content of vineyard soils in New South Wales, Australia. *Agric. Water Manag.* **2015**, *148*, 34–42. [CrossRef]
10. Hu, W.; Si, B.C. Soil water prediction based on its scale-specific control using multivariate empirical mode decomposition. *Geoderma* **2013**, *193–194*, 180–188. [CrossRef]
11. Prasad, R.; Ravinesh, C.; Li, Y.; Maraseni, T. Weekly soil moisture forecasting with multivariate sequential, ensemble empirical mode decomposition and Boruta-random forest hybridizer algorithm approach. *Catena* **2019**, *177*, 149–166. [CrossRef]
12. Shoaib, M.; Shamseldin, A.Y.; Melville, B.W.; Khan, M.M. A comparison between wavelet based static and dynamic neural network approaches for runoff prediction. *J. Hydrol.* **2016**, *535*, 211–225. [CrossRef]
13. Kamilaris, A.; Francesc, X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
14. Yalcin, H. An Approximation for A Relative Crop Yield Estimate from Field Images Using Deep Learning. In Proceedings of the International Conference on Agro-Geoinformatics (Agro-Geoinformatics), Istanbul, Turkey, 16–19 July 2019.
15. Yu, J.; Tang, S.; Zhangzhong, L.; Zheng, W.; Xu, L. A Deep Learning Approach for Multi-Depth Soil Water Content Prediction in Summer Maize Growth Period. *IEEE Access* **2020**, *8*, 199097–199110. [CrossRef]
16. Fathi, M.T.; Ezziyyani, M.; Ezziyyani, M.; Mamoune, S.E. Crop Yield Prediction Using Deep Learning in Mediterranean Region. In Proceedings of the Advanced Intelligent Systems for Sustainable Development (AI2SD'2019), Marrakech, Morocco, 8–11 July 2019.
17. Ji, R.; Li, X.; Zhang, S.; Zheng, L. Prediction of soil moisture in multiple depth based on time delay neural network. *Trans. Chin. Soc. Agric. Eng.* **2017**, *33*, 132–136.
18. Gill, M.K.; Asefa, T.; Kemblowski, M.W.; McKee, M. Soil moisture predition using support vector machines. *J. Am. Water Resour. Assoc.* **2006**, *42*, 1033–1046. [CrossRef]
19. Pan, J.; Shangguan, W.; Li, L.; Yuan, H.; Zhang, S.; Lu, X.; Wei, N.; Dai, Y. Using data-driven methods to explore the predictability of surface soil moisture with FLUXNET site data. *Hydrol. Process.* **2019**, *33*, 2978–2996. [CrossRef]
20. Tharani, P.P.; Baranidharan, B. An Analysis on Application of Deep Learning Techniques for Precision Agriculture. In Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2–4 September 2021.
21. Gumiere, S.J.; Camporese, M.; Botto, A.; Lafond, J.A.; Paniconi, C.; Gallichand, J.; Rousseau, A.N. Machine Learning vs. Physics-Based Modeling for Real-Time Irrigation Management. *Front. Water* **2020**, *2*, 8. [CrossRef]
22. Li, P.; Zha, Y.; Shi, L.; Tso, C.-H.; Zhang, Y.; Zeng, W. Comparison of the use of a physical-based model with data assimilation and machine learning methods for simulating soil water dynamics. *J. Hydrol.* **2020**, *584*, 124692. [CrossRef]
23. Liu, D.; Liu, C.; Tang, Y.; Gong, C. A GA-BP Neural Network Regression Model for Predicting Soil Moisture in Slope Ecological Protection. *Sustainability* **2022**, *14*, 1386. [CrossRef]
24. Li, Q.; Li, Z.; Shangguan, W.; Wang, X.; Li, L.; Yu, F. Improving soil moisture prediction using a novel encoder-decoder model with residual learning. *Comput. Electron. Agric.* **2022**, *195*, 106816. [CrossRef]
25. Prakash, S.; Sharma, A.; Sahu, S.S. Soil Moisture Prediction Using Machine Learning. In Proceedings of the Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 20–21 April 2018.
26. Adeyemi, O.; Grove, I.; Peets, S.; Domun, Y.; Norton, T. Dynamic Neural Network Modelling of Soil Moisture Content for Predictive Irrigation Scheduling. *Sensors* **2018**, *18*, 3408. [CrossRef] [PubMed]
27. Xu, J.W.; Zhao, J.F.; Zhang, W.C.; Xu, X.X. A Novel Soil Moisture Predicting Method Based on Artificial Neural Network and Xinanjiang Model. *Adv. Mater. Res.* **2010**, *121–122*, 1028–1032. [CrossRef]
28. Li, N.; Zhang, Q.; Yang, F.X.; Deng, Z.L. Research of adaptive genetic neural network algorithm in soil moisture prediction. *Comput. Eng. Appl.* **2018**, *54*, 54–59+69.
29. Notarnicola, C.; Angiulli, M.; Posa, F. Soil moisture retrieval from remotely sensed data: Neural network approach versus Bayesian method. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 547–557. [CrossRef]
30. Wei, W.; Zhang, J.; Zhou, L.; Xie, B.; Zhou, J.; Li, C. Comparative evaluation of drought indices for monitoring drought based on remote sensing data. *Environ. Sci. Pollut. Res.* **2021**, *28*, 20408–20425. [CrossRef]
31. Sandholt, I.; Rasmussen, K.; Andersen, J. A simple interpretation of the surface temperature/vegetation index space for assessment of surface moisture status. *Remote Sens. Environ.* **2002**, *79*, 213–224. [CrossRef]
32. Zheng, W.; Zhangzhong, L.; Zhang, X.; Wang, C.; Zhang, S.; Sun, S.; Niu, H. A Review on the Soil Moisture Prediction Model and Its Application in the Information System. In Proceedings of the Computer and Computing Technologies in Agriculture XI, Jilin, China, 12–15 August 2017.
33. Jiang, A.J.; Peng, H.Y.; Wang, B.M. The analyses of Jiangsu climate variety in forty years. *J. Meteorol. Sci.* **2006**, *26*, 525–529.
34. Qi, Y.; Darilek, J.L.; Huang, B.; Zhao, Y.; Sun, W.; Gu, Z. Evaluating soil quality indices in an agricultural region of Jiangsu Province, China. *Geoderma* **2009**, *149*, 325–334. [CrossRef]
35. Wang, J.Q.; Zhao, Y.F.; Ren, Z.H.; Gao, J. Design and Verification of Quality Control Methods for Automatic Soil Moisture Observation Data in China. *Meteorology* **2018**, *44*, 244–257.

36. Wang, S.; Fu, G. Modelling soil moisture using climate data and normalized difference vegetation index based on nine algorithms in alpine grasslands. *Front. Environ. Sci.* **2023**, *11*, 1130448. [CrossRef]
37. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the ACM, San Francisco, CA, USA, 13–17 August 2016.
38. Bergstra, J.; Bengio, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305.
39. Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Int. Jt. Conf. Artif. Intell.* **1995**, *14*, 1137–1145.
40. Eisenman, R.L. A profit-sharing interpretation of shapley value for n-person games. *Syst. Res. Behav. Sci.* **1967**, *12*, 396–398. [CrossRef]
41. Niazkar, M. Assessment of artificial intelligence models for calculating optimum properties of lined channels. *J. Hydroinform.* **2020**, *22*, 1410–1423. [CrossRef]
42. Agatonovic-Kustrin, S.; Beresford, R. Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *J. Pharm. Biomed. Anal.* **2000**, *22*, 717–727. [CrossRef]
43. Biau, G. Analysis of a random forests model. *J. Mach. Learn. Res.* **2012**, *13*, 1063–1095.
44. Cherkassky, V.; Ma, Y. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Netw.* **2004**, *17*, 113–126. [CrossRef]
45. Matei, O.; Rusu, T.; Petrovan, A.; Mihuţ, G. A Data Mining System for Real Time Soil Moisture Prediction. *Procedia Eng.* **2017**, *181*, 837–844. [CrossRef]
46. Nguyen, T.T.; Ngo, H.H.; Guo, W.; Chang, S.W.; Nguyen, D.D.; Nguyen, C.T.; Zhang, J.; Liang, S.; Bui, X.T.; Hoang, N.B. A low-cost approach for soil moisture prediction using multi-sensor data and machine learning algorithm. *Sci. Total Environ.* **2022**, *833*, 155066. [CrossRef]
47. Filipovi, N.; Brdar, S.; Mimi, G.; Marko, O.; Crnojevi, V. Regional soil moisture prediction system based on long short-term memory network. *Biosyst. Eng.* **2022**, *213*, 30–38. [CrossRef]
48. Li, Q.; Zhu, Y.; Shangguan, W.; Wang, X.; Li, L.; Yu, F. An attention-aware LSTM model for soil moisture and soil temperature prediction. *Geoderma* **2022**, *409*, 115651. [CrossRef]
49. Cai, Y.; Zheng, W.; Zhang, X.; Zhangzhong, L.; Xue, X. Research on soil moisture prediction model based on deep learning. *PLoS ONE* **2019**, *14*, e0214508. [CrossRef] [PubMed]
50. Bell, J.E.; Sherry, R.; Luo, Y. Changes in soil water dynamics due to variation in precipitation and temperature: An ecohydrological analysis in a tallgrass prairie. *Water Resour. Res.* **2010**, *46*, W03523. [CrossRef]
51. Feng, H.; Liu, Y. Combined effects of precipitation and air temperature on soil moisture in different land covers in a humid basin. *J. Hydrol.* **2015**, *531*, 1129–1140. [CrossRef]
52. Ragab, R. Towards a continuous operational system to estimate the root-zone soil moisture from intermittent remotely sensed surface moisture. *J. Hydrol.* **1995**, *173*, 1–25. [CrossRef]
53. Yan, H.; Dechant, C.; Hamid, M. Improving Soil Moisture Profile Prediction with the Particle Filter-Markov Chain Monte Carlo Method. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 6134–6147. [CrossRef]
54. Huang, Y.; Jiang, H.; Wang, W.F.; Wang, W.; Sun, D. Soil moisture content prediction model for tea plantations based on SVM optimised by the bald eagle search algorithm. *Cogn. Comput. Syst.* **2021**, *3*, 351–360. [CrossRef]
55. Wang, X.; Lv, J.; Wang, C.; Xie, D. Soil moisture content prediction using wavelet transform and support vector machine with genetic algorithm optimization. *ICIC Express Lett. Part B Appl.* **2014**, *5*, 1141–1148.

agriculture

MDPI

Article

Design and Experiment of a Visual Detection System for Zanthoxylum-Harvesting Robot Based on Improved YOLOv5 Model

Jinkai Guo [1,2], Xiao Xiao [1,2], Jianchi Miao [1,2], Bingquan Tian [1,2], Jing Zhao [1,2,*] and Yubin Lan [3,*]

[1] School of Agricultural Engineering and Food Science, Shandong University of Technology, Zibo 255000, China
[2] National Sub-Center for International Collaboration Research on Precision Agricultural Aviation Pesticide Spraying Technology, Shandong University of Technology, Zibo 255000, China
[3] College of Electronic Engineering, South China Agricultural University, Guangzhou 510642, China
* Correspondence: zhaojing@sdut.edu.cn (J.Z.); ylan@scau.edu.cn (Y.L.)

Abstract: In order to achieve accurate detection of mature Zanthoxylum in their natural environment, a Zanthoxylum detection network based on the YOLOv5 object detection model was proposed. It addresses the issues of irregular shape and occlusion caused by the growth of Zanthoxylum on trees and the overlapping of Zanthoxylum branches and leaves with the fruits, which affect the accuracy of Zanthoxylum detection. To improve the model's generalization ability, data augmentation was performed using different methods. To enhance the directionality of feature extraction and enable the convolution kernel to be adjusted according to the actual shape of each Zanthoxylum cluster, the coordinate attention module and the deformable convolution module were integrated into the YOLOv5 network. Through ablation experiments, the impacts of the attention mechanism and deformable convolution on the performance of YOLOv5 were compared. Comparisons were made using the Faster R-CNN, SSD, and CenterNet algorithms. A Zanthoxylum harvesting robot vision detection platform was built, and the visual detection system was tested. The experimental results showed that using the improved YOLOv5 model, as compared to the original YOLOv5 network, the average detection accuracy for Zanthoxylum in its natural environment was increased by 4.6% and 6.9% in terms of mAP@0.5 and mAP@0.5:0.95, respectively, showing a significant advantage over other network models. At the same time, on the test set of Zanthoxylum with occlusions, the improved model showed increased mAP@0.5 and mAP@0.5:0.95 by 5.4% and 4.7%, respectively, compared to the original model. The improved model was tested on a mobile picking platform, and the results showed that the model was able to accurately identify mature Zanthoxylum in its natural environment at a detection speed of about 89.3 frames per second. This research provides technical support for the visual detection system of intelligent Zanthoxylum-harvesting robots.

Keywords: YOLOv5; deformable convolution; attention mechanism; visual detection system; Zanthoxylum-harvesting robot

Citation: Guo, J.; Xiao, X.; Miao, J.; Tian, B.; Zhao, J.; Lan, Y. Design and Experiment of a Visual Detection System for Zanthoxylum-Harvesting Robot Based on Improved YOLOv5 Model. *Agriculture* **2023**, *13*, 821. https://doi.org/10.3390/agriculture13040821

Academic Editors: Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 17 February 2023
Revised: 28 March 2023
Accepted: 30 March 2023
Published: 31 March 2023

1. Introduction

Zanthoxylum is widely cultivated in various parts of China, with a cultivated area of about 17.284 million mu and an annual output of over 500,000 tons. It is an important medicinal material and food ingredient. However, manual harvesting of Zanthoxylum is faced with the problems of low efficiency, high temperature and humidity in the work environment, severe mosquito bites, and injuries to workers. Developing a Zanthoxylum-harvesting robot can reduce labor intensity and improve the efficiency of harvesting [1–3].

To enhance the efficiency and quality of Zanthoxylum harvesting, researchers have studied the mechanical harvesting of Zanthoxylum. Wan Fangxin and others designed a comb-type Zanthoxylum harvester using the principles of brushing and air suction [4].

Zheng Tianyun designed an electromagnetic Zanthoxylum picker [5]. During the development of the Zanthoxylum harvester, it was intended to be lightweight and easy to operate, with semi-automated portability [6]. Although these studies have, to some extent, improved the efficiency of Zanthoxylum harvesting and reduced the labor intensity, the lack of an automatic recognition and positioning system for Zanthoxylum makes it difficult to achieve fully automatic harvesting [7,8]. Many scholars have proposed target detection and localization algorithms for Zanthoxylum based on computer vision. Zhang Yongmei and others used a combination of color analysis and image fusion algorithms to identify Zanthoxylum [9]. Yang Ping et al. used the K-means clustering algorithm for target extraction of Zanthoxylum [10]. As deep learning technology advances, the application of object detection algorithms in agriculture is becoming increasingly widespread [11–15]. At the same time, many deep learning-based object detection models have emerged for agricultural harvesting robots. For example, Xie Jiaxing et al. proposed a model called YOLOv5-litchi that detects lychees in natural environments by using an attention mechanism and increasing the small object detection layer, achieving an mAP@0.5 of 87.1% [16]. Zhipeng Cao et al. presented a real-time mango detection model based on YOLOv4 which improved the model detection speed by adjusting the network's width and depth and deleting some convolutional layers; this achieved an mAP@0.5 of 95.12% [17]. Jinhai Wang et al. utilized the Swin Transformer and DETR models to achieve grape bunch detection [18].

Deep learning-based object detection algorithms are mainly divided into two stages: two-stage detection and one-stage detection. Two-stage detection is based on candidate box detection algorithms, and some of the representative algorithms include the R-CNN seriesp [19], SPPNet [20], Fast R-CNN [21], and Faster R-CNN [22]. On the other hand, one-stage detection is simpler compared to two-stage detection as it is based on regression detection algorithms, directly generating the location coordinates and class probability of the target. One-stage detection has a lower training difficulty and a faster detection speed, and the YOLO series [23–26] is a typical representative of one-stage detection algorithms. Currently, the YOLOv5 algorithm is mostly applied for the detection of fruits such as apples [27], cherries [28], and tomatoes [29]; there are fewer studies on the automatic detection of Zanthoxylum.

In summary, the feature extraction-based methods used in previous studies to identify Zanthoxylum place high demand on the dataset, resulting in low detection accuracy when facing complex backgrounds and lighting conditions in natural environments; significant occlusion among Zanthoxylum branches, leaves, and fruits; and irregular shapes of each fruit on the Zanthoxylum spike. To achieve the detection of mature Zanthoxylum and assist the Zanthoxylum-harvesting robot in building a visual detection and positioning system, in this paper, a red-ripe Zanthoxylum image dataset is constructed. To enhance the directed feature extraction, the YOLOv5 model is used, and the CA (coordinate attention) mechanism is introduced to weaken the feature extraction of complex backgrounds. To specifically solve the problems of irregular Zanthoxylum spike shapes, complex field backgrounds, and dense Zanthoxylum fruits, the deformable convolution is introduced to improve the accuracy of mature Zanthoxylum recognition under natural conditions. In the second year, we deployed the model onto the platform constructed for the Zanthoxylum-harvesting robot for field tests in order to evaluate its effectiveness and generalizability.

2. Materials and Methods

2.1. Mature Zanthoxylum Image Collection

The images of mature Zanthoxylum were captured in Zijing Village, Shima Town, Boshan District, Zibo City, Shandong Province, from the Zanthoxylum plantations of local farmers. The images were collected on 25 August 2021 using a DJI motion camera, a Sony 5T camera, and a mobile phone in the natural environment.

The resolution of each collected image was 1280 × 1024 (pixels) with a 4:3 aspect ratio, and the original images were saved in JPG format. The images should include cases with single mother trees, multiple mother trees, Zanthoxylum branches, Zanthoxylum leaves,

and Zanthoxylum alone. In the natural environment, Zanthoxylum often has shading and backlighting, so when taking pictures with a camera, cases with shading and backlighting should be included as often as possible.

The growth and distribution of ripe Zanthoxylum on the Zanthoxylum tree in its natural environment are illustrated in Figure 1,. It can be seen from the figure that under natural conditions, the growth direction and the number of fruits in each cluster of Zanthoxylum are often not regular. The cluster of Zanthoxylum is a discrete target with an irregular shape. Under natural lighting conditions, shading and back-light are inevitable, making the color characteristics of Zanthoxylum unreliable. The overlap of branches and leaves in Zanthoxylum also results in an incomplete shape of the collected Zanthoxylum.

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

Figure 1. Images of Zanthoxylum peppercorn fruit under natural conditions. (**a**) Occlusion situation; (**b**) multi-mother plant overlap; (**c**) backlit conditions; (**d**) shading conditions.

2.2. Construction of the Dataset

To construct a deep learning model for the effective detection of Zanthoxylum under natural environmental conditions, this study only screened and removed images that were excessively blurred due to the shooting equipment not being completely focused. A total of 2827 images were collected at the trial site, and after screening and removal, 2368 images remained. The images were randomly divided into a training set, a validation set, and a test set in a 7:1:2 ratio. The dataset was annotated using the LabelImg annotation tool. The mature Zanthoxylum plants were selected by using a mouse to create a rectangular bounding box around the outer edge of the target contour, forming a quadrilateral bounding box. This study annotated the irregularly shaped Zanthoxylum clusters, with no requirement for the size of the quadrilateral. The area of the quadrilateral bounding box was kept as close as possible to the area of the Zanthoxylum it contained. The blue rectangle shows the ripe prickly ash fruit. An example of the annotated sample is shown in Figure 2.

Figure 2. Sample image annotation.

To enhance the robustness of the object detection model while taking into account the tilt angle, illumination intensity, and different resolutions that may exist in the image acquisition process of field equipment, the original images used for modeling were augmented using methods such as geometric transformation, color transformation, and mixed

transformation. Ultimately, 12,000 images were obtained. The data augmentation examples are shown in Figure 3.

Figure 3. Data enhancement sample.

2.3. Network Model Construction

2.3.1. YOLOv5

YOLO is a representative of one-stage object detection algorithms which views object detection as a regression problem and performs feature extraction, object classification, and boundary box regression in a deep neural network, realizing end-to-end inference. It has a fast detection speed and can detect and classify objects in an image simultaneously.

The Zanthoxylum detection method, based on deep learning YOLO, can locate Zanthoxylum using real-time video and return its coordinates, category, and confidence. In the YOLO neural network, the input data are represented by an image, which is divided into $S \times S$ grids. When the center of the Zanthoxylum falls into a grid, the grid will detect it. Each grid detects B targets, and each target will receive 5 prediction parameters: x, y, w, h, and confidence, where (x, y) represent the target's coordinates and (w, h) represent the width and height of the boundary box.

YOLOv5 has four main parts in its network structure: the input end, backbone network, neck network, and output end. The input end represents the input image and includes some image preprocessing, including resizing the input image to the input size of the network and normalizing it. The backbone network of YOLOv5 uses Focus [30] as the benchmark network, which mainly uses slicing operations to crop the input image. In the neck portion, YOLOv5 adopts the fast spatial pyramid pooling [31] (SPPF) module for multi-scale feature fusion, as well as the feature pyramid network (FPN) [32] and the path aggregation network PAN [33] modules for network feature fusion and strengthening. The output end is used to output the object detection results.

2.3.2. A YOLOv5 Model Incorporating Attention Mechanisms and Deformable Convolutions

The introduction of attention mechanisms in deep learning networks can enhance the interested target region. Deformable convolution kernels can be adjusted based on the actual size and shape of the detected target, thus more effectively extracting the features of the detected object. As the shapes and sizes of mature chili pepper fruit spikes are irregular, to improve the detection accuracy of mature chili pepper fruit, this paper incorporates the attention mechanism module and the deformable convolution module into the YOLOv5 network.

Coordinate attention (CA) [34] is a kind of attention mechanism proposed by Qibin Hou et al. in 2021. The mechanism embeds position information into channel attention. The module decomposes channel attention into a 1D feature-decoding process in which features

are aggregated along different directions. During this process, the long-range features are extracted along one spatial direction, and precise position information is retained along the other spatial direction. The resulting feature maps are then encoded and aggregated to produce position- and direction-sensitive feature maps, thus enhancing the interest target area [35]. The specific structure of the CA attention mechanism module is shown in Figure 4.

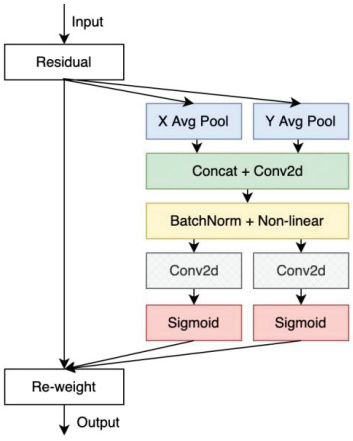
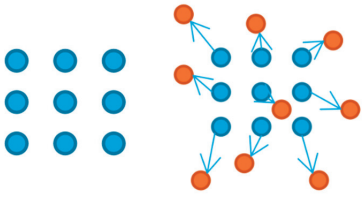


Figure 4. Structure diagram of the coordinate attention mechanism.

The detection of mature Zanthoxylum is highly correlated with the region of the fruit in the image. The model's sensitivity to the position of mature Zanthoxylum helps to improve detection accuracy.

In this paper, the model was required to be deployed on mobile devices. The early stage of the network is focused on shallow features, and adding the CA module during this stage would decrease the training and detection speed due to the high number of features considered in this stage. Therefore, the CA module was added before the SPPF module.

Deformable convolutional networks (DCNs) [36] are novel convolutional methods introduced by Dai et al. in 2017. The deformable convolution adds a direction offset to each element of the convolutional kernel, allowing the kernel to adjust its shape according to the actual object being detected and to better extract the input features. This type of convolution captures local features more effectively, especially when the object shape changes, making the advantages of deformable convolution more apparent. Figure 5 shows a comparison between a conventional 3×3 convolutional kernel and deformable convolution.



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

Figure 5. Comparison of conventional 3×3 convolution and deformable convolution. (**a**) Normal convolution; (**b**) deformable convolution.

DCN can improve the model's ability to extract features from objects with deformation. The offset is learned by parallel convolutional layers, and the kernel can be shifted at the

sampling points on the input feature map. This causes the model to focus on the target area for detection and on making the kernel shape more suitable for the target shape, rather than being limited to a square sampling area. However, the first generation of deformable convolution may extend beyond the target area of interest and cause performance degradation, so Deformable ConvNets v2 (DCNv2) [37,38] introduced the addition of weight to each sampling point while learning the offset. This not only enhanced the acceptance of the input feature position, but also regulated the amplitude of the input feature, thus increasing the model's ability to model and learn. The following is the operation flow of the DCNv2 module.

Initially, if a 3×3 convolutional kernel is adopted, the definition of the kernel is R, and the size of the kernel is two-dimensional, as shown in Formula (1).

$$R = \{(-1,1),(0,1),\ldots,(0,1),(1,1)\} \tag{1}$$

DCNv2 first extracts the feature map using conventional convolution kernels, and then takes the obtained feature map as input, applying another convolution layer to the feature map to obtain the deformable convolution offset. The calculation formula of the normal convolution operation's output feature map is shown in Formula (2).

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n) \tag{2}$$

In the formula, p_0 is the center point of the conventional convolution kernel; p_n is the sampling point of the conventional convolution kernel; x is the input feature map; and y is the output feature map.

The formula for calculating the output feature map using a deformable convolution kernel is shown in Formula (3).

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n + \Delta p_n) \Delta m_n \tag{3}$$

In this formula, Δp_n represents the adjusted offset, Δm_n represents the weight coefficient, and the remaining variables are the same as those in the traditional convolution operation. The deformable convolution introduces the position offset of the sampling points on the basis of the traditional convolution, which enables the output feature map to better represent the features of irregular targets. The offset Δp_n shifts the points in region R based on the distribution of target features, and since the offset is generated by convolving the input feature map with another convolution layer, it is usually represented by a decimal. Therefore, by performing bilinear interpolation on the offset, the formula of the deformable convolution is transformed into Formula (4).

$$X(p) = \sum_q G(q,p) \cdot x(q) \tag{4}$$

In Equation (4), q represents the position of the sample point after being offset, p represents the integer grid point, and G (q, p) represents the integer form of the sample point position obtained from the bilinear interpolation operation. The structure diagram of deformable convolution is shown in Figure 6.

The model proposed in this paper needs to be deployed on mobile devices, so the YOLOv5s model with the smallest number of parameters was selected for improvement. Figure 7 shows the structure of the improved YOLOv5 network model. The CA attention mechanism module was inserted before the SPPF module of the backbone network in this model, and the deformable convolution module was introduced into the neck of the model.

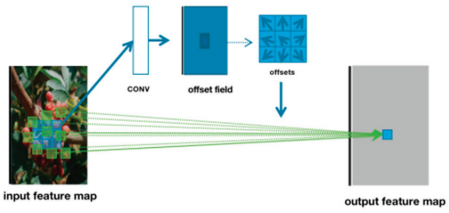


Figure 6. Deformable convolutional structure.

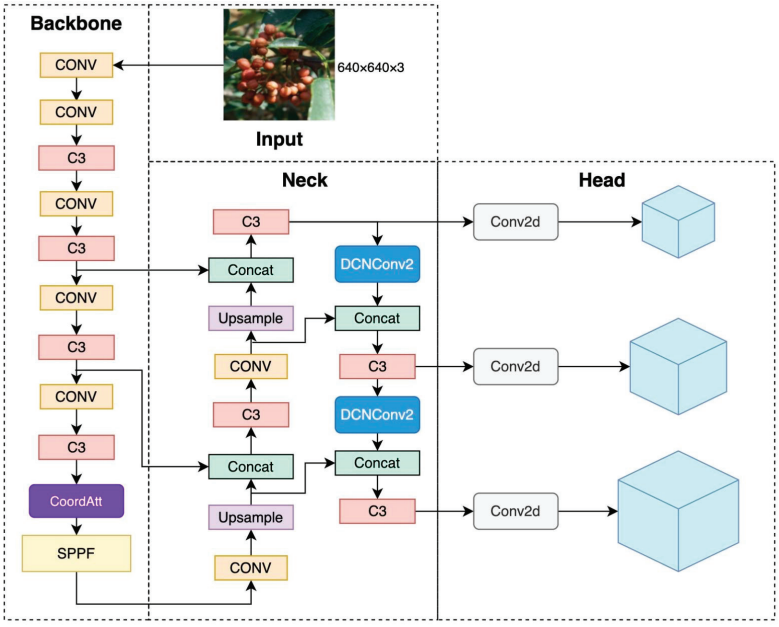


Figure 7. Improved YOLOv5 network map. Conv represents convolution, C3 represents a module consisting of three Convs and multiple bottleneck layers, SPPF refers to a spatial pyramid pooling-fast structure, Concat represents a feature fusion method of channel connection, Upsample represents up-sampling, and DCNConv2 refers to the deformable convolution module.

3. Experimental Design

3.1. Model Training

3.1.1. Model Training Parameters

The platform for training and testing the model in this paper was a workstation computer with an Intel Core (TM) i9-9820X processor, operating at a frequency of 3.3 GHz, with 32 GB running memory and a GeForce GTX 2080ti GPU with 11 GB of memory. The operating environment was Ubuntu18.04 LTS. The training and testing of the model were based on the Pytorch framework, using the Python programming language and libraries such as CUDA, Cudnn, and OPENCV for setup.

The input image size for model training was 640 pixels by 640 pixels, and the model was trained for a total of 200 epochs. In order to evaluate the performance of the model, the weights parameters were saved after each epoch. The learning rate was warmed up using a warm-up method, and during this stage, the learning rate was updated through linear interpolation followed by the use of the cosine annealing algorithm.

The loss function is a value that represents the level of agreement between the model's prediction and the truth. Its magnitude determines the performance of the model. During

the model training process, factors that affect the training accuracy include the box loss (box_loss), object confidence loss (obj_loss), and classification loss (cls_loss). The loss function for the model in this paper is defined as shown in Equation (5).

$$\text{Loss} = 0.3 \times \text{box\_loss} + 0.4 \times \text{obj\_loss} + 0.3 \times \text{cls\_loss} \tag{5}$$

The change curve of the model's loss value during the training process is shown in Figure 8. From the figure, it can be seen that the loss value rapidly decreased in the first 15 rounds of training. After 100 rounds of training, the loss value was basically stable; the loss of the training set and the validation set have converged, and the gap between them is very small. The model did not show overfitting.

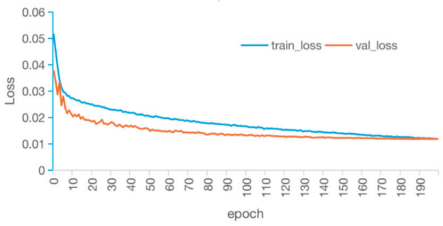


Figure 8. Change curve of loss value.

3.1.2. Model Evaluation Index

In this study, we primarily evaluated the performance of the output model using precision (P), recall (R), F1 score, mean average precision (mAP), and frame per second (FPS). The most intuitive metric for measuring the model's detection and classification ability is mAP, and the higher the accuracy of the model, the higher the mAP value. Therefore, we use the size of the mAP value as the primary evaluation criterion for the model. The intersection over union (IOU) is the ratio of the intersection and union of the generated candidate box and the original annotated box. mAP@0.5 indicates that the average precision mean is calculated when the IOU threshold is set to 0.5, and mAP@0.5:0.95 indicates the average value of mAP at different IOU thresholds (from 0.5 to 0.95 with a step of 0.05). The Zanthoxylum detection algorithm proposed in this paper is intended for use in the visual detection system of an intelligent Zanthoxylum-picking robot, which requires certain accuracy in terms of locating mature Zanthoxylum, so we evaluated the model using both mAP@0.5 and mAP@0.5:0.95.

Precision, denoted as P, is the ratio of the number of accurately predicted samples to the total number of samples, and its formula is as follows:

$$\text{P} = \frac{\text{T}_\text{P}}{\text{T}_\text{P} + \text{F}_\text{P}} \times 100\% \tag{6}$$

T_P is the number of positive samples that were correctly predicted as positive, and F_P is the number of negative samples that were wrongly predicted as positive. R is the proportion of all positive samples that were correctly predicted as positive, and is calculated as follows:

$$\text{R} = \frac{\text{T}_\text{P}}{\text{T}_\text{P} + \text{F}_\text{N}} \times 100\% \tag{7}$$

where F_N is the number of positive samples that were wrongly predicted as negative.

F1 score is a metric that balances precision and recall, and is calculated as the harmonic mean of precision and recall. The formula is as follows:

$$\text{F1} = 2 \times \frac{\text{P} \times \text{R}}{(\text{P} + \text{R})} \tag{8}$$

The average precision (AP) is the area under the P-R curve, with recall R as the *X*-axis and precision P as the *Y*-axis. The mean average precision (mAP) is the average of the AP values for each category, obtained by summing up the AP values of each category and dividing by the total number of categories. The formula for this calculation is as follows:

$$AP = \int_0^1 P(r)dr \tag{9}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{10}$$

where P(r) represents the expression of the function for the P-R curve, N denotes the number of categories, and AP_i represents the average precision value for category i.

3.2. Test Platform Construction

The Zanthoxylum detection algorithm proposed in this paper was intended to be applied to the visual system of a smart Zanthoxylum-picking robot. To test the practicality and problems of the algorithm, a Zanthoxylum-picking robot platform was set up as shown in Figure 9.

Figure 9. Construction of test platform.

The platform consisted of a tracked chassis and a six-axis robotic arm. The trained detection model was embedded within an industrial control computer and mounted on a D435i depth camera for image acquisition and the detection of ripe Zanthoxylum on trees. The software and hardware parameters of the industrial control computer are listed in Table 1.

Table 1. Hardware and software parameters of industrial computer.

Name	Parameter
CPU	I7-1165G7
Memory	16GB
GPU	RTX2060-6GB
System	Ubuntu18.04
Python version	3.8.13
Pytorch version	1.12.0

To verify the effectiveness and generalizability of the trained model, it was deployed on the experimental platform, and a field test for Zanthoxylum detection was conducted in Zijing Village, Shima Town, Boshan District, Zibo City, Shandong Province, on 29 September 2022. The performance of the Zanthoxylum detection model was tested in real-world scenarios, as shown in Figure 10.

Figure 10. Peppercorn-picking robot test platform.

4. Results

4.1. Comparative Analysis of Algorithm Optimization Experiment and Results

Table 2 lists the different models used in this study, along with their corresponding descriptions. The models were trained and tested using the same dataset.

Table 2. Model name and comparison description.

Number	Models	Explain
1	YOLOv5	YOLOv5s
2	All-DCNv2-YOLOv5	The convolutional layers in the backbone network are all replaced with deformable convolutions
3	CA-YOLOv5	The CA module is added to the backbone network
4	DCNv2-YOLOv5	The neck network introduced DCNv2
5	CA-DCNv2-YOLOv5	The backbone network adds the CA, and the Neck network introduces the DCNv2
6	Faster R-CNN	A typical two-stage detection algorithm
7	SSD [39]	A typical one-stage detection algorithm
8	CenterNet [40]	A typical one-stage detection algorithm

4.1.1. Ablation Study

In order to demonstrate the effectiveness of the proposed CA-DCNv2-YOLOv5 model, an ablation study was designed to verify the impact of different usage methods of the CA and DCNv2 on the model's performance in terms of detecting ripe Zanthoxylum.

A comparison of the changes in accuracy, recall, mAP@0.5, and mAP@0.5:0.95 during the 200-round training processes of different improved YOLOv5 algorithms is shown in Figure 11.

As shown in Figure 11, it can be seen that simply replacing the conventional convolution in the backbone network with deformable convolution modules had a limited ability to improve the model's performance, and caused relatively severe oscillation in the first half of the model training. Additionally, as each deformable convolution module required separate calculation of the offset, the computation of the model was increased, leading to an increase in both the training and detection times of the model. The results of the ablation comparison experiment are shown in Table 3.

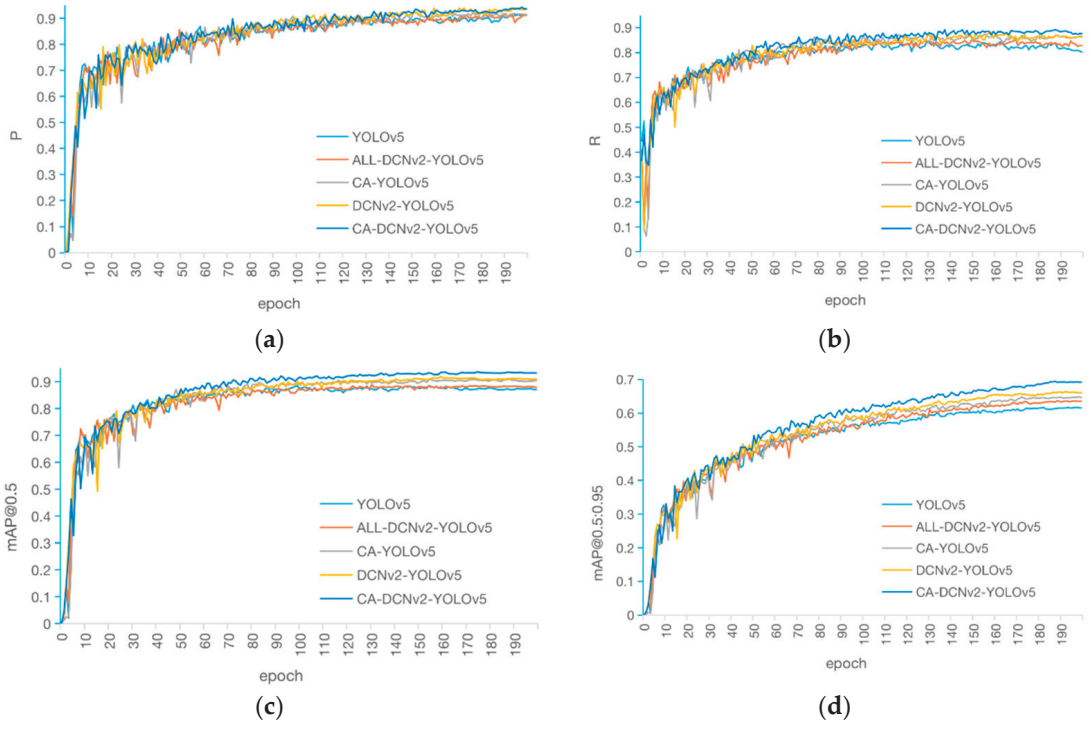


Figure 11. Evaluation index changes of different YOLO algorithms in the training process. (**a**) The precision change curve; (**b**) the recall change curve; (**c**) the change curve of mAP@0.5; (**d**) the change curve of mAP@0.5:0.95.

Table 3. Ablation comparison experiment results.

Number	F1 Score	mAP@0.5/%	mAP@0.5:0.95/%	Speed/(Frame/s)	Model Size/(M)
1	0.86	88.9	62.6	97.1	14.5
2	0.87	88.3	63.6	82.7	14.7
3	0.89	90.4	64.8	97.1	14.8
4	0.90	91.1	66.4	95.2	14.6
5	0.91	93.5	69.5	95.2	14.7

Table 3 shows the results of the ablation study. The CA-YOLOv5 model, which incorporated the CA attention mechanism module, improved the mAP@0.5 and mAP@0.5:0.95 by 1.5% and 2.2%, respectively, compared to the original model. The DCNv2-YOLOv5 model, which introduced the deformable convolution modules, improved the mAP@0.5 by 2.2%, and the mAP@0.5:0.95 by 3.8%. However, simply replacing the conventional convolution modules in the backbone network with deformable convolution modules resulted in limited improvement to the model's accuracy and a substantial decrease in detection speed. The improved YOLO V5 model, which combined both improvements, further enhanced the detection accuracy of Zanthoxylum, with a 2.9% improvement in mAP@0.5 and a 2.9% improvement in mAP@0.5:0.95 compared to the CA-YOLOv5 model and the DCNv2-YOLOv5 model, respectively. Compared to the original YOLOv5 object detection model, the detection speed remained largely unchanged, with 4.6% and 6.9% improvements in mAP@0.5 and mAP@0.5:0.95, respectively.

From the above experiments, it can be seen that the introduction of the CA attention mechanism module and the proper replacement of the deformable convolution module

can effectively improve the target detection accuracy for mature Zanthoxylum. However, both combined in the CA-DCNv2-YOLOv5 model resulted in the best mAP@0.5 and mAP@0.5:0.95.

4.1.2. Comparison of Different Models

In addition to the ablation study, typical two-stage object detection algorithms, e.g., Faster R-CNN, and typical one-stage object detection algorithms, e.g., SSD and CenterNet, were trained using the dataset in this paper and tested with the same test set. The results are shown in Table 4.

Table 4. Performance comparison of different models.

Model	F1 Score	mAP@0.5/%	mAP@0.5:0.95/%	Speed/(Frame/s)	Model Size/(M)
CA-DCNv2-YOLOv5	0.91	93.5	69.5	95.2	14.7
Faster R-CNN	0.85	85.9	55.9	16.0	113.4
SSD	0.76	79.6	40.6	98.9	95.5
CenterNet	0.69	77.8	38.5	81.1	131

The F1 score of the CA-DCNv2-YOLOv5 model was 0.91, with a mAP@0.5 of 93.5% and a mAP@0.5:0.95 of 69.5%, outperforming other network models. In terms of model size, that of the CA-DCNv2-YOLOv5 model was 14.7 M. On the other hand, that of the Faster R-CNN model was 113.4 M, which was close to 8 times the size of the CA-DCNv2-YOLOv5 model, and its detection speed was only one-sixth that of the CA-DCNv2-YOLOv5 model.

Comparing the four object detection algorithms, it can be seen that the CA-DCNv2-YOLOv5 model had the smallest model size, the highest F1 score, the highest mAP and the fastest detection speed. It achieved the best performance in detecting mature Zanthoxylum in natural environments.

4.1.3. Comparison of Model Detection Effect

A comparison of the different models' partial detection results can be seen in Figure 12. It can be seen that, except for Network 5, there were different degrees of false negatives, false positives, and repeated box selections in other models. However, the proposed Model 5 did not have these problems, and still showed good detection results for Zanthoxylum that was severely occluded by leaves. In addition, the size and position of the detection box were more accurate.

4.1.4. Network Attention Visualization

To demonstrate CA-DCNv2-YOLOv5's feature extraction capabilities more intuitively, this paper visualizes a feature heat map [41]. The results of model feature visualization are shown in Figure 13, where red areas indicate the regions on which the network is highly focused, with deeper colors indicating stronger levels of attention.

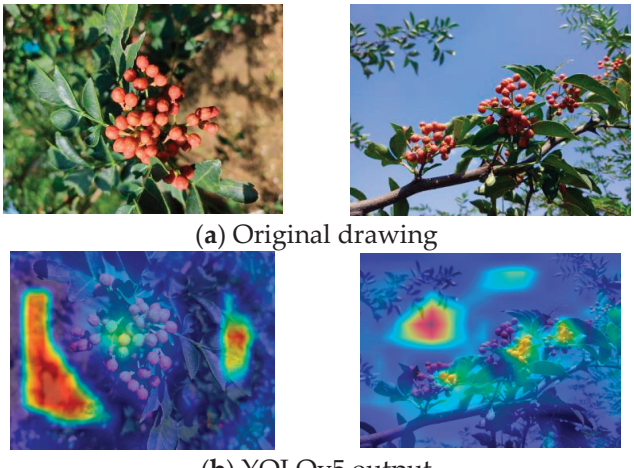
(**a**) YOLOv5 instance detection (**b**) ALL-DCNv2-YOLOv5 instance detection

(**c**) YOLOv5 instance detection (**d**) ALL-DCNv2-YOLOv5 instance detection

(**e**) YOLOv5 instance detection (**f**) ALL-DCNv2-YOLOv5 instance detection

(**g**) YOLOv5 instance detection (**h**) ALL-DCNv2-YOLOv5 instance detection

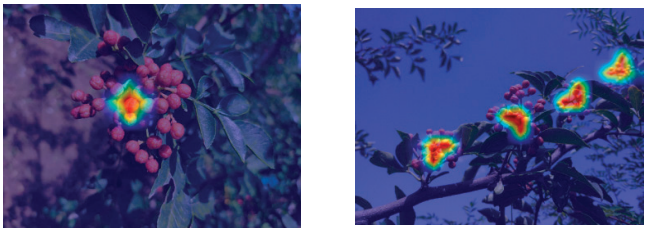
Figure 12. Comparison of partial test results of different models. (**a**) YOLOv5 instance detection; (**b**) ALL-DCNv2-YOLOv5 instance detection; (**c**) CA-YOLOv5 instance detection; (**d**) DCNv2-YOLOv5 instance detection; (**e**) CA-DCNv2-YOLOv5 instance detection; (**f**) Fsater R-CNN instance detection; (**g**) m SSD instance detection; (**h**) Centernet instance detection.



(**a**) Original drawing

(**b**) YOLOv5 output

Figure 13. *Cont*.



(**c**) CA-DCNv2-YOLOv5 Output

Figure 13. Class activation mapping.

It can be seen from the figure that compared to the YOLOv5 model, the CA-DCNv2-YOLOv5 model pays more attention to the local areas of the Zanthoxylum in the feature extraction process, and relatively less attention to irrelevant information. Thus, it showed higher accuracy in detecting mature Zanthoxylum.

4.1.5. Model Recognition Performance for Zanthoxylum under Occlusion Conditions

To further verify the detection performance of the improved CA-DCNv2-YOLOv5 model on occluded Zanthoxylum, a separate test set was constructed by manually selecting 100 images with occlusions from the test set, and both YOLOv5 model and CA-DCNv2-YOLOv5 model were used for prediction. The prediction results are shown in Table 5.

Table 5. Comparison of the detection effect of the occluded target fruit before and after the improvement.

Models	mAP@0.5/%	mAP@0.5:0.95/%
YOLOv5	86.5	58.9
CA-DCNv2-YOLOv5	91.9	63.6

As shown in Table 5, the CA-DCNv2-YOLOv5 model outperformed the YOLOv5 model in the test set with occlusions, with improvements of 5.4 and 4.7 percentage points in mAP@0.5 and mAP@0.5:0.95, respectively. These results demonstrate that the improved YOLOv5 target detection algorithm proposed in this paper helps to increase the detection accuracy of mature Zanthoxylum with occlusions.

4.2. Field Experiment

The feasibility and practicality of the proposed Zanthoxylum detection algorithm were demonstrated through the collection, recognition, and location of Zanthoxylum images. These images were obtained from different positions on trees in their natural environment by means of a mechanical arm in different poses and the improved YOLOv5 model. The results of the actual performance tests of the model are shown in Figure 14. The model was able to detect and recognize mature Zanthoxylum in the field and output the coordinate information, with an average detection time of 11.2 ms and a detection speed of 89.3 frames per second, thus satisfying the real-time detection requirements. The recognition and detection information can be used in real time to drive the Zanthoxylum-harvesting robot to perform the cutting, grasping, and collection tasks. The harvesting performance of the robot is shown in Figure 15.

Figure 14. Detection effect of ripe prickly ash fruit in the field.

Figure 15. Prickly ash-picking robot work diagram.

5. Discussion

Zanthoxylum features a growth pattern in which the fruits are discrete, with an irregular spike shape. This makes the detection of the fruit challenging due to cross-occlusion between fruits. To address this challenge, this paper introduces a deformable convolutional module to better adapt to the shape of the Zanthoxylum and extract more features. At the start of the network, the feature map has a large number of features. Adding the deformable convolutional module at this point will cause the model to learn a significant number of irrelevant features and, thus, considerably reduce both the training speed and the detection speed. The proposed model is deployed on a mobile device, and has a certain requirement for detection speed. The experimental results showed that introducing the deformable convolutional module into the neck of the model instead into the backbone network significantly improves the detection speed. Furthermore, the introduction of a CA attention mechanism into the backbone network increased the model's sensitivity to positional information. Combining this with the deformable convolution improves the accuracy of the model in detecting mature Zanthoxylum.

Zanthoxylum undergoes a red maturation process that takes place over a period of time, approximately two months, during August and September. During this time, all Zanthoxylum is in the red maturation stage. Images of red mature Zanthoxylum collected during different months are shown in Figure 16. Figure 16a shows red mature Zanthoxylum from August, with more plump and fresh red fruit. Figure 16b shows mature red Zanthoxylum from September, the fruits of which are relatively dry and have a deep red color. In subsequent algorithmic improvements, the changes in color and fruit shape during the red maturation process of Zanthoxylum should be fully considered in order to further improve the robustness and detection accuracy of the algorithm.

(**a**) Red ripe pepper fruit in August (**b**) September red ripe prickly ash fruit

Figure 16. Comparison of red ripe prickly ash at different growth stages.

6. Conclusions

The visual detection system is a key module for the Zanthoxylum-harvesting robot. In order to achieve accurate detection of mature Zanthoxylum this paper presents an improved YOLO algorithm to detect Zanthoxylum in natural environments. The main conclusions are as follows:

1. An improved YOLOv5 model was proposed for Zanthoxylum cluster detection in its natural environment by adding the CA attention mechanism module into the backbone network and introducing the deformable convolutional module into the neck. The testing results showed that the improved model had an average accuracy of 93.5% in mAP@0.5 and 69.5% in mAP@0.5:0.95, which improved by 4.6% and 6.9%, respectively, compared to the original YOLOv5 model, while maintaining the basic detection speed. In addition, the CA-DCNv2-YOLOv5 model proposed in this paper demonstrated a significant performance advantage compared to Faster R-CNN, SSD, and CenterNet.
2. The improved YOLOv5 network model was tested on an image dataset that included occlusions of Zanthoxylum. The average precision scores, mAP@0.5 and mAP@0.5:0.95, improved by 5.4% and 4.7%, respectively, compared to the original YOLOv5 network.
3. The improved YOLOv5 network model had a detection speed of approximately 89.3 frames per second on mobile devices, meeting the real-time detection requirements of the Zanthoxylum-harvesting robot.

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

Funding: This study was funded by the "One Case, one Discussion" special fund for the introduction of top talents in Shandong Province (Lu Zhengban Zi [2018] No.27), supported by the Natural Science Foundation of Shandong Province (ZR2021MD091).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Xu, K.; Jun, W.; Shang, J.; Li, Y.; Cao, J. Development status and countermeasures of Dahongpao pepper industry in Shantin district, Zaozhuang city. *Agric. Dev. Equip.* **2021**, *6*, 47–48.
2. An, J.; Yang, H.; Lu, W.; Wang, X.; Wang, W. The current situation and development trend of pepper harvesting machinery. *Agric. Sci. Technol. Inf.* **2019**, *6*, 57–59.

3. Li, K. Research and Design of Pepper Harvesting Robot Control System. Master's Thesis, Lanzhou University of Technology, Lanzhou, China, 2020.
4. Wan, F.-X.; Sun, H.-B.; Pu, J.; Li, S.-Y.; Zhao, Y.-B.; Huang, X.-P. Design and experiment of comb-air pepper harvester. *Agric. Res. Arid. Areas* **2021**, *39*, 219–227+238.
5. Zheng, T.-Y. Design of electromagnetic pepper harvester. *Electr. Autom.* **2017**, *39*, 108–110.
6. Qi, R.L. Research on Pepper Target Recognition and Positioning Technology Based on Machine Vision. Master's Thesis, Shaanxi University of Technology, Hanzhong, China, 2020. [CrossRef]
7. Zhang, J. Target extraction of fruit picking robot vision system. *J. Phys. Conf. Ser.* **2019**, *1423*, 012061. [CrossRef]
8. Tang, S.; Zhao, D.; Jia, W.; Chen, Y.; Ji, W.; Ruan, C. Feature extraction and recognition based on machine vision application in lotus picking robot. In Proceedings of the International Conference on Computer & Computing Technologies in Agriculture, Beijing, China, 27–30 September 2015.
9. Zhang, Y.M.; Li, J.X. Study on automatic recognition technology of mature pepper fruit. *Agric. Technol. Equip.* **2019**, *1*, 4–6.
10. Yang, P.; Guo, Z.C. Vision recognition and location solution of pepper harvesting robot. *J. Hebei Agric. Univ.* **2020**, *43*, 121–129. [CrossRef]
11. Bai, Q.; Gao, R.; Zhao, C.; Li, Q.; Wang, R.; Li, S. Multi-scale behavior recognition method of cow based on improved YOLOv5s network. *J. Agric. Eng.* **2022**, *38*, 163–172.
12. Hao, J.; Bing, Z.; Yang, S.; Yang, J.; Sun, L. Detection of green walnut with improved YOLOv3 algorithm. *J. Agric. Eng.* **2022**, *38*, 183–190.
13. Cong, P.; Feng, H.; Lv, K.; Zhou, J.; Li, S. MYOLO: A Lightweight Fresh Shiitake Mushroom Detection Model Based on YOLOv3. *Agriculture* **2023**, *13*, 392. [CrossRef]
14. Xu, D.; Zhao, H.; Lawal, O.M.; Lu, X.; Ren, R.; Zhang, S. An Automatic Jujube Fruit Detection and Ripeness Inspection Method in the Natural Environment. *Agronomy* **2023**, *13*, 451. [CrossRef]
15. Phan, Q.-H.; Nguyen, V.-T.; Lien, C.-H.; Duong, T.-P.; Hou, M.T.-K.; Le, N.-B. Classification of Tomato Fruit Using Yolov5 and Convolutional Neural Network Models. *Plants* **2023**, *12*, 790. [CrossRef] [PubMed]
16. Xie, J.; Peng, J.; Wang, J.; Chen, B.; Jing, T.; Sun, D.; Gao, P.; Wang, W.; Lu, J.; Yetan, R.; et al. Litchi Detection in a Complex Natural Environment Using the YOLOv5-Litchi Model. *Agronomy* **2022**, *12*, 3054. [CrossRef]
17. Cao, Z.; Yuan, R. Real-Time Detection of Mango Based on Improved YOLOv4. *Electronics* **2022**, *11*, 3853. [CrossRef]
18. Wang, J.; Zhang, Z.; Luo, L.; Zhu, W.; Chen, J.; Wang, W. SwinGD: A Robust Grape Bunch Detection Model Based on Swin Transformer in Complex Vineyard Environment. *Horticulturae* **2021**, *7*, 492. [CrossRef]
19. 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; pp. 580–587.
20. 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]
21. Girshick, R.B. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
22. 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]
23. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
24. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
25. Redmon, J.; Farhadi, A. YOLOv3: An incremental improvement [DB/OL]. *arXiv* **2018**, arXiv:1804.02767v1.
26. Bochkovskiy, A.; Wang, C.; Liao, H. YOLOv4: Optimal speed and accuracy of object detection [DB/OL]. *arXiv* **2020**, arXiv:2004.10934v1.
27. Yan, B.; Fan, P.; Wang, M.; Shi, S.; Lei, X.; Yang, F. Real-time recognition of apple picking methods based on improved YOLOv5m for harvesting robots. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53*, 28–38+59.
28. Zhang, Z.; Luo, M.; Guo, S.; Liu, G.; Li, S. Cherry fruit detection method in natural scene based on improved YOLOv5. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53* (Suppl. 1), 232–240.
29. He, B.; Zhang, Y.; Gong, J.; Fu, G.; Zhao, Y. Fast recognition of night greenhouse tomato fruit based on improved YOLO v5. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53*, 201–208.
30. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9627–9636.
31. Huang, Z.; Wang, J.; Fu, X.; Yu, T.; Guo, Y.; Wang, R. DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection. *Inf. Sci.* **2020**, *522*, 241–258. [CrossRef]
32. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
33. 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 City, UT, USA, 18–22 June 2018; pp. 8759–8768.

34. 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.
35. Zha, M.; Qian, W.; Yi, W.; Hua, J. A lightweight YOLOv4-Based forestry pest detection method using coordinate attention and feature fusion. *Entropy* **2021**, *23*, 1587. [CrossRef]
36. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. *Comput. Vis. Pattern Recognit.* **2017**, *9*, 334–420.
37. Zhu, X.; Hu, H.; Lin, S.; Dai, J. Deformable convnets v2: More deformable, better results. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 9308–9316.
38. Park, H.; Paik, J. Pyramid attention upsampling module for object detection. *IEEE Access* **2022**, *10*, 38742–38749. [CrossRef]
39. 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.
40. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6569–6578.
41. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.

agriculture

MDPI

Article

Multi-Modal Late Fusion Rice Seed Variety Classification Based on an Improved Voting Method

Xinyi He [1], Qiyang Cai [1], Xiuguo Zou [1], Hua Li [2], Xuebin Feng [2], Wenqing Yin [2] and Yan Qian [1,*]

[1] College of Artificial Intelligence, Nanjing Agricultural University, Nanjing 210031, China
[2] College of Engineering, Nanjing Agriculture University, Nanjing 210031, China
* Correspondence: qianyan@njau.edu.cn; Tel.: +86-25-5860-6585

Abstract: Rice seed variety purity, an important index for measuring rice seed quality, has a great impact on the germination rate, yield, and quality of the final agricultural products. To classify rice varieties more efficiently and accurately, this study proposes a multimodal l fusion detection method based on an improved voting method. The experiment collected eight common rice seed types. Raytrix light field cameras were used to collect 2D images and 3D point cloud datasets, with a total of 3194 samples. The training and test sets were divided according to an 8:2 ratio. The experiment improved the traditional voting method. First, multiple models were used to predict the rice seed varieties. Then, the predicted probabilities were used as the late fusion input data. Next, a comprehensive score vector was calculated based on the performance of different models. In late fusion, the predicted probabilities from 2D and 3D were jointly weighted to obtain the final predicted probability. Finally, the predicted value with the highest probability was selected as the final value. In the experimental results, after late fusion of the predicted probabilities, the average accuracy rate reached 97.4%. Compared with the single support vector machine (SVM), k-nearest neighbors (kNN), convolutional neural network (CNN), MobileNet, and PointNet, the accuracy rates increased by 4.9%, 8.3%, 18.1%, 8.3%, and 9%, respectively. Among the eight varieties, the recognition accuracy of two rice varieties, Hannuo35 and Yuanhan35, by applying the voting method improved most significantly, from 73.9% and 77.7% in two dimensions to 92.4% and 96.3%, respectively. Thus, the improved voting method can combine the advantages of different data modalities and significantly improve the final prediction results.

Keywords: rice seed; variety classification; multimodal fusion; machine vision; point cloud

Citation: He, X.; Cai, Q.; Zou, X.; Li, H.; Feng, X.; Yin, W.; Qian, Y. Multi-Modal Late Fusion Rice Seed Variety Classification Based on an Improved Voting Method. *Agriculture* **2023**, *13*, 597. https://doi.org/10.3390/agriculture13030597

Academic Editor: Wen-Hao Su

Received: 30 January 2023
Revised: 24 February 2023
Accepted: 28 February 2023
Published: 1 March 2023

1. Introduction

As the most primitive and fundamental means of production in agricultural development, seeds not only determine the survival rate and growth activity of seedlings but also affect subsequent product processing. In agricultural production, with improvements in the production capacity and product quality requirements of various crops, effectively selecting and breeding good varieties has become a hot research topic.

Machine vision research is the process of processing visual information, usually including the image brightness, shape, position, color, and texture. Using machine vision to classify varieties can achieve the effect of nondestructive testing, so it has become a good research direction in recent years. Initially, the variety detection of rice seeds started from 2D images. In [1], an automatic rice quality evaluation system based on an artificial neural network (ANN) and support vector machine (SVM) classifiers was proposed. Experiments showed that the overall accuracy of the proposed ANN classifier was 83%, while that of the SVM was 91%. In [2], rice varieties were classified according to color, shape, and texture characteristics. Principal component analysis (PCA) was used to reduce the dimension of the data. Using discriminant analysis (DA), the accuracy of segregation of rice, brown rice, and white rice cultivars was 89.2%, 87.7%, and 83.1%,

respectively. To identify and classify the desired species, a multilayer perceptron neural network was implemented based on the most effective components. The results showed that the network was 100% accurate in identifying and classifying all of the mentioned rice varieties. In [3], using seven morphological features extracted from each variety of rice, a model was created by using LR (logistic regression), MLP (multilayer perceptron), SVM, DT (decision tree), RF (random forest), NB (naïve Bayes), and weighted k-nearest neighbor (kNN) machine learning techniques, and the performance measurement values were obtained. The experimental results showed that the classification accuracy rates of the models were 93.02% (LR), 92.86% (MLP), 92.83% (SVM), 92.49% (DT), 92.39% (RF), 91.71% (NB), and 88.58% (kNN). As a branch of machine learning, neural networks are gradually being widely used. A new method using a deep convolution neural network (CNN) as a general feature extractor was proposed in [4]. The extracted features were classified using an ANN, cubic SVM, quadratic SVM, kNN, boosted tree, bagged tree, and linear discriminant analysis (LDA). Compared with a model based on simple features, the model trained with CNN-extracted features showed better classification accuracy. The CNN-ANN classifier showed the best performance. The classification accuracy was 98.1%, recall 98.1%, and F1-score 98.1%, in 26.8 s. In [5], the authors proposed a seed classification system based on CNN and transfer learning, which contained models and used advanced deep learning techniques to classify 14 common seeds. The techniques used in that study included the decayed learning rate, model checkpointing, and hybrid weight adjustment. The proposed model exhibited 99% recognition accuracy for 234 training and testing images.

Compared with simple two-dimensional (2D) image recognition, the three-dimensional (3D) information obtained from the surface of rice seeds can describe the seed appearance more completely and accurately. However, the application of 3D computer vision in rice seed modeling is still at the research stage, and its implementation in crop seed modeling and nondestructive testing (NDT) is still being popularized on a small scale. In [6], a rice variety classification method based on 3D point cloud data of the rice seed surface and a deep learning network was proposed. The preprocessed point cloud was input into the improved PointNet network for feature extraction and variety classification. The average classification accuracy of the improved PointNet model for eight rice varieties was 89.4%. In [7], a rice seed recognition platform was constructed by combining 3D laser scanning technology and the BP neural network algorithm. Information on the rice seed surface was collected from four angles, and three morphological characteristics and projection characteristics of the main plane cross-section were obtained by feature calculation. The results showed that for input vectors composed of nine surface morphological features in 3D, the recognition rates of five rice varieties were 95%, 96%, 87%, 93%, and 89%, respectively. The recognition rates for the input vectors composed of nine projective features of the rice seed cross section were 96%, 96%, 90%, 92%, and 89%, respectively. The 3D grain character measurement method based on CT was studied in [8]. Here, 3D rice spike images were reconstructed by 3D reconstruction software, and grain phenotypes were analyzed. The results show that the recognition accuracy of a random forest classifier was higher than that of an LDA classifier and SVM classifier, and the average cognition accuracy was 95.19%.

The 2D and 3D models provide complementary information. Each pixel of an RGB image obtains various colors by changing the three color channels of red (R), green (G), and blue (B) and superimposing them. In 2D, the original image collected by the camera is an RGB image. RGB images have a higher resolution than the depth images or point clouds and contain rich textures not available in the point domain. In addition, images can cover "blind spots" caused by reflective surfaces that depth sensors cannot perceive. In contrast, 2D images are limited in 3D detection tasks because they lack absolute object depth and scale measures, which can be provided by 3D point clouds.

Multimodal technology helps artificial intelligence understand the external world more accurately by cooperating with perceptual information in multiple modalities. According to the chronological order of fusion, the methods for merging 2D images and 3D

point clouds can be divided into two types: early fusion and late fusion. Early fusion fuses the features extracted from different modalities, which is also called feature fusion [9]. In [10], the authors proposed a fusion method combining RGB and depth information. The model consisted of a two-stream CNN that can automatically fuse information from RGB and depth using a specific encoding method before classification. Finally, the goal of learning rich features from two domains was achieved. The authors of [11] proposed a method for fruit leaf disease classification based on feature fusion. They used transfer learning to adjust the extracted deep features and then fused multiple features into the final feature through a multilevel fusion algorithm based on entropy-controlled threshold calculation. The fused features were input into a main classifier multi-SVM. The experimental results showed that the method improved the recognition accuracy (97.8%) and sensitivity (97.6%) of the five diseases. The authors in [12] used partial least squares (PLS) regression to perform feature selection from the extracted deep feature set. The acquired features were input into the ensemble baggage tree classifier to realize the automatic disease identification of tomato, potato, and corn crops. The accuracy rate was approximately 90.1%. In [13], they proposed a corn seed variety detection method that weighted the data at different stages after the feature extraction of corn seed images and then fused the shallow features with the deep features to construct multiscale fusion features. Experiments showed that the average precision of the MFSwin Transformer model on the test set was 96.53%, which was higher than that of the other models. Late fusion fuses the prediction scores of multiple single modalities, also known as the score fusion [9]. The authors in [14] proposed a weed classification method by multimodal late-fusion deep neural networks (DNNs) using a Bayesian conditional probability-based method, or determining the priority weights to calculate the score vector. The results showed that the method was effective in plants. The accuracy rate on a seedling dataset was 97.31%. The study in [15] proposed a method to estimate the ripeness of papaya fruit by combining hyperspectral and visible light images, enabling multimodality through the late fusion of image-specific networks. Experimental results showed that the model obtained an improved F1-score of up to 0.97. The compatibility of early fusion and late fusion is relatively good, and this approach can adapt to most detection algorithms based on point clouds. However, when there is a problem with the classification model in early fusion, the correct detection of the variety can no longer be achieved. For late fusion, the result is fused by the classification results of multiple models, so this approach is less affected by a single model and is more robust.

It can be seen from the previous research results that most of the research on variety detection in the past only stayed in a single 2D or 3D, and did not combine the advantages of the two modes. In addition, there is a lack of effective weights to predict the outcome during late fusion. Therefore, on the basis of our predecessors, we proposed a new experimental approach. In this study, 2D RGB images and 3D point cloud data captured by Raytrix light field cameras were used as input data for recognition, and an improved voting method was used to fuse the recognition results. We pursued this research goal as follows. (1) The data of the 2D rice pictures and 3D point cloud sets were established and divided into a training set and a validation set at a ratio of 8:2. (2) SVM, kNN, CNN, and MobileNet were used to classify the 2D images. PointNet models were used to classify the 3D point clouds. Finally, a voting method was used to fuse the classification results of multiple models to obtain the final variety detection results. (3) The final classification results were evaluated and compared with the general model classification results through visualization. For the results of this study, we propose the following hypotheses. (1) Multimodal fusion to achieve variety detection can achieve higher accuracy by combining the advantages of 2D and 3D. (2) The late-fusion method can improve the accuracy and robustness of the classification process. Moreover, when the homogeneity of multiple classification models is smaller, the accuracy of the final fusion result is higher.

2. Materials and Methods

2.1. Point Cloud Collection System

The rice seed 3D point cloud acquisition hardware platform included a camera and lens, a vertical lifting device, a light source and its control module, and a camera calibration tool. There were four parts in total. The research-grade light field camera was model R42, manufactured by the German Raytrix, with a maximum resolution of 41.5 MegaRays and 7708 × 5352 pixels. The imaging lens used in the acquisition process was a 3D light field lens with a focal length of 50 mm and an aperture of F/2.80. The camera and lens were composed of a light field camera and an imaging lens. The computer was equipped with a high-performance GPU, model NVIDIA GTX 1080, for real-time light field processing. The camera calibration tool is a calibration board evenly covered with a dot matrix with a pitch of 2.09 mm. The point cloud acquisition hardware platform collects 3D point cloud data on the surface of rice seeds through cameras and lenses and uses vertical lifting devices to realize rough adjustment and fine adjustment of the height of the camera placement. Camera calibration tools are used to perform light field camera calibration, and the light source control module controls the environment of the experimental site. The experimental environment for point cloud collection is shown in Figure 1.

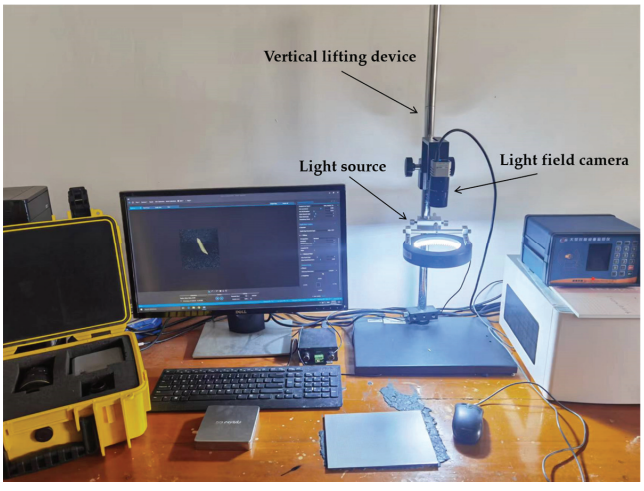


Figure 1. Point cloud collection hardware platform.

The camera operation software supports RxLive4.0 software. The software can identify the connected camera, control it, and change the camera parameters. The light field camera can be calibrated using the camera calibration module in the software, with the function of evaluating the calibration effect and evaluating the grade of the calibration result. The software includes a point cloud preprocessing function, which can perform filtering, noise reduction, sharpening, smoothing, and cropping on the point cloud. The data export function can set various file types, file naming formats, and export file storage locations for data export.

2.2. Dataset Preparation

In this study, eight common rice seeds in China were selected as the datasets for model training and testing. The seeds selected for the experiment were preliminarily screened and cleaned manually to avoid irregularities such as attached impurities and gaps from affecting the classification results. The storage of the selected samples strictly follows the standardized storage environment. Rice seeds were stored in a dry, low temperature, and airtight environment to avoid being affected by the external environment. The seeds

included in the dataset were Nanjing9108, Zhenghan10, Hannuo35, Yuanhan35, Liannuo13, Hyou518, Huanghuazhan, and Liusha, and the representative RGB images of each category are shown in Figure 2.

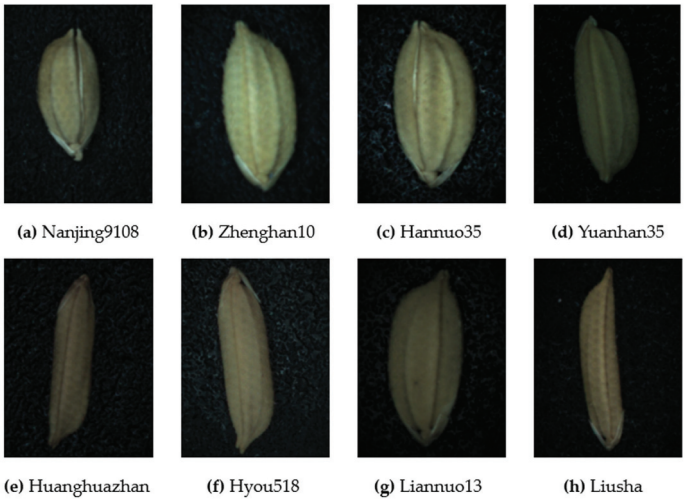


Figure 2. RGB images of the maize seed grains. (**a**) Nanjing9108. (**b**) Zhenghan10. (**c**) Hannuo35. (**d**) Yuanhan35. (**e**) Huanghuazhan. (**f**) Hyou518. (**g**) Liannuo13. (**h**) Liusha.

The above-mentioned seeds were placed separately on the camera stage to sequentially collect the 2D images and 3D point clouds of the front and back sides. After simple preprocessing of the 2D image, the redundant background was cut according to the size and shape of the seed. Establishing the 3D point cloud first requires preprocessing operations such as denoising, smoothing, and cutting on the original data. However, the rice seed point cloud collected at this time still contains considerable redundant data. Storing, processing, and displaying these point cloud data would increase the burden on the computer processing process and at the same time occupy a large amount of computer hardware and software resources, reducing the efficiency of the operation process. However, if the point cloud is too small, it will lose its features for classification. Therefore, this study downsampled the point cloud to a scale of 2048 points. The final point cloud data after processing are shown in Figure 3.

Figure 3. The point cloud data after processing.

In the experiment, the dataset of each rice species was divided into a training set and a test set at a ratio of 8:2. Each seed had a corresponding 2D picture and 3D point cloud on the front and back. There were eight varieties for a total of 3194 samples; the total size of the training set was 2560, and the total size of the test set was 634, as shown in Table 1.

Table 1. Rice seed dataset.

No.	Cultivar Name	Training Set	Validation Set
1	Nanjing9108	320	79
2	Zhenghan10	320	79
3	Hannuo35	320	79
4	Yuanhan35	320	80
5	Huanghuazhan	320	80
6	Hyou518	320	78
7	Liannuo13	320	80
8	Liusha	320	79
Total		2560	634

2.3. Classification Model

For the voting method, we need to consider the possible impact of different base models. In theory, the base model can be any model that has been trained. However, in practical applications, if the voting method is to produce better results, two conditions need to be met:

1. The effect between the base models cannot be too different. When a base model performs poorly relative to other base models, the model is likely to be noisy.
2. There should be less homogeneity among the base models. For example, when the prediction effect of the base model is similar, voting based on a tree model and a linear model is often better than voting based on two tree models or two linear models.

Based on the above principles, this paper selected SVM, kNN, CNN, and MobileNet as the base models for 2D classification and point net as the base model for 3D detection.

The SVM is a method of machine learning that was developed based on statistical theory [16]. When SVM is used for classification, it can achieve better classification results when the number of training samples is smaller. Different varieties of rice seeds have large differences in color from the appearance point of view, so an image histogram can be used for classification. First, the image is scaled to a uniform size, and then the image histogram is calculated, as shown in Figure 4. The histogram can be used to obtain the number of pixels of each brightness level of the image of each sample, and displays the distribution of the pixels in the image. The SVM was used to process the image histogram, and the kernel function was linear. The trained model was used to predict the test set, and the prediction probability corresponding to each seed was saved as the input data of the final voting method.

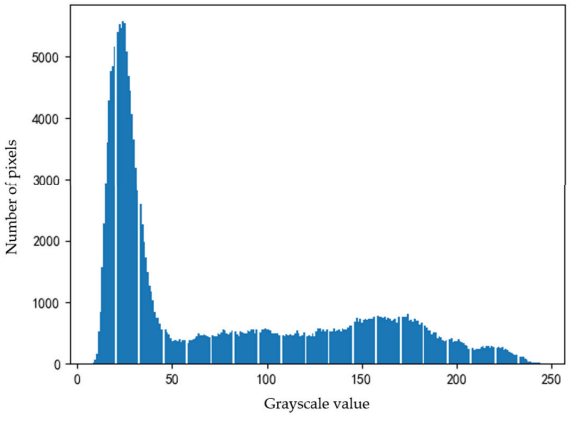


Figure 4. Calculated image histogram.

In machine learning, the kNN algorithm is a widely used classification and regression method [17]. This algorithm determines the similarity of the samples to be tested according to the distance characteristics of the nearest neighbor samples to classify them; that is, the category of the samples to be tested is determined by calculating the distance between the sample to be tested and the k-nearest neighbor samples in the training set. The three basic elements of the kNN algorithm are the distance measure, the selection of the k value of the number of neighbors, and the classification decision rule. The histogram calculated according to the training set is input into the kNN model with the neighbor value k parameter of 11 for training to obtain the predicted probability.

The CNN, a type of neural network, is one of the best algorithms for image content, and it performs very well in related operations such as image segmentation, classification, detection, and retrieval. CNNs are structurally composed of multilayer networks, and each layer can be regarded as a plane composed of independent neurons. The main layers include the input layer, convolutional layer, pooling layer, fully connected layer, and classifier [18]. In this experiment, the CNN model was first normalized, and numbers between 0–255 were normalized to between 0 and 1. Then, the output was set to a convolutional layer with 32 channels; the size of the convolution kernel was 3×3, and the activation function was ReLU. Then, a pooling layer was added, with a pooled kernel size of 2×2. Then, the output was set to a convolutional layer with 64 channels; the convolution kernel size was 3×3, and the activation function was ReLU. Another pooling layer was added to perform a pooling operation on a 2×2 area. Finally, the 2D output was converted into one-dimensional output through the softmax function to output the model to the neuron of the class name length, and the activation function adopted the corresponding probability value of softmax. The specific network architecture of the CNN is shown in Figure 5.

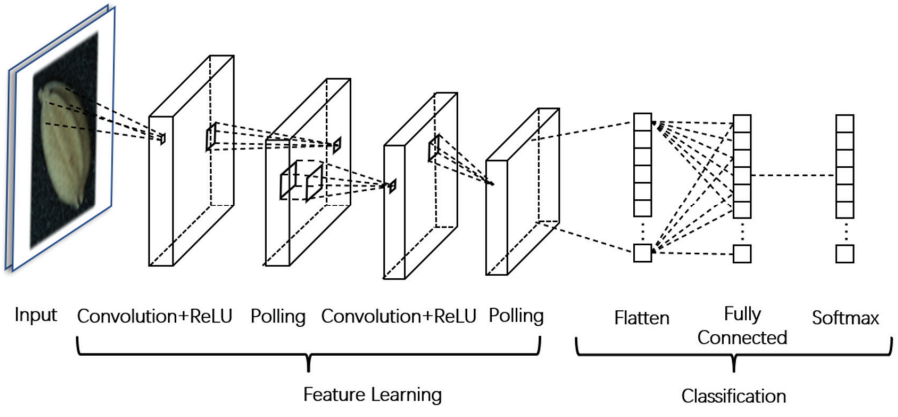


Figure 5. CNN network structure.

Compared with the CNN, MobileNet abandons the traditional convolution and combines depth-wise convolution and pointwise convolution as the basic network module [19]. We call this approach depth-wise separable convolution. It uses a very simple stacking structure that has the advantage of improving network computing efficiency and reducing the number of parameters. In this study, we first loaded the pretrained MobileNet model as the backbone model and normalized the input image. Then, the output of the backbone model was the global average pooled and mapped to the final classification number through the fully connected layer.

PointNet is a pioneering approach to feeding point cloud data directly into neural networks. The framework mainly solves the problems of point cloud disorder and permutation invariance. Considering the disorder of the point cloud, PointNet does not convert the point cloud into a multi-view or voxel grid but processes the points directly. For permuta-

tion invariance, this method used a multilayer perceptron to extract features independently for each point and then uses the maximum pooling layer to aggregate the information of all points to obtain global features. In addition, the framework adds T-Net to spatially align the input point cloud and its features by constructing a transformation matrix to solve the problem of transformation invariance. This study used the basic PointNet network model as the classification model, and its structure is shown in Figure 6. After inputting the point cloud data, T-Net was first performed for affine transformation, which was specifically expressed as multiplying the transformation matrix by 3×3, and then feature extraction was performed through the convolutional layer. According to the model structure, the number of convolution kernels of the two MLP convolutional layers (64, 64) was 64. The convolution kernel size of the first layer of convolution was 1×3, and the second layer was a 1×1 kernel. Then, the same feature transform was performed, and in the next MLP (64, 128, 1024), the size of the convolution kernel was 1×1. After the pooling layer, three fully connected layers were connected, and the number of output nodes was 512, 256, and k in turn. Finally, the softmax function was used to obtain the result.

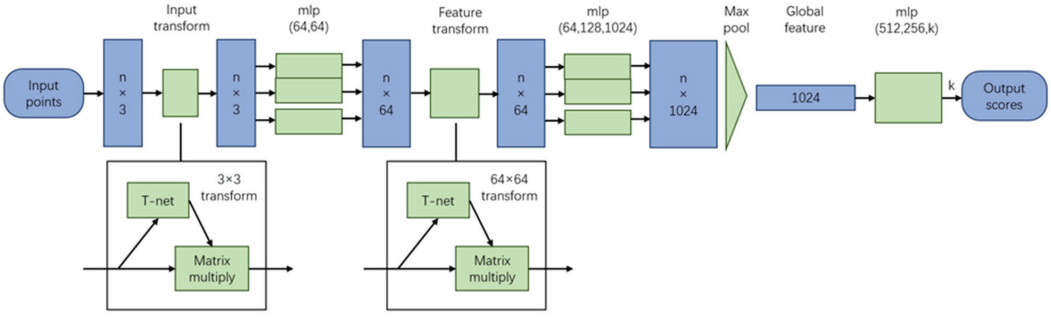


Figure 6. Basic PointNet model structure used in this article.

After the above model training was completed, the results were used to predict the data of the test set and output the predicted probability of each sample separately as the input data of the voting method for late fusion.

2.4. Improved Voting Method

The voting method is a commonly used technique in ensemble learning that can improve the generalization ability of the model and reduce the associated error rate. The traditional voting method follows the principle of the minority obeying the majority and integrates multiple models to reduce the variance and improve model robustness. Ideally, the forecasting performance of the voting method should be better than that of any one of the base models. When the voting method is applied to the classification model, its prediction result is the most frequent prediction result among all models. According to different prediction methods, classification voting can be divided into hard voting and soft voting. Hard voting simply counts the most common class among all model predictions as the final result. Soft voting calculates the sum of the probability values of the prediction results of each model and selects the class with the highest probability value as the final result. Compared with the hard voting method, the soft voting method takes into account the additional information of the prediction probability, enabling it to obtain more accurate prediction results than the hard voting method.

The traditional voting method has certain limitations: it treats all models the same. That is, for the voting method, all models contribute equally to the prediction. Vote predictions can be biased if some models are good in certain situations but poor in others. Therefore, this study presents an improved soft voting method, which improves the process of calculating the arithmetic mean in the traditional method. This method determines the weights of different models according to their performance scores, combines the probabili-

ties of the predicted result classes of each model to obtain the comprehensive score vector of each model, and uses this vector to determine the predicted results. The late-fusion process based on the improved voting method is shown in Figure 7.

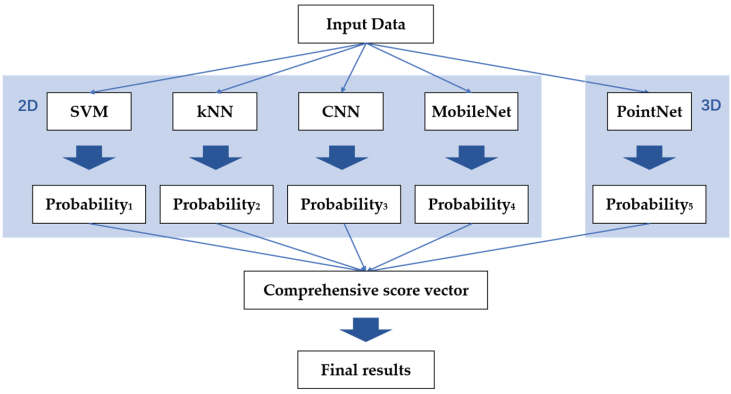


Figure 7. Flowchart of the improved voting method.

The F1-score is an indicator used to measure the accuracy of a binary classification model. It takes into account both the precision and recall of the classification model. Later, the traditional F1-score was extended to a multicategory F1-score, which can be divided into macro-F1 and micro-f1 according to the suitable dataset. Macro-F1 is applicable to the classification situation where each category has equal status and the same size [20]. Since the size of the dataset of each variety in this experiment is the same, the Macro-F1 value can be used to determine the scoring vector of each model.

In the binary classification problem, it is assumed that the sample has two categories: positive and negative. When the classifier prediction ends, we can divide the classification results into the following categories:

- True positive (TP): Positive samples are successfully predicted as positive.
- True negative (TN): Negative samples are successfully predicted as negative.
- False positive (FP): Negative samples are incorrectly predicted as positive.
- False negative (FN): Positive samples are incorrectly predicted as negative.

In the binary classification problem, the calculation method of the F1-score is as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Re}call = \frac{TP}{TP + FN} \tag{2}$$

$$F1 - score = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

The F1-score can balance the two indicators of precision and recall at the same time, so it can be used to reflect the classification performance of the model. To extend the calculation method in the binary classification problem to the multiclassification problem, each category can be regarded as a binary classification problem, and the precision and recall can be calculated separately. Therefore, for each class, the distribution of its results in the confusion matrix is as shown in Figure 8.

Actual \ Predicted	Class1	Class2	Class3
Class1			FP
Class2			FP
Class3	FN	FN	TP

Figure 8. Distribution of TP, FP, and FN in multiclassification problems.

Formulas (1) and (2) were used to calculate the respective precision, recall, and F1-score in each category, respectively, denoted as $P_1, P_2, \ldots, P_n$; $R_1, R_2, \ldots, R_n$; and $F_{11}, F_{12}, \ldots, F_{1n}$. Since the datasets of each variety have the same size, the overall Macro-F1 calculation method is as follows:

$$Macro - F1 = \frac{\sum_{i=1}^{n} F1_i}{n} \tag{4}$$

Macro-F1 calculated by each multiclassification model is used as the weight of voting to form a scoring vector, which is recorded as $S_1, S_2, \ldots, S_n$. Assuming that there are m varieties in total, the probability of each model predicting that the result is m is $M_1, M_2, \ldots, M_n$. Then, the final probability (Final-P) of each variety predicted by the voting method is:

$$Final - P = \frac{\sum_{i=1}^{n} M_i \times S_i}{n} \tag{5}$$

The variety corresponding to the highest probability is selected as the final prediction result.

3. Results and Discussion

First, we used 2D pictures as input datasets to train SVM, kNN, CNN, and MobileNet.

For both the CNN and MobileNet, 30 epochs are allowed to pass. The accuracy and loss changes of the training set and validation set in each epoch of the two models are represented by line graphs, as shown in Figure 9.

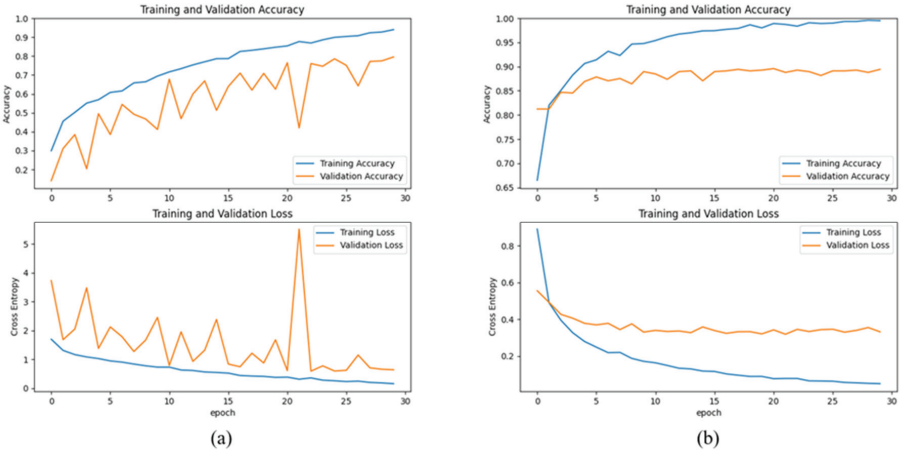


Figure 9. Accuracy and loss of the training set and validation set for each epoch: (**a**) CNN, (**b**) MobileNet.

In the early stage of training (the first ten epochs), the overall accuracy of the model gradually increases, and the loss gradually decreases. However, the two indicators of the CNN's validation set fluctuate greatly, while those of MobileNet change steadily with small fluctuations. At the twentieth epoch, the accuracy and loss of the CNN continue to fluctuate greatly. It may be that the appearance similarity of some rice varieties is relatively high, which affects the classification effect of the model but then gradually reduces the fluctuation range. Both curves tend to be smooth. Compared with the CNN, the change in MobileNet is more stable. The convergence is basically completed at the 15th epoch, and the curve basically fluctuates over a small range only. A comparison of the two neural networks revealed that the training process of CNN fluctuated greatly, while MobileNet was relatively stable. After the final stabilization, the accuracy of the CNN reached 79%, and the loss was 0.64. The accuracy of MobileNet was 89%, and the loss was 0.33. Although the accuracy of the CNN was relatively low, this did not have a negative impact on the final result because the prediction probabilities of multiple models need to be fused during late fusion.

We input the 3D coordinates of the processed point cloud data into the PointNet model; Figure 10 shows the accuracy and loss variation during the steps of the training process. From 0 to 2000 steps, the accuracy and loss varied greatly, and the curves were very steep. After 6000 steps, both the accuracy and loss gradually converged, basically fluctuating over a small range only. This change showed that the result of this training was convergent, and the final average accuracy of PointNet was 88.75%. Compared with other models, the recognition accuracy of PointNet was not very high. The reason may be that the method of downsampling during dataset preprocessing is not effective enough, with many important feature points screened out in the process. Therefore, the final classification effect was affected to a certain extent.

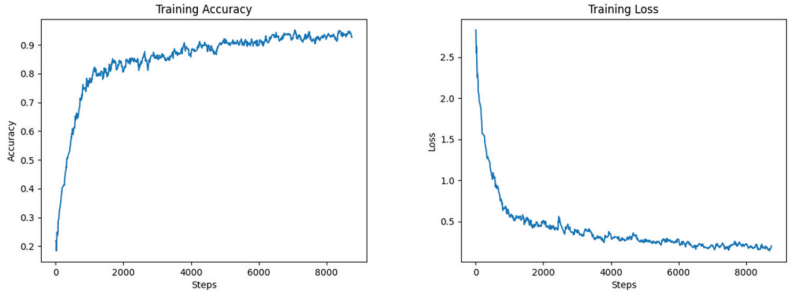


Figure 10. Changes in the accuracy and loss of the training set with the step size.

After all of the model training had been completed, we uniformly used the test set for testing and calculated the accuracy of the different models for different types of seeds, as shown in Figure 11. The four models for classifying 2D images had high recognition accuracy for Huanghuazhan and Liusha, and the recognition accuracy reached 98% and 95%, respectively. Moreover, the recognition accuracy of the four 2D models for Huanghuazhan was maintained over the extremely small range of 96% to 99%, indicating that each model has a good classification effect on this variety. However, for Hannuo35 and Yuanhan35 rice seeds, the recognition accuracy for the 2D images was relatively low, and the accuracy of the CNN for Hannuo35 was the lowest, only 52%. The reason may be the appearance similarity of these two kinds of seeds; the currently proposed model may not be adaptable enough to them. In the 3D point cloud recognition, the classification accuracy for each species of seeds using the PointNet model indicates that the difference in the recognition accuracy for the eight seeds was small, with the values remaining between 85% and 95%. The recognition accuracy for Zhenghan10 was relatively low, at 85%. Liusha had the highest recognition accuracy, reaching 92%. Compared with the accuracy for some varieties in 2D, which was notably low, the recognition effect for 3D was higher, and the

values were all maintained at a high level. This shows that the point cloud data can better distinguish the eight kinds of seeds, especially Hannuo35 and Yuanhan35, which cannot be classified correctly in 2D. Therefore, the 3D point cloud data can be used as an effective supplement to 2D classification results.

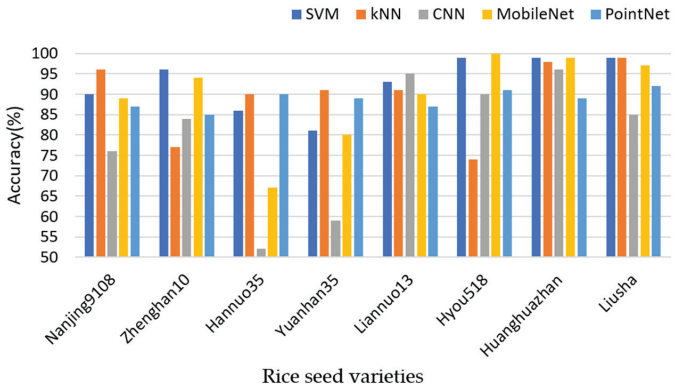


Figure 11. Accuracy of the different models for each species of rice seeds.

According to the prediction results, we can build a confusion matrix for each algorithm separately, as shown in Figure 12. TP, FP, and FN of the model can be calculated through this matrix, and the corresponding precision, recall, and Macro-F1 can be calculated based on these parameters. Macro-F1 is used as the weight for late fusion. The final evaluation index calculation results are shown in Table 2.

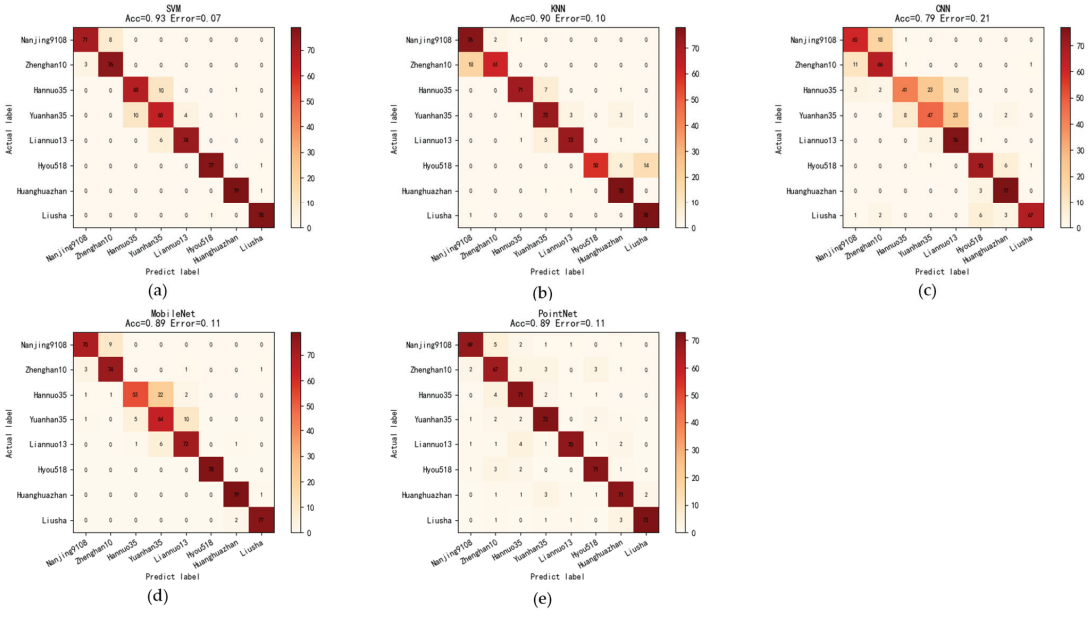


Figure 12. Confusion matrix. (**a**) SVM. (**b**) kNN. (**c**) CNN. (**d**) MobileNet. (**e**) PointNet.

Table 2. Measures to evaluate the model performance.

Model	Macro-Precision	Macro-Recall	Macro-F1
SVM	0.928	0.929	0.929
kNN	0.908	0.894	0.894
CNN	0.801	0.795	0.790
MobileNet	0.899	0.894	0.894
PointNet	0.892	0.889	0.890

Macro-F1 was combined as the comprehensive scoring vector of the voting method. The SVM, kNN, CNN, MobileNet, and PointNet models predicted the probability of each sample in the test set as the input of the voting method. The models used the comprehensive scoring vector for weighted combination, and selected the recognition result with the highest probability as the final prediction value of late fusion. Table 3 shows the prediction results of all kinds of rice seed varieties after the final statistic of the improved voting method for late fusion. Using the improved voting method for late fusion, the final accuracy was 97.4%. Finally, the prediction accuracy for all varieties was more than 90%. Compared with the recognition accuracy of each model alone, the prediction results after fusion by the voting method were significantly improved. The recognition accuracy for Hyou518, Huanghuazhan, and Liusha was 100%. Although the accuracy for Hannuo35 was lower than that of the other varieties, it also reached 92.4%. The recognition accuracy of each variety before and after late fusion is compared in Table 3. It can be seen from the results that the improved voting method improved the accuracy of Hannuo35 and Yuanhan35 the most, from the average accuracy of the 2D recognition of 73.9% and 77.7% to the final accuracies of 92.4% and 96.3%, respectively. For these two kinds of rice, the classification effect of the 2D classification model is not ideal, but the classification accuracy of PointNet was relatively high. The 2D recognition effect was poor, for which there may be two reasons. One is that some 2D models (such as CNN and MobileNet) have poor classification effects on these two rice species. Second, the difference between the 2D images of these two rice species and other varieties is relatively small, so it will cause interference, and a large number of correct prediction results cannot be obtained. However, their differences in the 3D point cloud data were more obvious, so the 3D point cloud features can be used to classify them. For seeds such as Zhenghan10, Liannuo13, and Nanjing9108, PointNet's classification effect was not ideal, and the accuracy was lower than 89%. The reason may be that the differences in these seeds on the 3D point cloud were not very obvious and cannot provide a reliable basis for classification. However, they can be efficiently classified using the feature values of their 2D images.

Table 3. Accuracy comparison for identification of various rice varieties.

Varieties	SVM	kNN	CNN	MobileNet	PointNet	Late Fusion
Nanjing9108	90.1%	96.3%	75.8%	88.7%	87.4%	98.9%
Zhenghan10	96.1%	77.4%	84.1%	93.8%	85.0%	97.4%
Hannuo35	86.2%	90.3%	52.1%	67.0%	89.0%	92.4%
Yuanhan35	80.6%	91.4%	58.6%	80.0%	90.0%	96.3%
Huanghuazhan	93.2%	91.3%	94.7%	90.3%	87.5%	97.5%
Hyou518	99.1%	77.4%	90.3%	100.0%	91.3%	98.9%
Liannuo13	99.0%	99.2%	95.5%	98.9%	89.8%	98.9%
Liusha	98.5%	99.4%	84.8%	97.4%	90.0%	98.9%

Table 4 reflects the time taken by each model to predict the test separately and the time consumed to make predictions using late fusion. Comparing the late fusion time with the time consumption of the previous single model, it is not difficult to see that the time complexity caused by late fusion using the improved voting method mainly depends on the model it chooses. In other words, the more complex the model selected (such as

PointNet), the longer it takes. However, the voting method takes only 23.62 milliseconds, and it can be seen that it does not generate a large computational burden.

Table 4. Time-consuming comparison of various classification methods.

Model	Prediction Time
SVM	18.75 s
kNN	19.98 s
CNN	10.35 s
MobileNet	20.76 s
PointNet	9792 s
Late fusion	9861.86 s

By comparing this experiment with the existing research, it can be seen that the point cloud data obtained by using the light field camera can be used as an important basis for 3D classification. Furthermore, past studies have only focused on a single modality in 2D or 3D. However, the information contained in the two modalities can complement each other to classify from multiple perspectives. In the late fusion, this study replaces the process of calculating the average value in the traditional voting method by calculating the comprehensive score vector of each model separately. The prediction results of each model were weighted and fused by using the score vector. Experimental results showed that this fusion method can not only comprehensively evaluate multiple modalities, but also correct the prediction results of individual models with poor classification effects and error-prone models. Finally, the classification of rice seed varieties was effectively realized.

4. Conclusions

Based on the principle of multimodal fusion, we experimentally evaluated a rice variety classification method that used an improved voting method to perform late fusion of 2D and 3D modalities. The experimental data came from eight common rice varieties in China, and a Raytrix light field camera was used to collect 2D images and 3D point cloud data. We proposed an improved late-fusion method to generate a dynamically changing scoring vector according to the actual situation of the model, which was used to adjust the influence on the final prediction result. After preprocessing the data by noise reduction, filtering, and sampling, a dataset was obtained for classification. We input the dataset into models corresponding to the modality to obtain the predicted probabilities of the test set. The scoring vector was used to calculate the probability weighting of different models, and the predicted value with the highest final probability was selected as the final value. Compared with other multimodal fusion methods, this method was more robust. Its prediction results are not easily affected by a single model, and at the same time, it avoids possible interference from excessive model homogeneity or poor performance. The improved voting method was used to perform late fusion on the prediction results of the test set, and the final average accuracy reached 97.4%. Compared with a single SVM, kNN, CNN, MobileNet, and PointNet, the accuracy was 4.9%, 8.3%, 18.1%, 8.3%, and 9.0% higher, respectively. It can be seen that late fusion had the best effect on improving the accuracy of CNN and late fusion improved the identification accuracy of Hannuo35 and Yuanhan35 most obviously. The experimental results showed that the improved voting method can combine the advantages of different modal data and significantly improve the final prediction results.

This study provides a new perspective for the future classification of rice varieties. In future experiments, the rice seed dataset can be further expanded to provide sufficient data for the recognition algorithm. The preprocessing of point cloud data can be further optimized. The selection of points during sampling is not effective enough. Some important feature points may be deleted during preprocessing, which ultimately affects the classification results of the model. In addition, to pursue fast detection, the point cloud of

this experimental species is half a seed, and the next step is to register the point cloud data to obtain the complete seed data for training to obtain better classification results.

Author Contributions: Conceptualization, X.H., X.Z. and Y.Q.; Data curation, H.L.; Formal analysis, H.L.; Investigation, X.F.; Methodology, X.H., X.Z. and Y.Q.; Project administration, X.Z. and Y.Q.; Resources, W.Y.; Supervision, X.Z. and Y.Q.; Validation, X.H. and Q.C.; Visualization, Q.C.; Writing—original draft, X.H.; Writing—review & editing, X.H., X.Z. and Y.Q. 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 Youth Fund Project (51305182); the Ministry of Agriculture Key Laboratory of Modern Agricultural Equipment (201602004); and the China University Industry-University-Research Innovation Fund Project (2021ZYA03010).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Mohan, D.; Raj, M.G. Quality Analysis of Rice grains using ANN and SVM. *J. Crit. Rev.* **2020**, *7*, 395–402.
2. Abbaspour-Gilandeh, Y.; Molaee, A.; Sabzi, S.; Nabipur, N.; Shamshirband, S.; Mosavi, A. A Combined Method of Image Processing and Artificial Neural Network for the Identification of 13 Iranian Rice Cultivars. *Agronomy* **2020**, *10*, 117. [CrossRef]
3. Cinar, I.; Koklu, M. Classification of rice varieties using artificial intelligence methods. *Int. J. Intell. Syst. Appl. Eng.* **2019**, *7*, 188–194. [CrossRef]
4. Javanmardi, S.; Ashtiani, S.H.M.; Verbeek, F.J.; Martynenko, A. Computer-vision classification of corn seed varieties using deep convolutional neural network. *J. Stored Prod. Res.* **2021**, *92*, 101800. [CrossRef]
5. Gulzar, Y.; Hamid, Y.; Soomro, A.; Alwan, A.; Journaux, L. A convolution neural network-based seed classification system. *Symmetry* **2020**, *12*, 2018. [CrossRef]
6. Qian, Y.; Xu, Q.; Yang, Y.; Lu, H.; Li, H.; Feng, X.; Yin, W. Classification of rice seed variety using point cloud data combined with deep learning. *Int. J. Agric. Biol. Eng.* **2021**, *14*, 206–212. [CrossRef]
7. Feng, X.; He, P.; Zhang, H.; Yin, W.; Qian, Y.; Cao, P.; Hu, F. Rice seeds identification based on back propagation neural network model. *Int. J. Agric. Biol. Eng.* **2019**, *12*, 122–128. [CrossRef]
8. Zhang, C. *Non-Destructive Measurement and Phenotypic Analysis of 3D Rice Grain Based on CT*; Huazhong University of Science and Technology: Wuhan, China, 2020.
9. Abavisani, M.; Joze, H.R.V.; Patel, V.M. Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, California, CA, USA, 12 August 2019; pp. 1165–1174.
10. Eitel, A.; Springenberg, J.T.; Spinello, L.; Riedmiller, M.; Burgard, W. Multimodal deep learning for robust RGB-D object recognition. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 681–687.
11. Khan, M.A.; Akram, T.; Sharif, M.; Saba, T. Fruits diseases classification: Exploiting a hierarchical framework for deep features fusion and selection. *Multimed. Tools Appl.* **2020**, *79*, 25763–25783. [CrossRef]
12. Saeed, F.; Khan, M.A.; Sharif, M.; Mittal, M.; Goyal, L.M.; Roy, S. Deep neural network features fusion and selection based on PLS regression with an application for crops diseases classification. *Appl. Soft Comput.* **2021**, *103*, 107164. [CrossRef]
13. Bi, C.; Hu, N.; Zou, Y.; Zhang, S.; Xu, S.; Yu, H. Development of deep learning methodology for maize seed variety recognition based on improved swin transformer. *Agronomy* **2022**, *12*, 1843. [CrossRef]
14. Trong, V.H.; Gwang-hyun, Y.; Vu, D.T.; Jin-young, K. Late fusion of multimodal deep neural networks for weeds classification. *Comput. Electron. Agric.* **2020**, *175*, 105506. [CrossRef]
15. Garillos-Manliguez, C.A.; Chiang, J.Y. Multimodal Deep Learning via Late Fusion for Non-Destructive Papaya Fruit Maturity Classification. In Proceedings of the 2021 18th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 10–12 November 2021; pp. 1–6.
16. Liakos, K.G.; Busato, P.; Moshou, D.; Pearson, S.; Bochtis, D. Machine learning in agriculture: A review. *Sensors* **2018**, *18*, 2674. [CrossRef] [PubMed]
17. He, Q.P.; Wang, J. Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes. *IEEE Trans. Semicond. Manuf.* **2007**, *20*, 345–354. [CrossRef]
18. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 1–74. [CrossRef] [PubMed]

19. Jaithavil, D.; Triamlumlerd, S.; Pracha, M. Paddy seed variety classification using transfer learning based on deep learning. In Proceedings of the 2022 International Electrical Engineering Congress (iEECON), IEEE, Khon Kaen, Thailand, 9–11 March 2022.
20. Juri, O.; Burst, S. Macro f1 and macro f1. *arXiv* **2019**, arXiv:1911.03347.

agriculture

MDPI

Article

Research on Winter Wheat Growth Stages Recognition Based on Mobile Edge Computing

Yong Li [1,2], Hebing Liu [1,2], Jialing Wei [1,2], Xinming Ma [1], Guang Zheng [1] and Lei Xi [1,2,*]

[1] College of Information and Management Science, Henan Agricultural University, Zhengzhou 450046, China
[2] Henan Engineering Laboratory of Farmland Monitoring and Control, Zhengzhou 450046, China
* Correspondence: xil@henau.edu.cn

Abstract: The application of deep learning (DL) technology to the identification of crop growth processes will become the trend of smart agriculture. However, using DL to identify wheat growth stages on mobile devices requires high battery energy consumption, significantly reducing the device's operating time. However, implementing a DL framework on a remote server may result in low-quality service and delays in the wireless network. Thus, the DL method should be suitable for detecting wheat growth stages and implementable on mobile devices. A lightweight DL-based wheat growth stage detection model with low computational complexity and a computing time delay is proposed; aiming at the shortcomings of high energy consumption and a long computing time, a wheat growth period recognition model and dynamic migration algorithm based on deep reinforcement learning is proposed. The experimental results show that the proposed dynamic migration algorithm has 128.4% lower energy consumption and 121.2% higher efficiency than the local implementation at a wireless network data transmission rate of 0–8 MB/s.

Keywords: mobile edge computing; convolutional neural network; deep reinforcement learning; wheat growth stages detection; dynamic migration algorithm

Citation: Li, Y.; Liu, H.; Wei, J.; Ma, X.; Zheng, G.; Xi, L. Research on Winter Wheat Growth Stages Recognition Based on Mobile Edge Computing. *Agriculture* **2023**, *13*, 534. https://doi.org/10.3390/agriculture13030534

Academic Editors: Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 12 December 2022
Revised: 16 February 2023
Accepted: 17 February 2023
Published: 23 February 2023

1. Introduction

Wheat is the second-largest food crop in the world and is crucial for food security and social stability [1]. Wheat growth monitoring refers to recording the morphological changes in wheat during different growth and development stages [2]. It is critical on smart farms to obtain high yields and is often performed using unmanned aerial vehicles (UAVs) and intelligent agricultural machinery [3]. Due to technological advances in smart agriculture, and intelligent agricultural machinery and mobile devices, deep learning (DL) models and algorithms have been increasingly used in this field [4]. However, mobile devices have relatively low computing power, low battery capacity, and high energy consumption. DL-based agricultural applications require mobile computing devices with high computing power, high battery capacity, and low energy consumption to provide longer working hours and better service quality. Thus, an imbalance exists between the high computing needs of smart agriculture and mobile devices with low computing power. Therefore, it is necessary to develop a lightweight DL model capable of running on intelligent mobile devices for wheat growth monitoring. As the use of artificial intelligence has increased, deep reinforcement learning (DRL) has attracted extensive attention from the academic community [5]. The data generated by users show exponential growth, promoting the rapid development of DRL. The deep Q-learning network (DQN) is an unsupervised learning algorithm based on reinforcement learning and a neural network [6]. It combines the learning ability of neural networks and the decision-making ability of reinforcement learning and can make decisions in a timely and intelligent manner according to changes in the environment [7].

Edge computing is an ideal solution for real-time applications to upload the core parameters or data of the DRL model to the network edge for processing [8,9]. Running

a DQN on an intelligent mobile device causes high battery energy consumption, and the model's identification efficiency depends on the quality of the network service when it runs on a remote server. Therefore, the server location, the power of the mobile device, and the quality of the network service must be carefully selected to enable the use of a DQN so that the unloading strategy of the edge nodes can be adapted to the environment. This approach enables the use of relatively few resources to obtain optimal results and reduces the communication and computing costs of edge computing [10,11]. Migration is used in mobile edge computing to migrate intensive computing tasks to the wireless network edge server for processing, alleviating the shortcomings of low computing power, poor real-time performance, and large power consumption of intelligent devices. This technology has attracted the attention of academia and industry [12–15], especially optimal migration decisions and the allocation of computing resources [16,17]. Chen et al. [18] proposed a task unloading and scheduling method based on DRL for unloading decisions with dependency in mobile edge computing. The goal was to minimize the application's execution time. Experiments showed that the proposed algorithm has good convergence ability, verifying the effectiveness and reliability of the method. Tian et al. [19] deployed a cognition model to the edge and designed an intelligent recognition device based on computer vision and edge computing for crop pest image recognition. Agricultural crop images were collected in realtime, and image recognition was used to identify crop pests. Zhang et al. [20] proposed an improved algorithm called the natural deep Q-learning network (NDQN) for resource scheduling and decision-making in edge computing. The results showed that the improved NDQN algorithm performed better than the local unloading and random unloading algorithms. Gu et al. [21] designed an embedded monitoring system based on edge computing that considered different planting conditions of crops in different regions. They established neural networks and crop data processing algorithms and deployed them in embedded devices. UAVs were used for crop monitoring. However, most of the above studies designed migration algorithms for relatively large computing tasks and complex models [22–26], whereas few studies designed migration strategies or algorithms based on lightweight recognition models for intelligent agricultural production scenarios.

Wheat is an important grain crop and is grown extensively worldwide. Wheat growth monitoring algorithms have high computational complexity, many parameters, and long task execution times. They require extensive computing resources and sufficient battery power. General migration algorithms and intelligent equipment are inadequate. This paper proposes a lightweight wheat growth stage detection model for intelligent devices. The wheat growth stage detection model is migrated to the wireless network edge server for processing to reduce energy consumption and computing time and simulate the cost of intelligent devices to make decisions by calculating the weighted sum of the battery energy consumption and computing time delay. The DQN algorithm is used to obtain the optimal output model because it reduces energy consumption and computing time delay in the DL model. The proposed method enables complex computing tasks on intelligent mobile devices in smart agriculture, and its use for the accurate identification of wheat growth stages is demonstrated. The innovation points of this study are as follows:

1. A wheat growth stage detection model that uses depth-wise separable convolutional layers and a residual network is designed. It has low energy consumption and computing delay and high accuracy in distinguishing the seedling stage (SS), tillering stage (TS), overwintering stage (OS), greening stage (GS), and jointing stage (JS). The average recognition accuracy of the five wheat growth stages is 98.6%, whereas the DenseNet model achieves an average accuracy of 99.2%.
2. A dynamic migration algorithm for the wheat growth detection model is designed using the DQN. This algorithm makes optimal migration decisions by monitoring the power consumption and network service quality of the equipment in real-time, considering the energy consumption and delay cost caused by the migration/non-migration, respectively. At a wireless network transmission data rate of 0–8 MB/s,

the overall energy consumption loss of the dynamic migration algorithm is 128.4% lower than that of the intelligent device.

In this paper, an artificial intelligence algorithm and experiment are used to identify wheat growth stages. A decision-making method for performing edge computing and migrating the wheat growth stage detection model to the wireless network edge server for processing is proposed. The dynamic migration strategy of the DQN-based identification model enables the execution of complex processes while minimizing energy consumption and processing time. This method is suitable for deploying application systems in agriculture. This paper is organized as follows: Section 1 presents the introduction. Section 2 describes the materials and methods. Section 3 provides the wheat growth stage detection model, and Section 4 presents the migration algorithm. The results are described in Section 5, and Section 6 provides the discussion.

2. Materials and Methods

2.1. Data Source

The study area for acquiring the wheat images was the Xuchang campus of Henan Agricultural University, Changge City, Henan Province, China (113°58′26″ E, 34°12′06″ N). The area has a northern temperate continental monsoon climate, with an average annual temperature of 14.3 °C. The average annual rainfall is 711.1 mm, and the frost-free period is 217 days. Due to the complex field environment, images of the wheat canopy at a fixed height using a tripod were acquired. The images of the wheat varieties "Yumai49", "week 27", and "Xinong 509" were acquired in five growth stages (October 2019 to June 2020). These varieties are grown in the eastern Henan Province.

In each stage, images were acquired of plots with two densities (300 and 350 plants per square meter) and two nitrogen contents (15 kg and 0 kg of pure nitrogen per 0.0667 hectare). Images were obtained every two days between 8 am and 15 pm using a Nikon D3100, 5/21 sensor CMOS camera with a maximum aperture of F/5.6, 14.2 megapixels, and a maximum resolution of 4608 × 3072. A tripod was used for fixed-height photography, and all images were collected under natural lighting conditions.

Data were obtained in the following wheat growth stages: SS (the day before the first day of emergence to the tillering stage), TS (the day before the first day of tillering to the overwintering stage), OS (the day before the first day of the overwintering stage to the greening stage), GS (from the first day of the greening stage to the day before the jointing stage), and JS (from the first day of jointing to the day before heading) (Figure 1). A total of 12,000 images were obtained in the five stages.

2.2. Data Processing

Large sample sizes result in a higher performance and generalization ability of DL models. However, the number and quality of samples sometimes do not meet the requirements of optimal model training in practical applications; thus, the enhancement of sample data is required [27]. Images are high-dimensional data. Image data are typically rotated and translated, or other operations are performed to improve the robustness of the model, prevent overfitting of the test set during training, and improve the model's generalization ability. Data enhancement is a simple and effective method to improve the detection accuracy of convolutional neural network models. Different data sets require different data enhancement methods. Images are typically slightly modified, which does not affect the model's training results and can increase the generalization ability of the model. The following data enhancement methods were used to improve the model's robustness.

(1) Normalization by dividing each pixel value by the standard deviation of the sample;
(2) Dislocation transformation. The x-coordinate of the image remains unchanged, and the y-coordinate is shifted according to a specific proportion. The degree of displacement is proportional to the vertical distance to the x-axis;
(3) Image scaling. Image scaling refers to resizing the image by the same amount in the length and width directions;

(4) Random flipping. Random flipping refers to extracting image data and performing random flipping;
(5) Standardization. Standardization refers to an enhancement operation that the model performs on all images before training. Each pixel value is divided by 255 to obtain a pixel value range from 0 to 1. This method speeds up the convergence of the model.

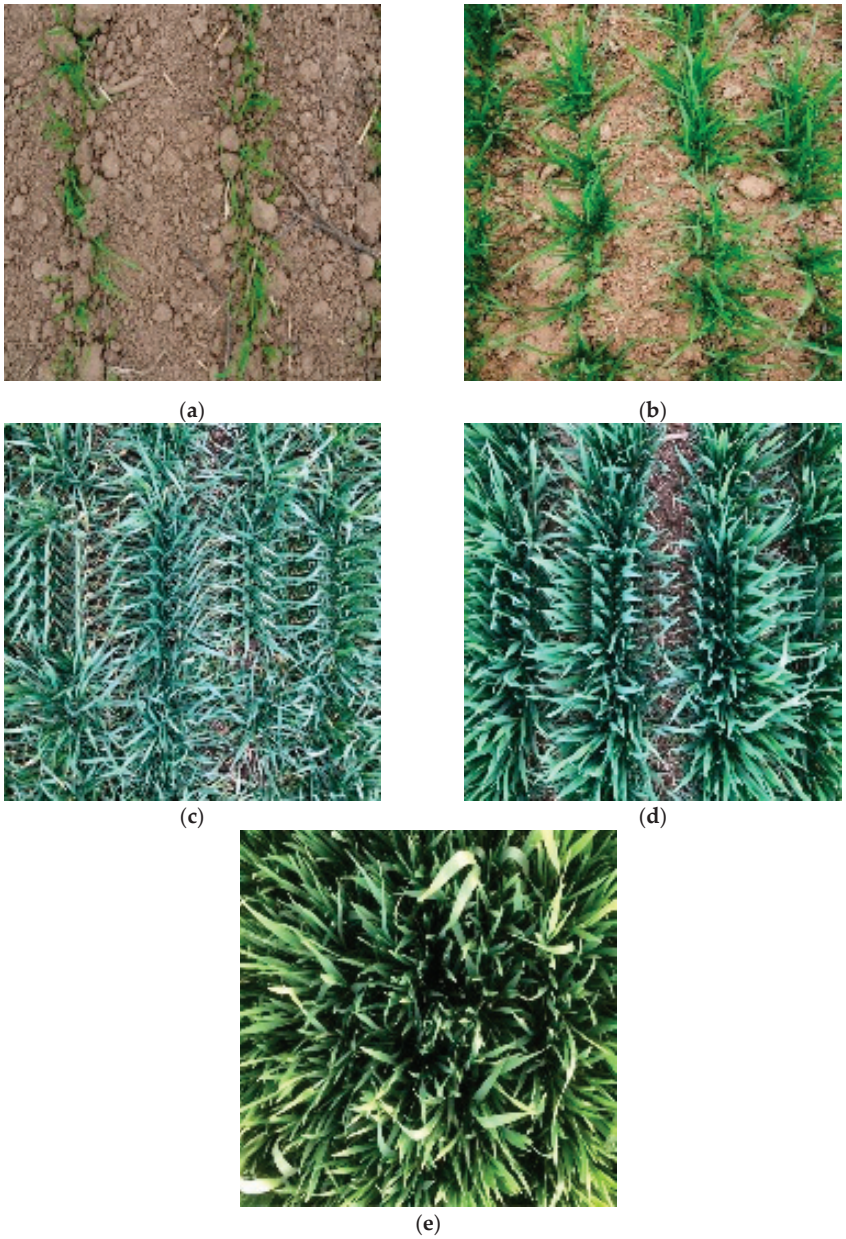


Figure 1. Wheat canopy images acquired in five stages: (**a**) seedling stage; (**b**) tillering stage; (**c**) overwintering stage; (**d**) greening stage; (**e**) jointing stage.

In the experiment, 80% of the images were randomly selected as the training set, and 20% were used as the test set. All comparative experiments in this study are conducted on this dataset. Figure 2 is an image of seedling emergence after the above image enhancement. Table 1 shows the number of images of the training set and test set in each growth stage of wheat.

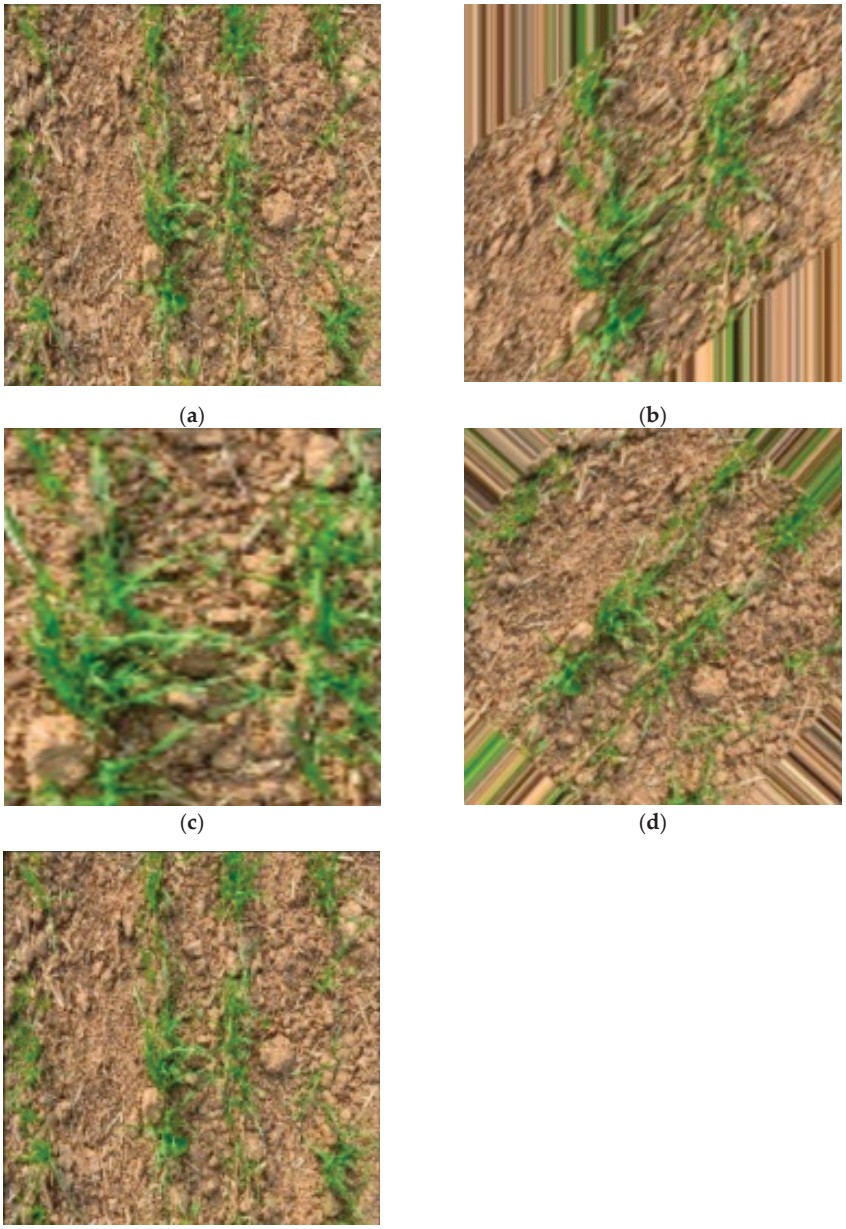


Figure 2. Data enhancement: (**a**) normalization; (**b**) dislocation transformation; (**c**) image scaling; (**d**) image flipping; (**e**) standardization.

Table 1. Number of wheat canopy image samples.

Wheat Growth Stages	Training Set/Piece	Test Sets/Piece	Total Sets/Piece
Seedling–tillering *	1920	480	2400
Tillering–overwintering	1920	480	2400
Overwintering–greening	1920	480	2400
Greening–jointing	1920	480	2400
Jointing–heading	1920	480	2400

* Seedling stage (SS) and tillering stage (TS).

3. Design of Wheat Growth Stage Detection Model

3.1. Framework of Wheat Growth Stage Detection Model

A lightweight recognition model based on depth-wise separable convolution [28] and a residual network [29] are proposed for use on intelligent mobile devices. The structure diagram of the convolutional neural network is shown in Figure 3. Conv2D, DSConv2D, and Conv2D-d represent the normal convolution, depth-wise separable convolution, and cavity convolution, respectively. A Relu6 activation function and a data standardization (batch normalization (BN)) operation are inserted after each convolution unit to ensure that the model can learn the sparse features of the wheat image and speed up its convergence. A linear activation function is used between the normal convolution and depth-wise separable convolution units to prevent gradient dispersion during model training. "Addition" in Figure 3 refers to the addition of the residual network. The residual network adds the outputs of the convolution units and uses them as the final output to achieve a greater model depth and prevent overfitting. The parameters of the network structure are listed in Table 2. The parameter input is the input of the current unit and the output of the upper unit. The parameters e and s1 represent the number and step size of the convolution kernels of the normal convolution, and the parameters O and s2 represent the number and step size of the convolution kernels of the depth-wise separable convolution. The parameter k is the size of the convolution kernel of the depth-wise separable convolution; it is 3×3 and 5×5. The parameter s indicates the presence of the residual network between the convolution units. A parameter value of d = 2 indicates that the normal convolution of the unit has been replaced by the void convolution. Softmax is the output function.

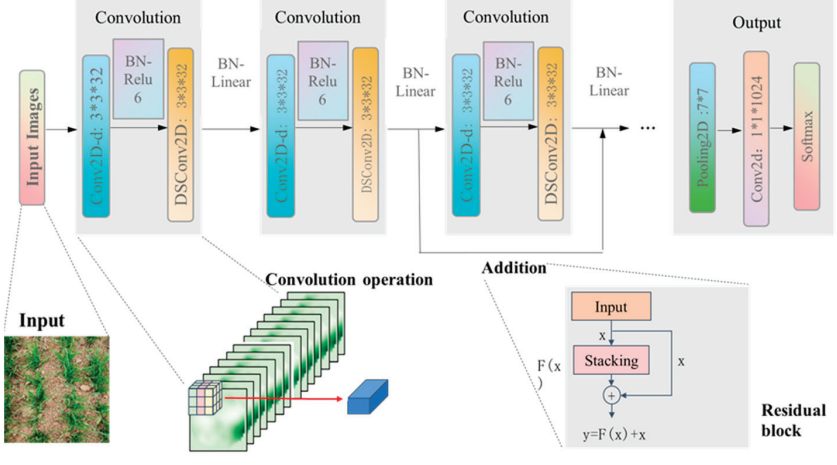


Figure 3. Structure of convolutional neural network.

Table 2. Network parameters of the wheat growth stage detection model.

Input Basic Unite	e	O	S1	S2	k	d	s
$224^2 \times 3$	Convolution32	32	1	2	3	2	false
$112^2 \times 32$	Convolution64	32	1	2	3	1	false
$56^2 \times 32$	Convolution128	32	1	1	3	1	true
$56^2 \times 32$	Convolution128	48	1	1	2	1	false
$56^2 \times 48$	Convolution196	48	2	1	3	1	false
$28^2 \times 48$	Convolution196	48	1	1	3	2	true
$28^2 \times 48$	Convolution256	64	1	1	5	1	false
$28^2 \times 64$	Convolution256	64	2	1	5	1	false
$14^2 \times 64$	Convolution400	64	1	1	5	1	false
$14^2 \times 64$	Convolution400	80	1	1	5	1	false
$14^2 \times 80$	Pooling2D (pool_size = 7, strides = 2)						
$4^2 \times 1024$	Conv2d 1×1(filters = 1024) Softmax						

3.2. Parameter Settings

The learning rate represents the speed of updating the model parameters during training, and the optimizer is a gradient descent updating method implemented during iteration. Different data sets have different learning rates and optimizer settings. Optimizing the hyperparameters improves the model's accuracy. The training batch represents the number of training images input into the model at each iteration. It is generally 32 and 64 batches in the image classification.

Canopy images of the five wheat growth stages were used: emergence, tillering, overwintering, greening, and jointing. There were 12,000 samples, including 2400 samples in each stage. The test set comprised 20% of the data, and the training set contained 80% of the data for model training and learning. Table 3 lists the results of the different learning rates, training batches, and optimizer training approaches. The optimization algorithms are the Adam optimizer and stochastic gradient descent (SGD) method, and 32 and 64 are used as the number of training batches. Adam-32 shows the training results of the Adam optimizer with a batch size of 32; 0.005, 0.001, 0.0005, and 0.0001 are the test values for the learning rate. The model achieves the highest accuracy when the learning rate is 0.001 and the Adam-32 optimizer is used. The accuracy is higher for 32 than for 64 training batches. Therefore, the Adam optimization algorithm with a learning rate of 0.001 and 32 batches was selected to train the wheat growth stage detection model.

Table 3. Comparison of hyperparameters.

Learning Rate	Adam-32(%)	Adam-64(%)	SGD-32(%)	SGD-64(%)
0.005	97.8	97.3	97.2	97.8
0.001	98.6	97.9	98.0	97.5
0.0005	97.9	96.1	96.9	97.3
0.0001	97.8	97.5	97.8	97.2

4. Design of Migration Algorithm

The proposed wheat growth stage detection model has a low battery energy consumption and delay. However, there is a need for intensive computing to perform intelligent fault monitoring in smart agriculture. When there are many computing tasks, moving them to the edge server improves crop monitoring efficiency. However, the dynamic changes

in the computing scenarios and the wireless network quality of the service may result in inadequate performance when tasks are executed at the edge. Therefore, intelligent mobile devices must dynamically decide whether to offload computing tasks to the edge of the network. When the wireless network transmission rate is high and the intelligent device has sufficient power, it is suitable to unload the task to the edge server, resulting in high performance. In contrast, when the wireless network transmission rate is low and the device power is insufficient, the task cannot be moved to the edge for processing. However, it is often impossible in real scenarios to determine whether task unloading is required due to the dynamic changes in the computing environment and the wireless network's quality of service.

4.1. Design for Dynamic Migration Algorithm with a Mobile Terminal

The residual power of mobile devices is a valuable energy resource in the migration of computing services to the mobile edge. In addition to variable factors, such as the dynamic characteristics of the mobile device's environment, especially the network conditions, many factors determine the migration decision of mobile devices. The strong perception of DRL can be used to learn the state information of the environment and modify the decision-making so that mobile users can complete the computing task at the lowest cost. The DQN is an unsupervised neural network learning algorithm based on reinforcement learning. It combines the learning ability of a neural network and the decision-making ability of reinforcement learning and makes intelligent decisions in a timely manner according to the changing environment [30]. The proposed dynamic migration algorithm makes the optimal decision by monitoring the power and wireless network speed of the device in real-time, considering the energy consumption and delay cost caused by the unloading/non-unloading decision, minimizing the calculation delay and power consumption.

$\varphi(S)$ is used as the input of the DQN. The greedy method is used to make random selections of an action selection to prevent the network from falling into a local minimum. Figure 4 shows the flowchart of the algorithm. The DRL model considers five key factors [14]: the environment, agent, action, status, rewards, and penalties.

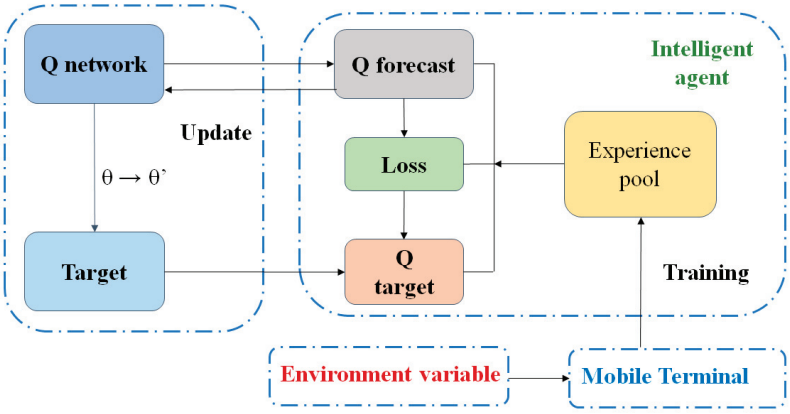


Figure 4. Flowchart of the DQN algorithm.

The following equation expresses the DRL model:

$$y_j = \begin{cases} R_j is\_end = true \\ R_j + \gamma max_{a'} Q(\varphi(S'_j), A'_j, w) is\_end = false \end{cases} \quad (1)$$

$\varphi(S)$ is the input of the deep Q-learning network. A greedy method is used to obtain the Q value. It uses a random selection to prevent the network from falling into the local optimum. The current action, A, in the state, S, is executed to obtain the feature vector

corresponding to the new state S′ with $\varphi(S\prime)$ and reward R to terminate the status, *is_end*. $\{\varphi(S), A, R, \varphi(S'), is\_end\}$ is used as the parameters in the experience pool. The agent obtains the experience value to learn the current Q value for y_j.

4.2. Energy Consumption and Calculation Delay of Wheat Growth Stage Detection Model

A mathematical equation was established to calculate the energy consumption and delay of the wheat growth stage detection model. The processing information of the mobile device is represented as a quaternion, M_i= (c_w,u_w,d_w,f_s), where c_w is the CPU power of the mobile device, u_w and d_w are the power of the mobile device to upload and download data, respectively, and f_s is the number of floating-point operations per second. The wireless network status is represented as a binary group, S_i = (u_s,d_s), where u_s represents the upload speed, and d_s represents the download speed of the wireless network. The decision space is defined as xi = 0 and xi = 1, where "0" denotes the task is processed on the intelligent device, and "1" denotes the task is unloaded to the edge server for processing. The delay includes the calculation delay and communication delay, when xi = 0, T_m represents the calculation delay of the mobile device, and when xi = 1, T_m represents the calculation delay of the edge server. The communication delay is represented by T_s, as shown in Equations (2) and (3):

$$T_m = \frac{F_l}{f_s} \tag{2}$$

$$T_s = \frac{P_{size}}{u_s} + \frac{P_{result}}{d_s} \tag{3}$$

where F_l represents the floating-point number required by the mobile device's CPU to complete the computing tasks, and P_{size} and P_{result} represent the size of the uploaded and received data, respectively. The energy consumption consists of the computing energy consumption and communication energy consumption; only the energy consumption of the mobile device is considered. The computing energy consumption and communication energy consumption are calculated by Equations (4) and (5), respectively.

$$E_m = c_w \times \frac{F_l}{f_s} \tag{4}$$

$$E_s = u_w \times \frac{P_{size}}{u_s} + d_w \times \frac{P_{result}}{d_s} \tag{5}$$

4.3. Design of Agent

After defining the energy consumption and time delay, it is necessary to determine the agent's learning ability to evaluate the two parameters and decide whether to migrate the services. The DQN evaluates the energy consumption and time delay dynamically. The weight of the energy consumption is small if the mobile devices have more residual power and vice versa, regardless of whether the services are migrated or not. Similarly, time delay also has a weight parameter. Figure 5 shows the structure of the agent. During the training of the DQN algorithm, the agent learns useful information as the environment changes. The agent is used to simulate the decision-making and calculation processes of intelligent devices. After the agent inputs the network and electricity status into the neural network, it calculates the energy consumption and time delay of the decision results and evaluates the decision quality to assess the rewards and penalties. Because the input consists of only two parameters (the network speed and power), the agent uses a small back propagation (BP) neural network to simulate the decision-making of intelligent devices. Figure 5 shows that the BP neural network for decision-making has four hidden layers, and the activation function is a leaky ReLU function. The decision-making results are obtained by inputting the network speed and power, and the agent learns using the reinforcement learning algorithm. The calculation of the energy consumption and time delay is expressed

by Equations (6) and (7), which are combined into Equation (8) to optimize the time delay and energy consumption jointly.

$$A(s_i, a_i) = k_t \times T_i + k_e \times E_i \tag{6}$$

$$T_i = \min_{x_i}\left(\frac{F_l}{f_s} + x_i\left(\frac{P_{size}}{u_s} + \frac{P_{result}}{d_s}\right)\right) \tag{7}$$

$$E_i = \min_{x_i}\left((1 - x_i) \times c_w \times \frac{F_l}{f_s} + x_i\left(u_w \times \frac{P_{size}}{u_s} + d_w \times \frac{P_{result}}{d_s}\right)\right) \tag{8}$$

where T_i and I represent the delay and energy consumption costs after the agent has made a decision, and $A(s_i, a_i)$ represents the weighted sum of the energy consumption and costs. k_t and k_e are the delay and energy consumption coefficients, indicating the importance of the delay and energy consumption. When the power is low, the energy consumption coefficient, k_e, is high, and when the network speed is high, the delay coefficient, k_t, is high.

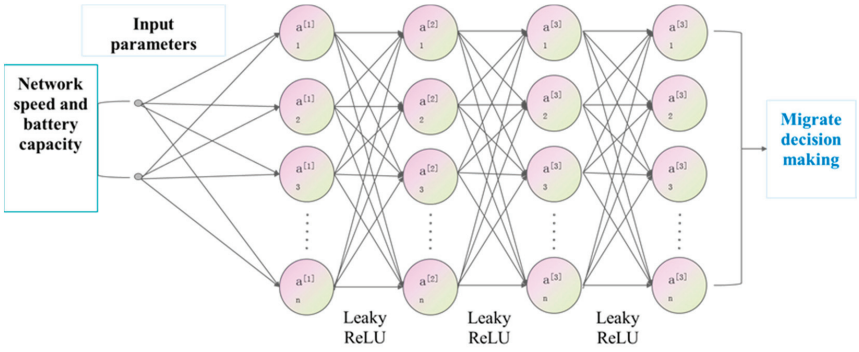


Figure 5. Agent structure.

5. Experimental Design and Results

5.1. Experimental Results of Lightweight Detection Model

The VGG16, ResNet50, InceptionV3, MobileNetV2, and DenseNet models were compared with the proposed lightweight wheat growth stage model. These classic models have achieved good results in many fields. The experimental environment and the hyperparameters were consistent for all of the models, and training was conducted locally using the Tensorflow framework [31]. The graphics card was a GTX1050 Ti. A 0.001 learning rate, and the Adam optimizer was used for training. The effect of the network structure on the detection performance was compared. The accuracy rate change in each epoch during training was recorded to compare the models' learning abilities. Only the accuracy rate change of the first 30 epochs is shown because all the models have a high learning ability. The performances of the different models for detecting the wheat growth stages are listed in Table 4.

The results indicate that the proposed model has a higher accuracy rate than the other models in the GS. Because the GS is difficult to identify, the accuracy rate is slightly higher than in the other growth stages. The average recognition accuracy of the five growth stages is 98.6% for the proposed model and 99.2% for DenseNet, which achieved the highest average accuracy.

Table 4. Performance of different models for detecting the wheat growth stages.

Model	JS (%) *	ES (%) *	GS (%) *	TS (%) *	OS (%) *	Average (%)
VGG16	99.2	100	94.6	96.0	97.3	97.8
Inception	99.4	99.6	93.1	100	97.4	97.9
ResNet50	99.6	99.2	94.2	99.8	98.2	98.2
Mobile Net	99.6	100	96.0	99.4	98.0	98.6
Dense Net	99.6	100	97.9	99.8	98.6	99.2
Proposed model	99.4	98.6	98.0	99.2	97.8	98.6

* JS: jointing stage; ES: emergence stage; GS: greening stage; TS: tillering stage; OS: overwintering stage.

5.2. Experimental Results of Deep Reinforcement Learning Recognition Model and Dynamic Migration Algorithm

5.2.1. Comparison of the Models' Operating Speeds

The model's operating speed is critical because it runs on a mobile terminal. A speed test was conducted using 100 wheat growth stage images to evaluate the performances of the models. Table 5 lists the results. The results show that the detection speed of the models does not increase with a decrease in the parameter number but is related to the model's structure. This effect is the most pronounced for the VGG because it has a relatively simple structure despite its many parameters; therefore, it has a fast detection speed. Although the DenseNet model has few parameters, its structure is complex, resulting in a large number of feature maps and low detection speed. The size of the proposed wheat growth stage detection model is only 1.3 MB. Thus, it has the highest detection speed due to the low parameter number. The parameter number of the proposed model is 58% lower, and its detection speed is 47% higher than that of MobileNetV2.

Table 5. Operating speeds of different models.

ModelVGG16	IV3 *	RT50 *	MT2 *	DT *	Proposed Model
Time(s) 32.88	163.09	116.81	84.88	212.16	45.07
Parameter (MB)134.3	21.8	27.9	3.1	7.0	1.3
Parameter (MB)134.3	21.8	27.9	3.1	7.0	1.3

* IV3: InceptionV3; RT50: Resnet50; MT2: MobileNetV2; DT: DenseNet.

5.2.2. Impact of Learning Rate and Experience Pool on Loss

The mobile device uses a Core i5-10500 processor, 8G (DDR43000) of memory, and no GPU acceleration. The edge computing server uses the Tencent lightweight server, CentOS7 system, and 2G memory, and the maximum bandwidth is 5 Mbps. Different data transmission rates were selected according to the wireless network communication mode [32]. The TensorFlow service's framework was used to deploy the model to the Linux server. The loss value was utilized to evaluate the error between the real and predicted values [33,34]. The change in the learning rate significantly affects the loss value of the DQN algorithm. Thus, the models with learning rates of 0.01, 0.001, and 0.0001 were assessed for 200 iterations. Figure 6 shows that when the experience pool is 500, the loss value fluctuates significantly with an increase in the epoch number when the experience pool is 500 and stabilizes at 2000. Therefore, a value of 2000 was used to store the decision data.

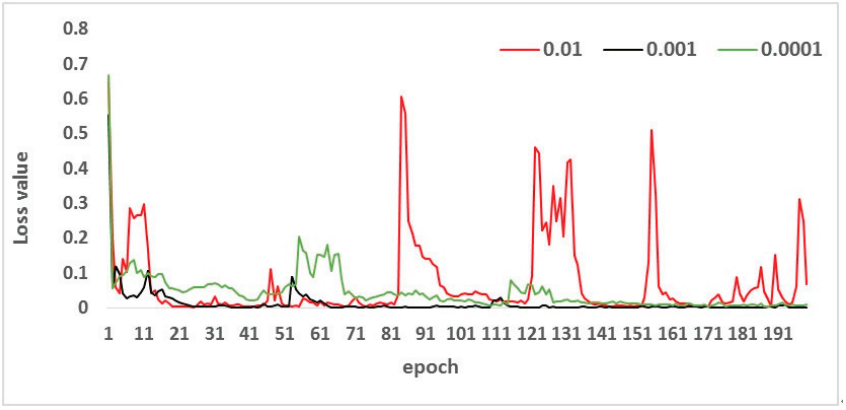


Figure 6. Impact of experience pool on loss value.

5.2.3. Energy Consumption and Delay

The gradient descent method is used to minimize the energy consumption and delay ($A(s_i, a_i)$). The values of k_t and k_e change with a change in the power and network speed. When the power is sufficient, the agent's learning strategy ensures that the delay is minimized, and when the network's speed is sufficient, the energy consumption is minimized. The time delay and energy consumption coefficients, k_t and k_e, at different network speeds are shown in Figure 7. When the coefficient, k_t, of the network speed exceeds 75%, the energy consumption coefficient remains unchanged, the delay coefficient increases, and the delay is reduced.

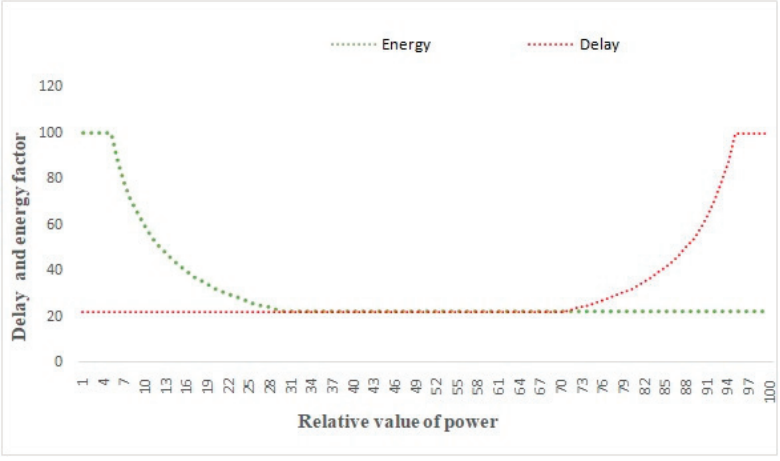


Figure 7. Time delay and energy consumption factors at different power values.

Energy consumption and time delay are critical parameters of migration decisions when mobile devices are used. The reinforcement learning algorithm continuously learns from the energy consumption and time delay resulting from each decision to minimize these parameters. Table 6 shows the energy consumption and delay for the different models. The proposed model has fewer parameters, a faster running speed, and lower energy consumption than the other detection models. The speed of performing the detection on one image on an intelligent device is 0.43 s, and the energy consumption is 0.023 mWh. These values are 49% lower than that of MobileNetV2 (MT2).

Table 6. Comparison of energy consumption and delay for different models.

Model IV3	RT50	DT	MT2	Proposed Method
Data (MB) 21.8 *	27.9	7.0	3.1	1.3
Delay (s) 1.63	1.16	2.12	0.84	0.43
Energy (mWh) 0.091	0.064	0.118	0.045	0.023

* IV3: InceptionV3; RT50: Resnet50; DT: DenseNet; MT2: MobileNetV2.

The energy consumption and delay of the proposed method at the mobile terminal and edge server are listed in Table 7.

Table 7. Energy consumption and delay of the proposed method.

Proposed Model	Data	Layers	Accuracy	Delay	Energy
Value	1.3 MB	21	98.6%	0.43 s	0.077 mWh

Experiments were conducted on performing and not performing decision-making to evaluate the effect of the DQN algorithm on the intelligent migration of the convolutional neural network model. Not performing decision-making was divided into execution on the device (local execution) and execution in the cloud (edge execution). The average operation times and average delay of the system were analyzed at the same power. Figure 8 shows the average running times of the model at different network speeds. The higher the average running time, the lower the energy consumption. At a network speed of 0–2 MB/s, the energy consumption is high, and the model decision is biased toward local execution because the network speed is low and the transmission time is long. However, as the network speed increases, the energy consumption is higher for local execution than for migration to the cloud; thus, cloud execution is preferable. The energy consumption of the intelligent migration algorithm is 128.4% lower than that of local execution at a network speed of 0–8 MB/s.

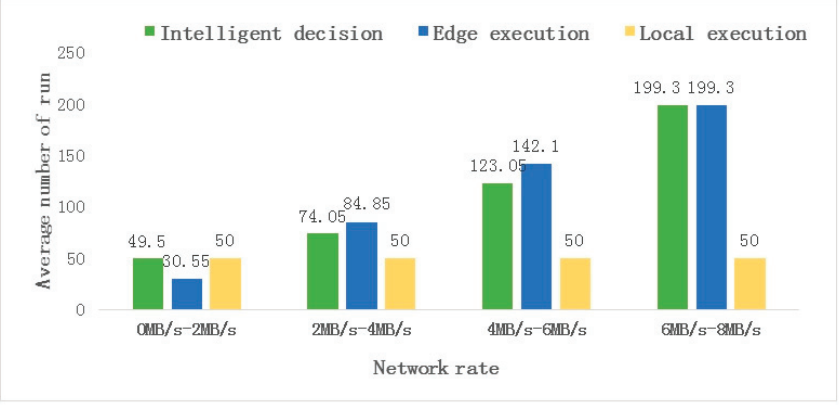


Figure 8. Average running times.

Figure 9 shows the average delay for the different network rates. The delay of edge execution is the highest at a network speed of 0–2 MB/s, and local execution is preferable. As the network speed increases, the network communication delay decreases, and edge execution becomes preferable. The average efficiency of the intelligent migration algorithm is 121.2% higher than the local execution at a network rate of 0–8 MB/s.

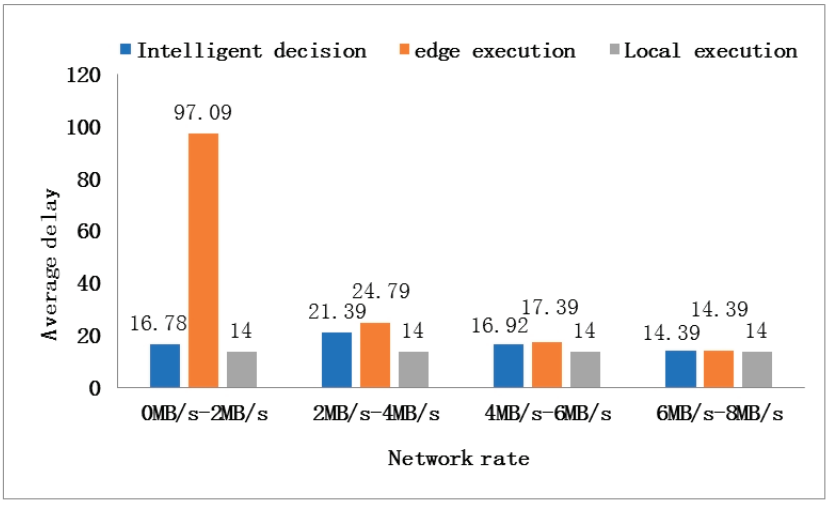


Figure 9. Average delay.

6. Discussion

Implementing a deep learning algorithm for wheat growth stage detection on mobile devices has high energy consumption and a large time delay. A lightweight detection model was proposed with low energy consumption and delay based on depth-wise separable convolution and a residual wireless network. A decision-making method was proposed for performing edge computing and migrating the wheat growth stage detection model to the wireless network edge server for processing. The dynamic migration strategy of the DQN-based identification model enabled the execution of complex processes while minimizing energy consumption and processing time. The proposed method is also applicable to other crops.

The experimental results show that the proposed model and algorithm have good performance and are suitable for practical applications. This approach can be used to develop a wheat growth period monitoring system. It can be implemented on mobile devices, and the calculations are performed on the server. The TensorFlowlite open-source framework can be used to implement this model on mobile devices. On the server side, Docker can be used to deploy the model server to execute requests and return the result to the mobile device.

Author Contributions: Formal analysis, X.M.; Investigation, J.W.; Methodology, Y.L.; Resources, G.Z.; Supervision, L.X.; Validation, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 61871322, and the Henan Province Key Scientific and Technological Project, grant number 222102110234.

Institutional Review Board Statement: This study does not require ethical approval, and we choose to exclude this statement.

Data Availability Statement: We will consider analyzing the research data in some way in http://en.henau.edu.cn/.

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

References

1. Hellegers, P. Food security vulnerability due to trade dependencies on Russia and Ukraine. *Food Secur.* **2022**, *14*, 1503–1510. [CrossRef]
2. Han, S.; Zhao, Y.; Cheng, J.; Zhao, F.; Yang, H.; Feng, H.; Li, Z.; Ma, X.; Zhao, C.; Yang, G. Monitoring Key Wheat Growth Variables by Integrating Phenology and UAV Multispectral Imagery Data into Random Forest Model. *Remote Sens.* **2022**, *14*, 3723. [CrossRef]
3. Ren, S.; Guo, B.; Wu, X.; Zhang, L.; Ji, M.; Wang, J. Winter wheat planted area monitoring and yield modeling using MODIS data in the Huang-Huai-Hai Plain, China. *Comput. Electron. Agric.* **2021**, *182*, 106049. [CrossRef]
4. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
5. Zhang, Y.; Wang, W.; Zhang, P.; Huang, P. Reinforcement-Learning-Based Task Planning for Self-Reconfiguration of Cellular Satellites. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *37*, 38–47. [CrossRef]
6. Hassan, N.; Yau, K.L.A.; Wu, C. Edge computing in 5G: A review. *IEEE Access* **2019**, *7*, 127276–127289. [CrossRef]
7. Clifton, J.; Laber, E. Q-learning: Theory and applications. *Annu. Rev. Stat. Its Appl.* **2020**, *7*, 279–301. [CrossRef]
8. Li, Y.; Jiang, C. Distributed task offloading strategy to low load base stations in mobile edge computing environment. *Comput. Commun.* **2020**, *164*, 240–248. [CrossRef]
9. Chen, C.; Chen, L.; Liu, L.; He, S.; Yuan, X.; Lan, D.; Chen, Z. Delay-optimized V2V-based computation offloading in urban vehicular edge computing and networks. *IEEE Access* **2020**, *8*, 18863–18873. [CrossRef]
10. Xiao, Z.; Chen, Y.; Jiang, H.; Hu, Z.; Lui, J.C.; Min, G.; Dustdar, S. Resource management in UAV-assisted MEC: State-of-the-art and open challenges. *Wirel. Netw.* **2022**, *28*, 3305–3322. [CrossRef]
11. Shen, H.; Jiang, Y.; Deng, F.; Shan, Y. Task Unloading Strategy of Multi UAV for Transmission Line Inspection Based on Deep Reinforcement Learning. *Electronics* **2022**, *11*, 2188. [CrossRef]
12. Lv, Z.; Chen, D.; Wang, Q. Diversified technologies in internet of vehicles under intelligent edge computing. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 2048–2059. [CrossRef]
13. Wang, K.; Wang, X.; Liu, X. A high reliable computing offloading strategy using deep reinforcement learning for iovs in edge computing. *J. Grid Comput.* **2021**, *19*, 15. [CrossRef]
14. Ding, X.; Zhang, W. Computing unloading strategy of massive internet of things devices based on game theory in mobile edge computing. *Math. Probl. Eng.* **2021**, *2021*, 2163965. [CrossRef]
15. Huang, J.; Gao, H.; Wan, S.; Chen, Y. AoI-aware energy control and computation offloading for industrial IoT. *Future Gener. Comput. Syst.* **2023**, *139*, 29–37. [CrossRef]
16. Chen, X.; Jiao, L.; Li, W.; Fu, X. Efficient multi-user computation offloading for mobile-edge cloud computing. *IEEE/ACM Trans. Netw.* **2015**, *24*, 2795–2808. [CrossRef]
17. Zhang, D.; Cao, L.; Zhu, H.; Zhang, T.; Du, J.; Jiang, K. Task offloading method of edge computing in internet of vehicles based on deep reinforcement learning. *Cluster Comput.* **2022**, *25*, 1175–1187. [CrossRef]
18. Chen, C.; Zhang, Y.; Wang, Z.; Wan, S.; Pei, Q. Distributed computation offloading method based on deep reinforcement learning in ICV. *Appl. Soft Comput.* **2021**, *103*, 107108. [CrossRef]
19. Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; Li, Y. Computer vision technology in agricultural automation—A review. *Inf. Process. Agric.* **2020**, *7*, 1–19. [CrossRef]
20. Zhang, Z.J.; Wu, T.; Li, Z.; Shen, B.; Chen, N.; Li, J. Research of offloading decision and resource scheduling in edge computing based on deep reinforcement learning. In Proceedings of the Smart Grid and Internet of Things: 4th EAI International Conference, SGIoT 2020, TaiChung, Taiwan, 5–6 December 2020.
21. Gu, M.; Li, K.C.; Li, Z.; Han, Q.; Fan, W. Recognition of crop diseases based on depthwise separable convolution in edge computing. *Sensors* **2020**, *20*, 4091. [CrossRef]
22. Sun, L.; Zhao, H.; Chen, J. Recognition method of crop diseases and insect pests based on multi-layer feature fusion. *Basic Clin. Pharmacol. Toxicol.* **2020**, *2020*, 127.
23. Albanese, A.; Nardello, M.; Brunelli, D. Automated pest detection with DNN on the edge for precision agriculture. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2021**, *11*, 458–467. [CrossRef]
24. Zhou, G.; Wen, R.; Tian, W.; Buyya, R. Deep reinforcement learning-based algorithms selectors for the resource scheduling in hierarchical Cloud computing. *J. Netw. Comput. Appl.* **2022**, *208*, 103520. [CrossRef]
25. Weichman, P.B. Quantum-enhanced algorithms for classical target detection in complex environments. *Phys. Rev.* **2021**, *103*, 042424. [CrossRef]
26. Ji, B.; Wang, Y.; Song, K.; Li, C.; Wen, H.; Menon, V.G.; Mumtaz, S. A survey of computational intelligence for 6G: Key technologies, applications and trends. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7145–7154. [CrossRef]
27. Peng, X.; Zhang, X.; Li, Y.; Liu, B. Research on image feature extraction and retrieval algorithms based on convolutional neural network. *J. Vis. Commun. Image Represent.* **2020**, *69*, 102705. [CrossRef]
28. Yun, J.; Jiang, D.; Liu, Y.; Sun, Y.; Tao, B.; Kong, J.; Tian, J.; Tong, X.; Xu, M.; Fang, Z. Real-Time Target Detection Method Based on Lightweight Convolutional Neural Network. *Front. Bioeng. Biotechnol.* **2022**, *10*, 861286. [CrossRef]
29. Lochbihler, A. A mechanized proof of the max-flow min-cut theorem for countable networks with applications to probability theory. *J. Autom. Reason.* **2022**, *66*, 585–610. [CrossRef]

30. Yang, Y.; Juntao, L.; Lingling, P. Multi-robot path planning based on a deep reinforcement learning DQN algorithm. *CAAI Trans. Intell. Technol.* **2020**, *5*, 177–183. [CrossRef]
31. Haghighat, E.; Juanes, R. SciANN: A Keras/TensorFlow wrapper for scientific computations and physics-informed deep learning using artificial neural networks. *Comput. Methods Appl. Mech. Eng.* **2021**, *373*, 113552. [CrossRef]
32. Li, Y.; Li, B.; Yang, M.; Yan, Z. A spatial clustering group division-based OFDMA access protocol for the next generation WLAN. *Wirel. Netw.* **2019**, *25*, 5083–5097. [CrossRef]
33. Yang, H.B.; Zhao, J.; Lan, Y.; Lu, L.; Li, Z. Fraction vegetation cover extraction of winter wheat based on spectral information and texture features obtained by UAV. *Int. J. Precis. Agric. Aviat.* **2019**, *2*, 54–61.
34. Čirjak, D.; Aleksi, I.; Miklečić, I.; Antolković, A.M.; Vrtodušić, R.; Viduka, A.; Lemic, D.; Kos, T.; Pajač Živković, I. Monitoring System for *Leucoptera malifoliella* (O. Costa, 1836) and Its Damage Based on Artificial Neural Networks. *Agriculture* **2023**, *13*, 67. [CrossRef]

agriculture

MDPI

Article

Global Reconstruction Method of Maize Population at Seedling Stage Based on Kinect Sensor

Naimin Xu [1], Guoxiang Sun [1,2,*], Yuhao Bai [1], Xinzhu Zhou [1], Jiaqi Cai [1] and Yinfeng Huang [1]

[1] College of Engineering, Nanjing Agricultural University, Nanjing 210095, China
[2] Jiangsu Province Engineering Lab for Modern Facility Agriculture Technology & Equipment, Nanjing 210031, China
* Correspondence: sguoxiang@njau.edu.cn; Tel.: +86-255-860-6585

Abstract: Automatic plant phenotype measurement technology based on the rapid and accurate reconstruction of maize structures at the seedling stage is essential for the early variety selection, cultivation, and scientific management of maize. Manual measurement is time-consuming, laborious, and error-prone. The lack of mobility of large equipment in the field make the high-throughput detection of maize plant phenotypes challenging. Therefore, a global 3D reconstruction algorithm was proposed for the high-throughput detection of maize phenotypic traits. First, a self-propelled mobile platform was used to automatically collect three-dimensional point clouds of maize seedling populations from multiple measurement points and perspectives. Second, the Harris corner detection algorithm and singular value decomposition (SVD) were used for the pre-calibration single measurement point multi-view alignment matrix. Finally, the multi-view registration algorithm and iterative nearest point algorithm (ICP) were used for the global 3D reconstruction of the maize seedling population. The results showed that the R^2 of the plant height and maximum width measured by the global 3D reconstruction of the seedling maize population were 0.98 and 0.99 with RMSE of 1.39 cm and 1.45 cm and mean absolute percentage errors (MAPEs) of 1.92% and 2.29%, respectively. For the standard sphere, the percentage of the Hausdorff distance set of reconstruction point clouds less than 0.5 cm was 55.26%, and the percentage was 76.88% for those less than 0.8 cm. The method proposed in this study provides a reference for the global reconstruction and phenotypic measurement of crop populations at the seedling stage, which aids in the early management of maize with precision and intelligence.

Keywords: Kinect; crop phenotypic; point cloud processing; three-dimensional reconstruction; singular value decomposition

Citation: Xu, N.; Sun, G.; Bai, Y.; Zhou, X.; Cai, J.; Huang, Y. Global Reconstruction Method of Maize Population at Seedling Stage Based on Kinect Sensor. *Agriculture* **2023**, *13*, 348. https://doi.org/10.3390/agriculture13020348

Academic Editor: Wen-Hao Su

Received: 31 December 2022
Revised: 26 January 2023
Accepted: 28 January 2023
Published: 31 January 2023

1. Introduction

The term "crop phenotype" describes the physical, physiological, and biochemical traits representing the structural and functional traits of crop cells, tissues, plants, and populations. The accurate and intelligent administration of modern agriculture depends critically on the phenotypic information [1,2]. Historically, crop phenotypic measurements have been performed manually. These approaches are damaging, extremely subjective, ineffective, and inappropriate for modern agricultural precision management [3]. Crop phenotypic assessment tends to exhibit high-throughput, high-precision, and automation owing to the rapid development of technologies like machine vision, agricultural robots, and artificial intelligence [4,5]. Currently, crop phenotype measurements are primarily based on 2D images and 3D point cloud techniques [6,7]. Nevertheless, owing to the complex nature of the plant morphology and mutual occlusion between leaves, 2D image-based and single-viewpoint 3D point cloud phenotyping techniques cannot accurately assess plant phenotypic data [8]. Consequently, the creation of 3D plant models using computer vision techniques for the precise and effective extraction of plant phenotypic

features has steadily grown to become a prominent research area in the field of crop phenotyping [9–13].

The 3D reconstruction approaches were divided into two categories based on active and passive vision. Techniques for functional vision-based 3D reconstruction include time-of-flight (TOF), laser scanning, and structured light. These techniques primarily employ visual tools to gather object surface information and perform 3D reconstruction. Monocular vision, binocular vision, and multi-visual vision methods are examples of passive-vision-based 3D reconstruction techniques. These methods primarily capture image sequences using visual sensors and subsequently achieve three-dimensional (3D) reconstruction.

The primary sensors used for 3D point cloud reconstruction are LIDAR, CT scanners, hyperspectral imagers, depth cameras, and RGB cameras. LIDAR has a low reconstruction efficiency, making it best suited for navigation and large-scale scene reconstruction. It cannot be used to reconstruct 3D point cloud models for smaller plants [14,15]. The CT scanner, which is mainly used for medical imaging, emits radiation. Environmental interference, sluggish imaging speed, and small measurement areas affect hyperspectral imagers. Owing to their extreme precision and low cost, vision sensors have been extensively utilized in agriculture in recent years [16]. Researchers from home and abroad have gradually replaced costly LIDAR in the study of the 3D reconstruction of crop phenotypes using visual sensors such as depth and RGB cameras. Peng et al. employed the iterative nearest-point approach for fine alignment to rebuild the 3D point cloud of tomato plants [17] based on the data collected by the KinectV2 depth camera to calculate the coarse alignment matrix from the end joint positions captured by the robotic arm. Using a reflecting single-frame camera and a multi-view stereo vision algorithm, Hu et al. reconstructed the structures of green pepper and eggplants in three dimensions [18]. He et al. employed the structure from motion (SFM) technique to create a 3D model of a strawberry based on an SLR camera [19]. Although there have been numerous studies on the 3D point cloud model reconstruction of crops, all of which have demonstrated high reconstruction accuracy and stable performance, most studies only reconstruct single plants, making it impossible to achieve global 3D reconstruction of crop populations. The global 3D reconstruction and measurement of crop populations remain challenging because of the complexity and unstructured nature of agricultural landscapes.

In this study, we developed a self-propelled crop phenotype measurement tool based on the ROS (Robot Operating System) mobile platform, combining the Harris corner point detection algorithm, singular value decomposition method, multiple measurement point alignment algorithm, and multiple filtering algorithms to achieve a global 3D reconstruction of the maize plant population. A reference for the global reconstruction and phenotypic assessment of maize populations at the seedling stage was provided by the methodology proposed in this study, which aids in the precise and careful management of maize in its early phases.

2. Materials and Methods

2.1. Experimental Data Collection

From 28 May to 27 June 2022, a maize population reconstruction experiment was conducted as part of this study at the Nanjing Agricultural University's Pukou Campus. Maize seedlings were chosen as test objects. There were 16 plants in total, and the variety chosen was Zhenzhennuo 99. The row and column spacing of a single maize plant was 80 cm and the average initial plant height was approximately 30 cm; 96 sets of maize plant point cloud data were collected every five days. The test setup is illustrated in Figure 1.

Figure 1. Physical picture of test scene. (**a**). Collection device; (**b**). crop plants.

2.2. Subsection Structure and Principles of Measurement Systems

The key components of the self-propelled crop phenotyping system were a Kinect-based mobile ROS platform, motorized rotating table, control cabinet, graphics workstation, and mobile power supply. A schematic of the hardware of the measurement system is shown in Figure 2. The system was built using a 20 × 20 mm aluminum structure, 40 cm long, 40 cm wide, and 140 cm high. A 2.0 version of the Kinect sensor, mainly composed of color and depth cameras, was used. The resolution of the color camera was 1920 × 1080 pixels, and that of the depth camera was 512 × 424 pixels. The measurement distance was 0.5–4.5 m and the viewing area angle was 70 × 60° (horizontal × vertical). The electric rotary table was the TBR 100 series with a table size of 102 mm, angle range of 360°, worm gear ratio of 180:1, using a 42 M-1.8 D-C-10 stepper motor, whole step resolution of 0.01°, positioning accuracy of 0.05°, a circuit control cabinet built-in motion control data acquisition card, driver, and switching power supply. The motion control data acquisition card model was the NET6043-S2XE built-in 10/100 M adaptive Ethernet card, 8-way 16-bit positive and negative 10 V range single-ended analog synchronous acquisition, up to 40 KSPS. The 8-way analog can be synchronized with a two-axis logic position or encoder high-speed synchronous acquisition and two-axis stepper/servo motor control. The driver model is AQMD3610NS-A2, which supports analog signals of 0–5/10 V, signal ports that can withstand voltages of 24 V, and standard mode voltage protection of 485. The DashgoB1 series is an intelligent mobile platform that includes navigation, map construction, obstacle avoidance, and other features. It has an STM 32 chassis controller, built-in LIDAR, ultrasonic radar, and wheel speed encoder. A shock absorber was mounted on the front side of the chassis, and the bottom shell was mounted underneath the shock absorber. For the cart to adjust to an uneven path, the bottom shell also has a stabilizer bar inside it. The Intel(R) Xeon(R) E-2176M CPU @2.70 GHz, 32 G of memory and an NVIDIA Quadro P600 4G graphics card comprised the graphics workstation processor. For the Ubuntu 18.04 system, MATLAB R2019 programming was used as the programming environment for the hardware and software of the system.

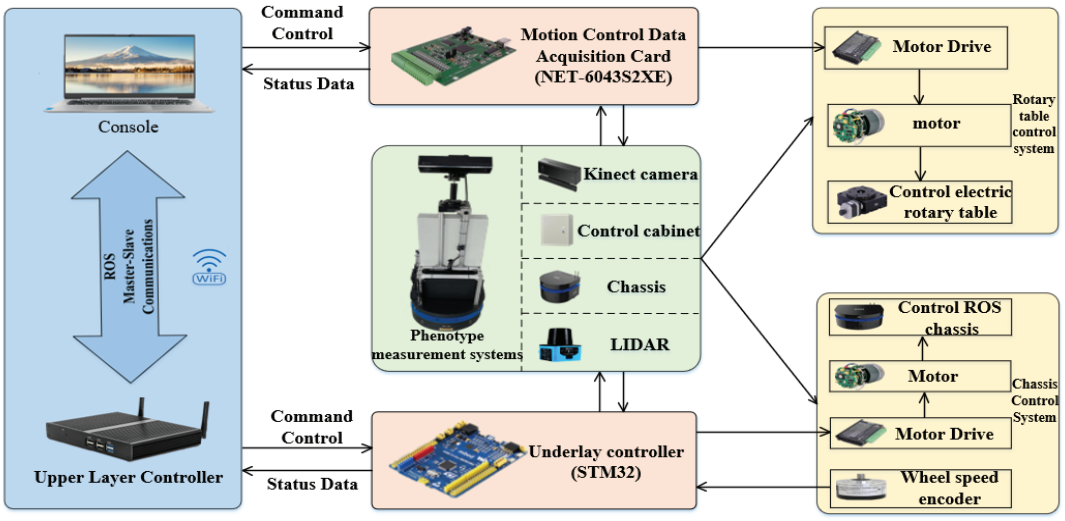


Figure 2. Schematic diagram of hardware structure.

The algorithm for the worldwide 3D reconstruction of the population of seedling maize is depicted in Figure 3 and works as follows:

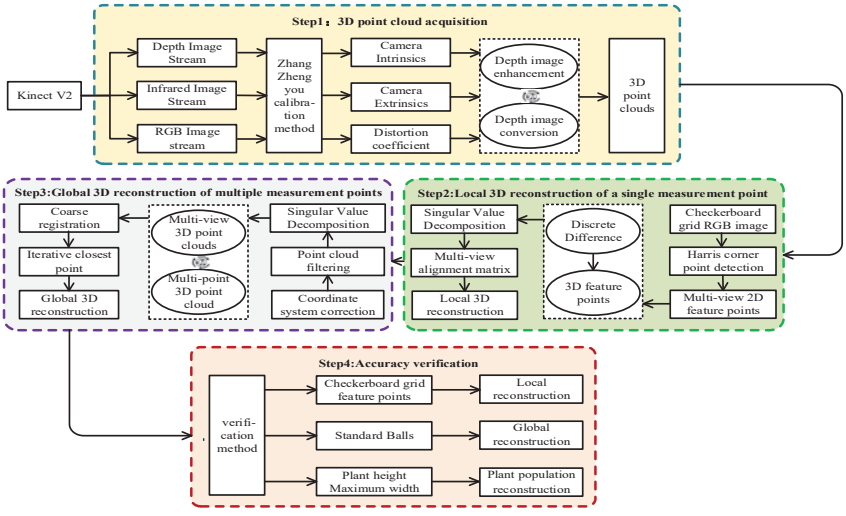


Figure 3. Flow chart of panoramic three-dimensional reconstruction of crops in self-propelled greenhouse.

(1) The camera parameters were obtained using the Zhang Zhengyou calibration method [20], and then, the depth images captured by the Kinect were used to create 3D point cloud maps using a similar triangle-based transformation.

(2) Pre-calibration of the multi-view alignment matrix for a single measurement point was performed using the singular value decomposition method and the Harris corner point identification technique.

(3) The camera coordinate system was adjusted using the region growth method and the random sample consensus (RANSAC) algorithm.

(4) A filtering technique was employed to eliminate point cloud noise from the non-corn plants.

(5) The ICP technique and coarse alignment procedure of several measurement locations were used to accomplish the global 3D reconstruction of the crop population. In this work, the checkerboard grid, standard sphere, plant height, and maximum width were used to calibrate the local, global, and crop group reconstruction accuracies, respectively.

2.3. Global 3D Reconstruction Method of Maize Population at Seedling Stage

2.3.1. Three-Dimensional Point Cloud Acquisition Method

Figure 4 illustrates the precise acquisition procedure used in this study to obtain the 3D point cloud data of the maize plant using the Kinect sensor. First, using the KinectV2 camera, RGB and depth images of the maize plants and infrared images at various angles of the checkerboard grid were collected. The internal camera parameters were then collected based on the infrared images using the Zhang Zhengyou calibration method, and the coordinates of the camera's center point (c_x, c_y) and focal length (f_x, f_y) were (256.00, 209.59), (365.18, 364.49). Using identical triangles and the camera small-aperture imaging method, the depth and RGB images of the maize plant were combined to create a 3D point cloud map with color information. Equations (1) and (2) depict how a point m (u, v) on the depth image and its corresponding 3D point M (X, Y, Z) relate to each other.

$$u = f_x \frac{X}{Z} + c_x \tag{1}$$

$$v = f_y \frac{Y}{Z} + c_y \tag{2}$$

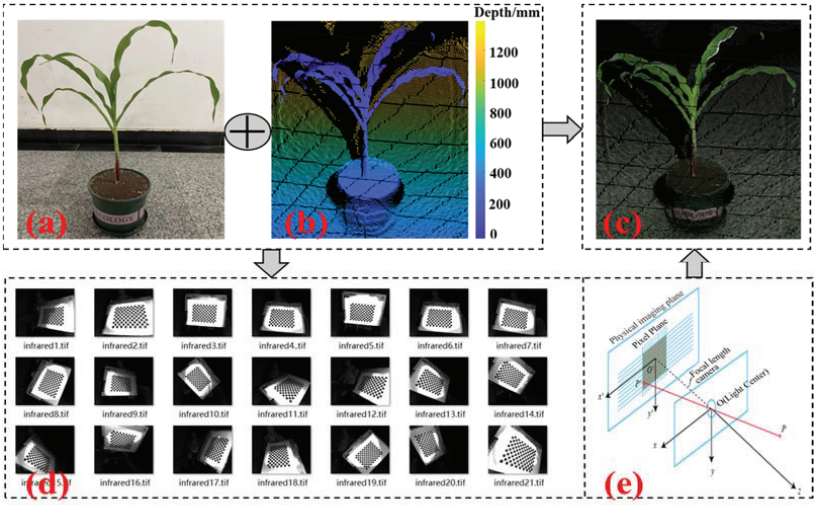


Figure 4. Acquisition process of 3D point cloud (**a**). Plant RGB image; (**b**) plant depth image; (**c**). plant 3D point cloud; (**d**). checkerboard grid infrared images; (**e**). similar triangle principle.

2.3.2. Single-Point Multi-View Alignment Matrix Pre-Calibration Method

In this study, the multi-view alignment matrix was pre-calibrated for a single measurement point using the singular value decomposition approach. The specific pre-calibration procedure is shown in Figure 5. The KincetV2 camera was mounted on the TBR 100 motorized rotary table to capture the RGB images of the checkerboard grid at two different viewing angles of 0° and 45° in the first stage. In the following stage, 2D points (X, Y) were generated in accordance with the mesh grid using the Harris corner detection technique to identify the 2D feature corner points of the RGB images of the checkerboard grid from the two viewpoints. In the third stage, discrete differences were used to determine the

mapping relations *Fx* and *Fy* between the 2D and 3D points, and the corresponding 3D feature points were computed using the 2D feature points of the checkerboard grid. The center of mass of the neighboring view feature points is determined in the fourth step using Equations (3) and (4), and the singular value decomposition method is applied to solve the rotation matrix *R* and translation matrix *T* of the 45° view transformation to the 0° view (there is no need to repeat the calibration subsequently; it is sufficient to calibrate once).

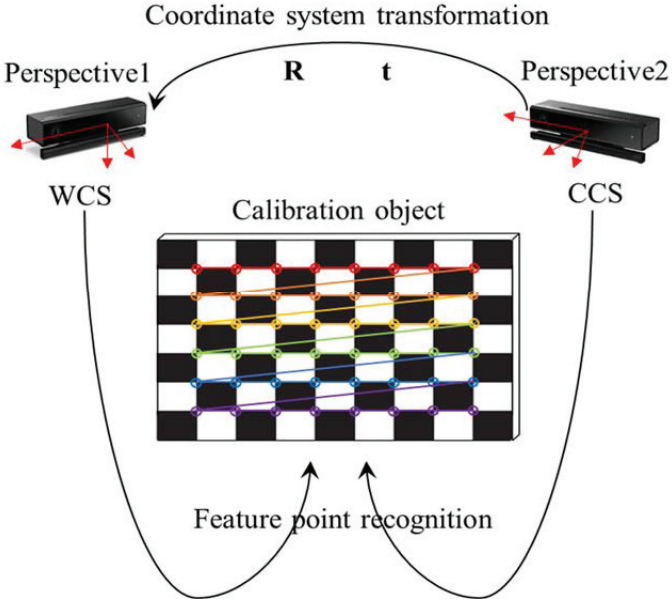


Figure 5. Pre-calibration diagram of multi-view alignment matrix for single-side measurement points.

In this study, a multi-view alignment matrix for a single measurement point was solved using the singular value decomposition method [21]. Assuming two eigen point sets, *P* and *Q*, the rotation translation matrix between them is solved as follows:

(1) The centroid coordinates P_c (x_c, y_c, z_c) and Q_c (x_c, y_c, z_c) of the feature point sets *P* and *Q* are calculated according to (3) and (4).

$$P_c(x_c, y_c, z_c) = \frac{\sum_{i=1}^{n} w_i \cdot P_i(x_i, y_i, z_i)}{\sum_{i=1}^{n} w_i} \tag{3}$$

$$Q_c(x_c, y_c, z_c) = \frac{\sum_{i=1}^{n} w_i \cdot Q_i(x_i, y_i, z_i)}{\sum_{i=1}^{n} w_i} \tag{4}$$

where w_i denotes the weight and P_i (x_i, y_i, z_i) and Q_i (x_i, y_i, z_i) are the 3D coordinates of the points within the point set.

(2) The covariance matrix *E* is calculated using (5), where *E* is a dn-dimensional matrix; *X*, *Y* are dn-dimensional matrices, and $W = \mathrm{diag}(w_1, w_2, w_3, \ldots, w_n)$.

$$E = XWY^T = \sum_{i}^{n} \frac{[(P_i - P_C)(Q_i - Q_c)^T]}{\sum_{i=0}^{n} w_i} \tag{5}$$

Equation (6) illustrates the execution of the singular value decomposition of the matrix E. The three matrices, U, V, and diagonal arrays, and (7) and (8) can be used to define the rotation matrix R and translation matrix T.

$$E = U \cdot \Lambda \cdot V^T \tag{6}$$

$$R = V \cdot U^T \tag{7}$$

$$T = Q_C - R \cdot P_c \tag{8}$$

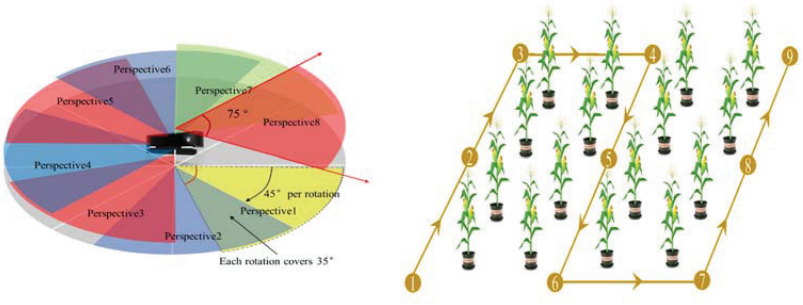
Equation (9) is applied to convert the point clouds of the other view camera coordinate systems to the first view camera coordinate system. R and T are the rotation and translation matrices, respectively.

$$PC_{j+1}{}' = R^j PC_{j+1} + \frac{1 - R^j}{1 - R} T \tag{9}$$

The 3D point cloud data of the *j-th* viewpoint world coordinate system are represented by PC_j, whereas the *j-th* viewpoint camera coordinate system are represented by $PC_j{}'$.

2.3.3. Multi-Point 3D Point Cloud Coarse Alignment Method

This study used the ROS mobile platform to achieve the coarse alignment of 3D point clouds from a multi-point maize plant. The first step is to locate them and the multi-survey point positioning and navigation; the RVIZ 2D map of maize plants was created using the DashgoB1 ROS mobile platform's map-building feature. Next, the robot's localization in the generated map was accomplished using the adaptive Monte Carlo localization technique (AMCL), which is based on particle filtering. The movement path on the map is planned using the global path-planning algorithm to steer the chassis along the way and eventually arrive at the desired target spot. The second step is the acquisition of single-point multi-view 3D point cloud data; for this, we used the TBR 100 electric rotary slide and set the multi-view acquisition interval to 45°. Figure 6a shows the acquisition diagram. The third phase is the local 3D reconstruction of the multi-view point cloud. Method 2.3.2 is used to calibrate the multi-view transformation matrix. To achieve the unification of the coordinate system and realize the local 3D reconstruction of the multi-view maize plant point cloud, the coordinate system of the first view camera is used as the global coordinate system. The point clouds of the other views are then aligned to the global coordinate system. The location of the acquisition point is determined in the fourth step. In this study, the reconstructed site area was 4 m × 4 m. Multi-point 3D point cloud data gathering is necessary because local reconstruction cannot obtain cloud data for the entire site. The noise level of the plant point cloud increases with the distance from Kinect, as does the inaccuracy. This study screened the point cloud data within 2.25 m of the Kinect origin by setting the radius of the enclosing box to 2.25 m.



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

Figure 6. 3D point cloud data acquisition. (**a**). Multi-view 3D point cloud data acquisition; (**b**). multi-measuring points 3D point cloud data acquisition.

To guarantee an adequate overlap area between the measurement points, nine measurement points were evenly distributed according to the spatial distribution of the corn plants. The horizontal distance between the consecutive measurement locations was fixed at 1.6 m for consistency. Figure 6b depicts the locations of the measurement points and the acquisition paths. The final step was to pre-calibrate the coarse alignment matrix of several measurement points. A checkerboard grid was established between the adjacent measurement sites, and the 2.3.2 algorithm was utilized to accomplish the pre-calibration of the coarse alignment transformation matrix between the adjacent measurement points. Because the target navigation point position was established, subsequent crop reconstruction and measurement did not require repeated calibration.

To achieve the coarse alignment of multipoint 3D point clouds, the fifth point camera coordinate system was employed as the global coordinate system, and the point clouds under other point camera coordinate systems were translated into the global coordinate system.

2.3.4. ICP Fine Alignment Using Overlapping Regions

Because the proportion of non-plant point clouds was too high and there were some noise points in the original point clouds, the ICP had alignment issues [22–24]. Therefore, point clouds must first be processed. The straight-pass filtering algorithm [25] was initially employed in this study to segregate the point clouds of corn and non-corn plants by utilizing 1 cm below the true height of the planters. Owing to the possibility of gray noise (such as soil) in the point cloud after direct-pass filtering, the super green factor index ExG was applied for additional filtering. Finally, for outlier elimination, statistical filtering using radius filtering is performed [26,27]. The standard deviation of each point in the point cloud with respect to its k-neighborhood (k value of 35) was computed, and the outliers were defined as points with a standard deviation larger than a threshold of 1.5. For radius filtering, the filter radius r and minimum number of points within the filter radius were set to 8 mm and 5, respectively, that is, points with less than five points within the 8 mm radius were deemed outliers.

Because of the stringent requirements of the ICP on the initial location and overlapping area, this study presented an ICP fine registration algorithm based on overlapping area point clouds to achieve global three-dimensional reconstruction of maize plants. The steps involved were as follows:

Step 1: Mean downsampling processing was conducted on the point cloud to lower the computation amount of the point cloud registration in the subsequent step.

Step 2: Using the fifth measurement point camera coordinate system as the world coordinate system, the 2.3.3 method was used to solve the coarse alignment transformation matrix, transform the other measurement point camera coordinate system point cloud to the fifth measurement point camera coordinate system, and obtain the coordinates of each measurement point (x_i, y_i, z_i, i = 1,2,...9) after the coarse alignment transformation.

Step 3: The first and fifth measurement points were used as examples, and the coordinates of the center point of the measurement point were determined using $(\frac{x1+x5}{2}, \frac{y1+y5}{2}, \frac{z1+z5}{2})$. The distance between the measurement points was estimated using $\sqrt{(x1-x5)^2+(y1-y5)^2+(z1-z5)^2}$. The overlapping region of the two measurement points was defined as the area where the circle with the coordinates of the center of the measurement point is the center, and the spacing between the measurement points was the diameter.

Step 4: Using the camera coordinate system of measurement point five as the global coordinate system, the ICP transformation matrix between the camera coordinate system of measurement point five and that between its neighboring measurement points was solved using the ICP based on the point cloud of corn plants in the overlapping area.

Step 5: Using the solved fine registration transformation matrix, the camera coordinate system of the other measuring points was transformed into a five-camera coordinate system of the measuring points. The fine registration of the 3D point cloud of various measuring locations was accomplished, and finally, the global 3D reconstruction of the corn seedling population was achieved. The ICP was primarily utilized to solve the fine registration transformation matrix.

The specific registration steps are as follows:

(1) From source point cloud P, the subset P_0, $P_0 \in P$ was selected.

(2) In the target point cloud Q, the matching point subset Q_0 of subset P_0, $Q_0 \in Q$ such that Q_i -P_i = min was found.

(3) The equation $f(R,T) = \sum_{i=1}^{n} \parallel Q_i - RP_i - T \parallel^2$ was satisfied as the minimum requirement by calculating the rotation matrix R and translation vector T and updating the subset P 0′ of the source point cloud.

(4) It was determined whether the iteration ends based on $d = \frac{1}{n}\sum_{i=1}^{n} \parallel Q_i - P_i \parallel^2$. If d is less than the specified threshold or the specified number of iterations is reached, the algorithm terminates; otherwise, the process was returned to Step (2) to continue the iteration.

2.4. Calibration Method for Accuracy of Global Reconstruction of Maize Population at Seedling Stage

2.4.1. Calibration of the Accuracy of Plant Height and Maximum Width Measurement

The major direction of the point cloud model of the maize plant based on the Kinect 3D reconstruction deviated from the 3D coordinate axis direction under visualization. To facilitate the subsequent measurement of the phenotypic parameters such as the plant height and maximum width, this study used the area growth method [28], the random sampling consistency algorithm [29], and the Harris corner point detection algorithm to uniformly convert the point clouds from different viewpoints into a ground-based global coordinate system.

(1) Plant height

In this study, a single corn plant was first segmented from the scene using a density-based point cloud clustering technique. Subsequently, all the point clouds of the corn plant were traversed, and the maximum and minimum values of the z-coordinate of the single corn plant were determined. Finally, the absolute value of the difference was considered to be the current corn plant height.

(2) Maximum width

To construct the matching projected point clouds, the extracted point clouds of individual maize plants were projected onto the OXY plane [30], and the largest outer circle of the projected point cloud in the OXY plane was calculated. The outer circle diameter represents the maximum width of the individual corn plants.

In this study, three metrics were used to assess the global reconstruction accuracy of crop populations. These are the root mean square error (*RMSE*), mean absolute percentage error (*MAPE*), and coefficient of determination (R^2). The following equations were used to calculate the above-mentioned metrics:

$$RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(P_i - Q_i)^2} \tag{10}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Q_i - P_i}{Q_i}\right| \times 100\% \tag{11}$$

$$R^2 = \left\{ \frac{\sum_{i=1}^{n}(Q_i - Q_{avg})\cdot(P_i - P_{avg})}{\left[\sum_{i=1}^{n}(P_i - P_{avg})^2\right]^{0.5}\left[\sum_{i=1}^{n}(Q_i - Q_{avg})^2\right]^{0.5}} \right\} \tag{12}$$

where,

P_i is the algorithm measurement of the i-th plant;
Q_i is the manual measurement of the i-th plant;
P_{avg} is the mean value of the algorithm measurement;
Q_{avg} is the mean value of the manual measurement;
n is the number of samples.

2.4.2. Calibration of the Accuracy of Global 3D Reconstruction of the Standard Sphere

$$HD(RP, GP) = \left\{ \min_{P_b \in MP}\{d(P_a, P_b)\} \right\} \tag{13}$$

where HD is the shortest distance between the generated and reconstructed point sets.

The spheres that generate the point set and reconstructed point set are denoted by GP and RP, respectively.

P_a and P_b are the points in RP and MP, respectively.

3. Analysis and Results

To validate the efficacy of the global reconstruction algorithm presented in this study for seedling maize populations, the accuracy was independently calibrated for neighboring viewpoint checkerboard grid feature points, multi-plant test subjects, and standard polystyrene foam balls.

3.1. Analysis of Single Measurement Point Local Reconstruction Accuracy

The tessellation grid feature points of the nearby views were used as measurement objects to assess the local 3D reconstruction accuracy of a single measurement point. The singular value decomposition approach was used to produce the tessellation grid 3D feature point matching and neighboring view alignment, and Figure 7 illustrates the 3D feature point matching accuracy analysis. The *RMSE* of the related tessellation 3D feature points was calculated as 0.18 cm, indicating that the reconstruction accuracy of the local 3D reconstruction was high.

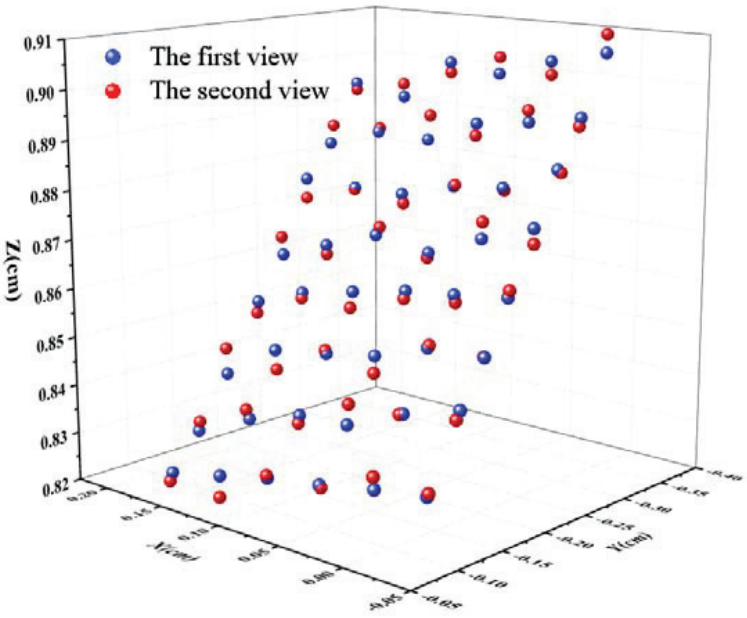


Figure 7. Matching accuracy analysis of 3D feature points.

3.2. Analysis of the Accuracy of Global 3D Crop Population Reconstruction

A total of 96 seedling maize plant samples were reconstructed in global 3D for six cycles in this experiment, and single maize point clouds were collected for quantitative analysis. The global reconstruction phase for the maize plant population is shown in Figure 8. Figure 9 depicts the global reconstruction findings for the same group of maize plants on different dates in the same position. Figure 10a,b illustrate a comparative evaluation of the algorithm's projected maize plant height and maximum width values with actual manual measurement values.

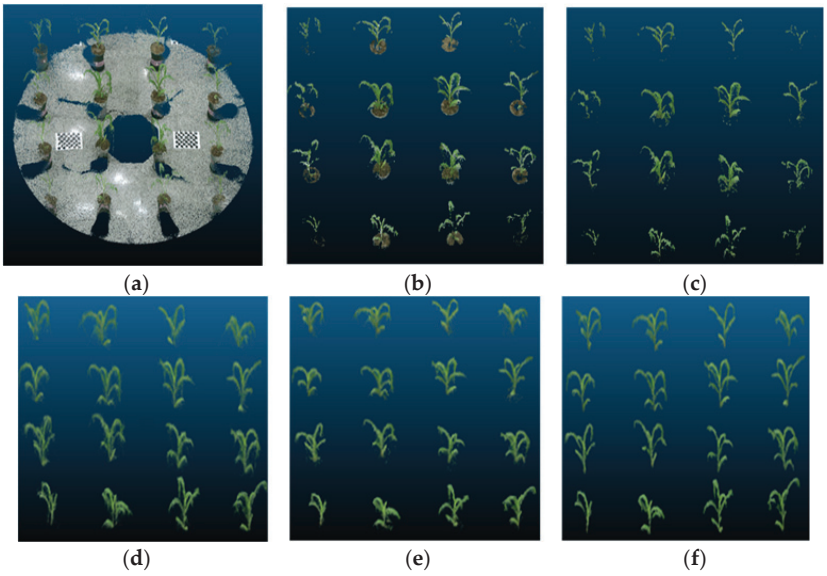


Figure 8. 3D reconstruction process of crop plants. (**a**). Local 3D reconstruction; (**b**). direct filtering; (**c**). ultra-green component value denoising; (**d**). 3D point cloud coarse registration; (**e**). ICP fine registration; (**f**). radius filtering and statistical filtering denoising.

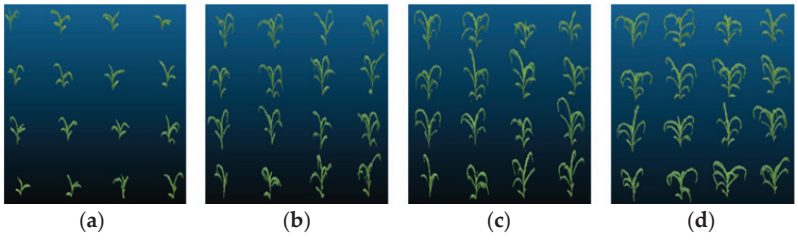


Figure 9. Global 3D point cloud model reconstruction of crop plants. (**a**). May 28th; (**b**). June 7th; (**c**). June 17th; (**d**). June 27th.

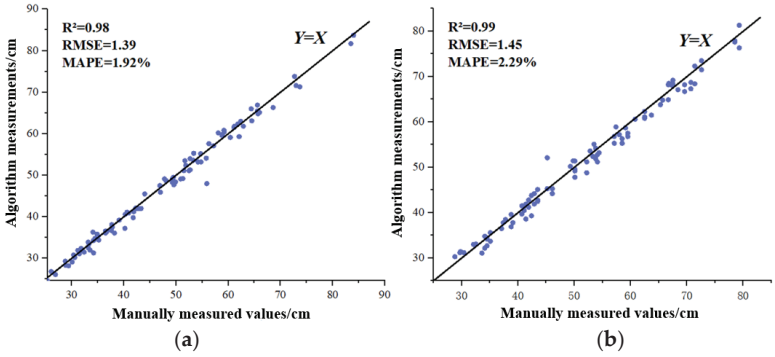


Figure 10. Comparison of artificial and algorithm measured values of crop plants. (**a**). Plant height; (**b**). maximum width.

From Figure 10a, R^2 = 0.98, *RMSE* = 1.39 cm, and *MAPE* = 1.92%, and the accuracy of the algorithm for measuring the plant height was 98.08%. As shown in Figure 10b,

R^2 = 0.99, *RMSE* = 1.45 cm, *MAPE* = 2.29%, and the accuracy of the algorithm for measuring the maximum plant width was 97.71%. The results in Figure 10 reveal that the algorithm measurement error of the maximum width is greater than that of the plant height, which is mostly due to the more involved manual measurement of the maximum width compared to the measurement of the plant height. The overall accuracy of this seedling maize population 3D reconstruction technique is excellent, and the algorithm measurements have a significant connection with manual measurements.

3.3. Analysis of Standard Sphere Global 3D Reconstruction Accuracy

Because of the flaws in hand measurements, a standard foam sphere was chosen as the measurement object for further evaluation of the global 3D reconstruction accuracy. The Hausdorff Distance set was used to quantify the global 3D reconstruction accuracy. Figure 11a,b show the RGB images of six foam spheres and single-measurement-point local 3D reconstructions. The worldwide 3D reconstruction of conventional foam spheres with various measurement sites is shown in Figure 11c. Figure 11d depicts the standard spheres produced by the CloudCompare software. Figure 12 shows the Hausdorff Distance set distribution for standard spheres with diameters of 200, 300, 350, 400, and 500 mm. The distance sets are separated into five segments: 0 cm $\leq$ HD $\leq$ 0.2 cm, 0.2 cm < HD $\leq$ 0.5 cm, 0.5 cm < HD $\leq$ 0.8 cm, 0.8 cm < HD $\leq$ 1.2 cm, and HD > 1.2 cm. The average distance set distribution of all spheres can be computed, with 55.26% of the Hausdorff Distance sets less than 0.5 cm. Approximately 76.83% of the cloud points had a distance less than 0.8 cm and only approximately 8.73% of the point clouds had distances larger than 1.2 cm, showing that the majority of the standard sphere reconstruction point sets vary within 0.8 cm of the original coordinate positions, and only a few point sets diverge from the original coordinate positions.

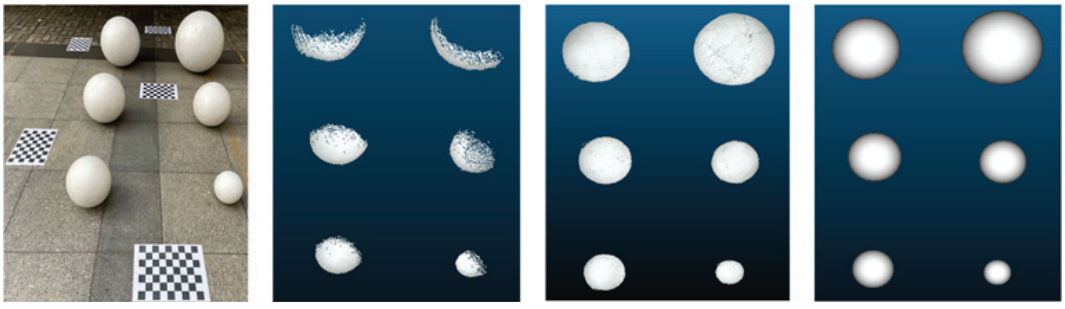


Figure 11. Standard sphere 3D point cloud model reconstruction. (**a**). Standard sphere RGB image; (**b**). local 3D reconstruction of the standard sphere; (**c**). global 3D reconstruction of the standard sphere; (**d**). standard sphere generation diagram.

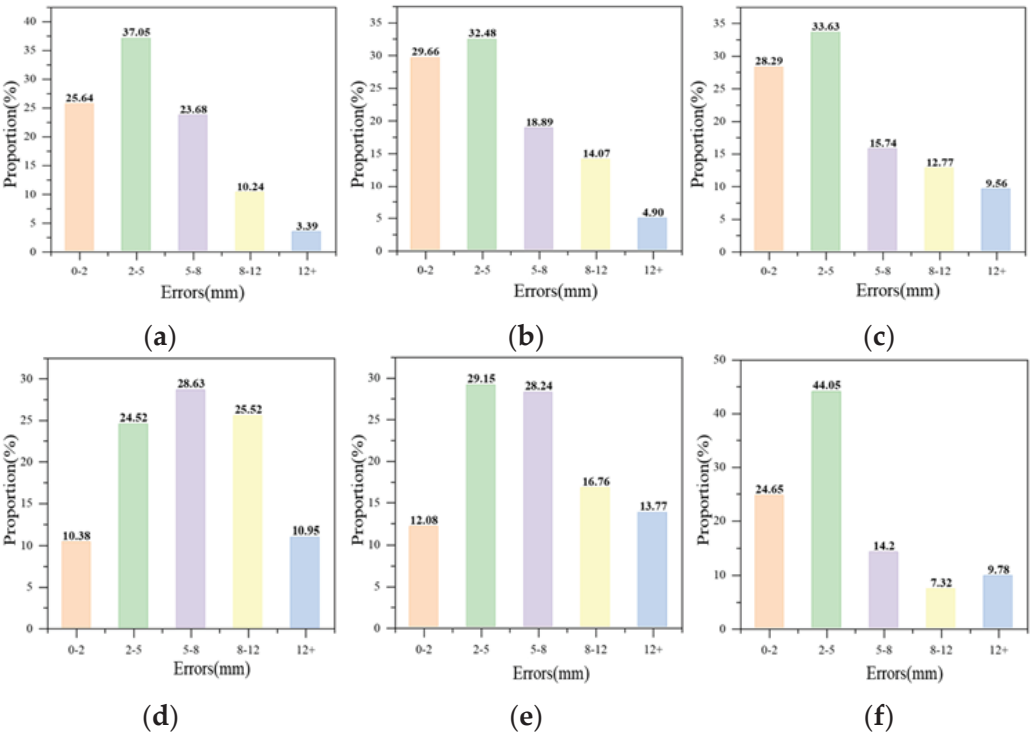


Figure 12. HD distribution ratio of standard spherical distance set. Diameters of (**a**). 200 mm; (**b**). 300 mm; (**c**). 350 mm; (**d**). 400 mm; (**e**). 500 mm; (**f**). 600 mm.

4. Conclusions

Because most previous crop phenotype measurement methods were only for individual plants, in this study, a self-propelled crop phenotype measurement device for 96 seedling maize plants in six cycles was designed, and a global 3D reconstruction method for the seedling maize plant population was evaluated, with the following main findings.

(1) The RMSE of the feature points corresponding to the local reconstruction of adjacent views was 0.18 mm, and the distance set HD between the standard sphere reconstructed point cloud and the software generated point cloud was less than 0.5 cm for 55.26%, less than 0.8 cm for 76.83%, and only 8.76% of the point cloud distance was greater than 1.2 cm. This indicated that most of the point clouds did not deviate from the original coordinate positions after alignment, and the reconstruction accuracy of this algorithm was sufficient to meet the phenotypic measurement needs of seedling maize plants.

(2) The MAPE of the maize plant height and maximum width were 1.92% and 2.29%, respectively, compared to real manual measurements. The RMSE values were 1.39 cm and 1.45 cm, respectively, and R^2 was 0.98 and 0.99. This demonstrated the high accuracy of the proposed seedling maize population reconstruction.

Through the 3D modeling of seedling maize populations, this study provided supporting data and theoretical guidance for phenotypic characterization and the accurate and intelligent management of maize. Because the alignment transformation matrix was obtained using a pre-calibration method, inter-crop occlusion will not impair the reconstruction accuracy, but it will result in missing information for some crops and will affect crop reconstruction integrity.

The influence of ground leveling was not considered in this study, and the algorithm was tested only on maize seedlings. The applicability and robustness of the algorithm need

to be confirmed, and more experimental studies on various growth stages of different crops will be undertaken at a later stage.

Author Contributions: Conceptualization, N.X.; Methodology, N.X. and G.S.; Software, N.X. and Y.B.; Validation, N.X. and X.Z.; Formal Analysis, N.X., Y.B. and X.Z.; Investigation, N.X., J.C. and Y.H.; Writing—Original Draft Preparation, N.X., G.S. and Y.B.; Writing—Review & Editing, N.X., Y.H. and J.C.; Project Administration, G.S.; Funding Acquisition, G.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Jiangsu agricultural science and technology Innovation Fund (No. CX(22)3097), Jiangsu agricultural science and technology Innovation Fund (No. CX(21)2006), The key R&D Program of Jiangsu Province(No. BE2022363) and High-end Foreign Experts Recruitment Plan of China (No. G2021145009L).

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: The author would like to thank the editors and reviewers for their comments on how to improve the quality of this work.

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

References

1. Fan, J.; Zhang, Y.; Wen, W.; Gu, S.; Lu, X.; Guo, X. The future of Internet of Things in agriculture: Plant high-throughput phenotypic platform. *J. Clean. Prod.* **2020**, *280*, 123651. [CrossRef]
2. Zhao, C.; Zhang, Y.; Du, J.; Guo, X.; Wen, W.; Gu, S.; Wang, J.; Fan, J. Crop phenomics: Current status and perspectives. *Front. Plant Sci.* **2019**, *10*, 714. [CrossRef]
3. Sun, G.; Wang, X.; Liu, J.; Sun, Y.; Ding, Y.; Lu, W. Multi-modal three-dimensional reconstruction of greenhouse tomato plants based on phase-correlation method. *Trans. Chin. Soc. Agric. Eng. (Trans. CSAE)* **2019**, *35*, 134–142. [CrossRef]
4. Song, P.; Wang, J.; Guo, X.; Yang, W.; Zhao, C. High-throughput phenotyping: Breaking through the bottle-neck in future crop breeding. *Crop J.* **2021**, *9*, 633–645. [CrossRef]
5. Selvaraj, M.G.; Valderrama, M.; Guzman, D.; Valencia, M.; Ruiz, H.; Acharjee, A. Machine learning for high-throughput field phenotyping and image processing provides insight into the association of above and be-low-ground traits in cassava (*Manihot esculenta* Crantz). *Plant Methods* **2020**, *16*, 87. [CrossRef]
6. Li, Y.; Liu, J.; Zhang, B.; Wang, Y.; Yao, J.; Zhang, X.; Fan, B.; Li, X.; Hai, Y.; Fan, X. Three-dimensional reconstruction and phe-notype measurement of maize seedlings based on multi-view image sequences. *Front. Plant Sci.* **2022**, *13*, 974339. [CrossRef]
7. Qiu, R.; Miao, Y.; Zhang, M.; Li, H. Detection of the 3D temperature characteristics of maize under water stress using thermal and RGB-D cameras. *Comput. Electron. Agric.* **2021**, *191*, 106551. [CrossRef]
8. Wu, J.; Xue, X.; Zhang, S.; Qin, W.; Chen, C.; Sun, T. Plant 3D reconstruction based on LiDAR and multi-view sequence images. *Int. J. Precis. Agric. Aviat.* **2018**, *1*, 37–43. [CrossRef]
9. Nguyen, T.T.; Slaughter, D.C.; Max, N.; Maloof, J.N.; Sinha, N. Structured Light-Based 3D Reconstruction System for Plants. *Sensors* **2015**, *15*, 18587–18612. [CrossRef]
10. Golbach, F.; Kootstra, G.; Damjanovic, S.; Otten, G.; van de Zedde, R. Validation of plant part measurements using a 3D reconstruction method suitable for high-throughput seedling phenotyping. *Mach. Vis. Appl.* **2016**, *27*, 663–680. [CrossRef]
11. Teng, X.; Zhou, G.; Wu, Y.; Huang, C.; Dong, W.; Xu, S. Three-dimensional reconstruction method of rapeseed plants in the whole growth period using RGB-D camera. *Sensors* **2021**, *21*, 4628. [CrossRef]
12. Peng, Y.; Yang, M.; Zhao, G.; Cao, G. Binocular-vision-based structure from motion for 3D reconstruction of plants. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 8019505.
13. Ma, X.; Zhu, K.; Guan, H.; Feng, J.; Yu, S.; Liu, G. Calculation Method for Phenotypic Traits Based on the 3D Reconstruction of Maize Canopies. *Sensors* **2019**, *19*, 1201. [CrossRef] [PubMed]
14. Lin, X.; Zhang, J. 3D Power Line Reconstruction from Airborne LiDAR Point Cloud of Overhead Electric Power Transmission Corridors. *Acta Geod. et Cartogr. Sin.* **2016**, *45*, 347.
15. Liu, R.; Liu, T.; Dong, R.; Li, Z.; Zhu, D.; Su, W. 3D modeling of maize based on terrestrial LiDAR point cloud data. *J. China Agric. Univ.* **2014**, *19*, 196–201.
16. Zheng, C.; Abd-Elrahman, A.; Whitaker, V. Remote sensing and machine learning in crop phenotyping and management, with an emphasis on applications in strawberry farming. *Remote Sens.* **2021**, *13*, 531. [CrossRef]
17. Peng, C.; Li, S.; Miao, Y. Stem-leaf segmentation and phenotypic trait extraction of tomatoes using three-dimensional point cloud. *Trans. Chin. Soc. Agric. Eng. (Trans. CSAE)* **2022**, *38*, 187–194, (In Chinese with English abstract).
18. Hu, P.; Guo, Y.; Li, B.; Zhu, J.; Ma, Y. Three-dimensional reconstruction and its precision evaluation of plant architecture based on multiple view stereo method. *Trans. Chin. Soc. Agric. Eng. (Trans. CSAE)* **2015**, *31*, 209–214, (In Chinese with English abstract).
19. He, J.Q.; Harrison, R.J.; Li, B. A novel 3D imaging system for strawberry phenotyping. *Plant Methods* **2017**, *13*, 93. [CrossRef]

20. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [CrossRef]
21. Sorkine-Hornung, O.; Rabinovich, M. Least-squares rigid motion using svd. *Computing* **2017**, *1*, 1–5.
22. Zhang, K.; Chen, H.; Wu, H.; Zhao, X.; Zhou, C. Point cloud registration method for maize plants based on conical surface fitting—ICP. *Sci. Rep.* **2022**, *12*, 6852. [CrossRef] [PubMed]
23. Vázquez-Arellano, M.; Reiser, D.; Paraforos, D.S.; Garrido-Izard, M.; Burce, M.E.C.; Griepentrog, H.W. 3-D reconstruction of maize plants using a time-of-flight camera. *Comput. Electron. Agric.* **2018**, *145*, 235–247. [CrossRef]
24. Lin, C.; Wang, H.; Liu, C.; Gong, L. 3D reconstruction based plant-monitoring and plant-phenotyping platform. In Proceedings of the 2020 3rd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, 4–6 December 2020; pp. 522–526.
25. Zou, W.; Shen, D.; Cao, P.; Lin, C.; Zhu, J. Fast Positioning Method of Truck Compartment Based on Plane Segmentation. *IEEE J. Radio Freq. Identif.* **2022**, *6*, 774–778. [CrossRef]
26. Han, X.-F.; Jin, J.S.; Wang, M.-J.; Jiang, W.; Gao, L.; Xiao, L. A review of algorithms for filtering the 3D point cloud. *Signal Process. Image Commun.* **2017**, *57*, 103–112. [CrossRef]
27. Bi, S.; Wang, Y. LiDAR Point Cloud Denoising Method Based on Adaptive Radius Filter. *Trans. Chin. Soc. Agric. Mach.* **2021**, *52*, 234–243, (In Chinese with English abstract).
28. Liu, M.; Shao, Y.; Li, R.; Wang, Y.; Sun, X.; Wang, J.; You, Y. Method for extraction of airborne LiDAR point cloud buildings based on segmentation. *PLoS ONE* **2020**, *15*, e0232778. [CrossRef]
29. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications To Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
30. Das Choudhury, S.; Maturu, S.; Samal, A.; Stoerger, V.; Awada, T. Leveraging Image Analysis to Compute 3D Plant Phenotypes Based on Voxel-Grid Plant Reconstruction. *Front. Plant Sci.* **2020**, *11*, 521431. [CrossRef]

agriculture

MDPI

Article

A Cascaded Individual Cow Identification Method Based on DeepOtsu and EfficientNet

Ruihong Zhang [1], Jiangtao Ji [1], Kaixuan Zhao [1,*], Jinjin Wang [1], Meng Zhang [1] and Meijia Wang [2]

[1] College of Agricultural Equipment Engineering, Henan University of Science and Technology, Luoyang 471023, China
[2] School of Electronic Information and Artificial Intelligence, Shannxi University of Science & Technology, Xi'an 710021, China
* Correspondence: kx.zhao@haust.edu.cn

Abstract: Precision dairy farming technology is widely used to improve the management efficiency and reduce cost in large-scale dairy farms. Machine vision systems are non-contact technologies to obtain individual and behavioral information from animals. However, the accuracy of image-based individual identification of dairy cows is still inadequate, which limits the application of machine vision technologies in large-scale dairy farms. There are three key problems in dairy cattle identification based on images and biometrics: (1) the biometrics of different dairy cattle may be similar; (2) the complex shooting environment leads to the instability of image quality; and (3) for the end-to-end identification method, the identity of each cow corresponds to a pattern, and the increase in the number of cows will lead to a rapid increase in the number of outputs and parameters of the identification model. To solve the above problems, this paper proposes a cascaded dairy individual cow identification method based on DeepOtsu and EfficientNet, which can realize a breakthrough in dairy cow group identification accuracy and speed by binarization and cascaded classification of dairy cow body pattern images. The specific implementation steps of the proposed method are as follows. First, the YOLOX model was used to locate the trunk of the cow in the side-looking walking image to obtain the body pattern image, and then, the DeepOtsu model was used to binarize the body pattern image. After that, primary classification was carried out according to the proportion of black pixels in the binary image; then, for each subcategory obtained by the primary classification, the EfficientNet-B1 model was used for secondary classification to achieve accurate and rapid identification of dairy cows. A total of 11,800 side-looking walking images of 118 cows were used to construct the dataset; and the training set, validation set, and test set were constructed at a ratio of 5:3:2. The test results showed that the binarization segmentation accuracy of the body pattern image is 0.932, and the overall identification accuracy of the individual cow identification method is 0.985. The total processing time of a single image is 0.433 s. The proposed method outperforms the end-to-end dairy individual cow identification method in terms of efficiency and training speed. This study provides a new method for the identification of individual dairy cattle in large-scale dairy farms.

Keywords: dairy cow; individual identification; body pattern image; binarization; cascaded classification

Citation: Zhang, R.; Ji, J.; Zhao, K.; Wang, J.; Zhang, M.; Wang, M. A Cascaded Individual Cow Identification Method Based on DeepOtsu and EfficientNet. *Agriculture* **2023**, *13*, 279. https://doi.org/10.3390/agriculture13020279

Academic Editors: Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 27 December 2022
Revised: 17 January 2023
Accepted: 20 January 2023
Published: 23 January 2023

1. Introduction

With the improvement in people's living standards and consumption levels, the demand for animal protein is gradually increasing [1,2]. With limited environmental resources and increasing labor costs, the large-scale development of farms is the key to meeting the above needs [3]. In the management of large dairy farms, the use of manual methods to monitor the health and production of each cow is not only time-consuming and labor-intensive, but also subjective. Therefore, precise dairy farming technology, which is used to monitor individual dairy cows in real time and make timely management

decisions, is an important way to improve efficiency and reduce costs in large-scale farms. The automatic identification of the individual identity of dairy cows is the premise and foundation for achieving precision management.

Currently, passive radio frequency identification (RFID) technology [4], active RFID technology [5], and other wireless technologies—such as radar [6] and wireless local area networks [7]—are sensor-based individual identification methods commonly used on farms. The above methods generally have high accuracy and wide applicability, but the identification system requires cows to wear ear tags or transponders, which not only cause stress to cows but are also prone to damage or loss [8].

In recent years, with the development of computer vision technology in dairy cow behavior analysis and health monitoring [9–13], individual cow identification methods based on biometrics have become a research hotspot [14,15]. Noncontact individual identification systems based on biometrics have the advantages of low cost and not inducing stress responses in cows and can be integrated into intelligent monitoring systems for cows. The muzzle print [16], iris [17], head contours and textures [18,19], tailhead pattern [20], body pattern [14], and gait characteristics [21] of a cow can be used as distinguishable cow identifiers. However, it is difficult to obtain clear images of specific areas of a cow's head—such as the muzzle print, iris, head texture, etc.—which requires the cow to have a high degree of coordination, and the shooting results are easily affected by the shooting angle and position. Gait activity and characteristics will change due to changes in the physiological state of cows (such as lameness, estrus, etc.), resulting in the reduced accuracy of individual identification. In addition, some scholars have tried to identify individual cows by locating and recognizing numbers on tags worn by cows (e.g., ear tags [22] and collar ID tags [23,24]), but the implementation of tags requires additional manpower and material resources.

Body pattern refers to the regular distribution of black and white hair in the trunk area of Holstein cows. The distribution area of the body pattern is wide, and the body pattern image can be obtained by collecting side-looking, top-looking images or videos of a cow in the walking process. Zhao et al. [14] extracted a 48×48 matrix from a cow's trunk image as the eigenvalue, and a convolutional neural network was constructed and trained as the individual cow identification model. The dataset contained 30 cows, and 90.55% of the images were correctly identified in the test. Li et al. [20] located a cow's tailhead, and the contour of the black and white pattern of the tailhead was obtained by binary image processing. Then, the feature matrix was extracted, and classification was carried out. The dataset contained 10 cows, and the final accuracy was 99.7%. The number of cows studied by the above methods is small, and the adaptability to large-scale farms is unknown. Therefore, scholars have begun to build datasets containing more cows for the individual identification of cow groups in large-scale farms. He et al. [25] preprocessed the back images of 89 cows and constructed a milking individual cow identification model based on the improved YOLO v3 algorithm, which achieved 95.91% identification accuracy. Hu et al. [8] used YOLO to detect the position of cows and separated the head, body, and legs from the detection frame of cows. The features of these three parts were extracted, fused, and classified. It achieved 98.36% accuracy for 93 cows. Shen et al. [26] used the YOLO model to obtain the detection box containing the cow, and the AlextNet algorithm was fine-tuned to identify cow individuals. The constructed dataset contained 105 cows, and the identification accuracy was 96.65%.

The output end of the individual cow identification model constructed by the method directly corresponds to the number of cows, the increase in the number of output ends causes an increase in identification network parameters, and the time cost for individual identification and retraining of the network correspondingly increases. In addition, the body patterns of different cows may be similar, which will increase the difficulty of correct identification. At the same time, the above methods all use RGB body images as the input of the identification model. However, the farming environment of dairy cows is complex, and light, stains, fences, and so on will affect the quality of body pattern images. This

means that the identification model should not only judge the classification of the target but also eliminate interference in the image, which increases the complexity of the identification network as well.

In view of the above problems, a cascaded dairy individual cow identification method based on EfficientNet [27] and DeepOtsu [28] is proposed in this paper and was applied to large-scale dairy farms. It can realize a breakthrough in dairy cow group identification accuracy and speed by binarization and cascaded classification of dairy cow body pattern images. The specific implementation steps of the proposed method are as follows: first, the body pattern image is obtained by using YOLOX to locate the trunk region of the cow, and then, the body pattern image is binarized by the DeepOtsu model. Then, primary classification is carried out according to the proportion of black pixels in the binary image. Then, for each subcategory obtained by primary classification, the EfficientNet model is used for secondary classification to identify the identity of the individual cow. Compared with the end-to-end identification method, the proposed cascaded identification method reduces the number of outputs and parameters of the individual identification model, which provides a new idea for the individual identification of dairy cows on large-scale dairy farms.

In general, we proposed a cascaded method for the individual identification of dairy cows that mainly consists of three modules: cow trunk localization, body pattern image binarization and cascaded classification. The main contributions of this paper are as follows.

- A new method of individual cow identification was proposed. The method comprises the following steps. First, the cow trunk region was detected to obtain a body pattern image. Then, the pattern image was binarized to highlight the distribution characteristics of the black and white patterns. Finally, the binary pattern image was classified to identify the individual cow.
- The body pattern images of cows were classed by utilizing a cascaded classification method. The method can reduce the number of output ends of the classification model and improve the efficiency of the training. The identification accuracy, speed, and training time of the proposed method were compared with those of the end-to-end identification method, and the results showed that the proposed method is superior to the end-to-end method.
- The body pattern image was binarized by the deep learning method. The experimental results showed that the deep learning method can better describe the features of RGB body pattern images, remove the interference factors in the images, and achieve better binarization accuracy.

2. Materials and Methods

2.1. Dataset Construction

2.1.1. Video Acquisition

In this study, 118 lactating Holstein cows were filmed at Coldstream Research Dairy Farm, University of Kentucky, USA. The cows returned to the cowshed after milking. A straight corridor was set on the only way back to the cowshed. Two electric fences were used on both sides as the boundary of the corridor. The width of the corridor was 2 m. The cows passed a weighing device before entering the corridor. The weighing device has electronically controlled doors to ensure an interval between cows when passing through the corridor, so individuals overlapping will not happen in the video. The image acquisition system consisted of a Nikon D5200 camera (Nikon, Tokyo, Japan) and a tripod, which was fixed on one side of the aisle at a distance of 3.5–4 m from the corridor and a height of 1.5 m from the ground. The specific location is shown in Figure 1. The acquisition time of the video was from 16:00 to 18:00 on sunny days from August to October 2016. The camera used a 35 mm lens (Nikon AF-S DX 35 mm f/1.8 G) (Nikon, Tokyo, Japan), and ISO 400, autoexposure and autofocus modes were selected when acquiring images. When a cow passed through the corridor, video shooting began, and when the cow walked to the

right edge of the field of view, shooting ended. The cows were filmed multiple times on different dates.

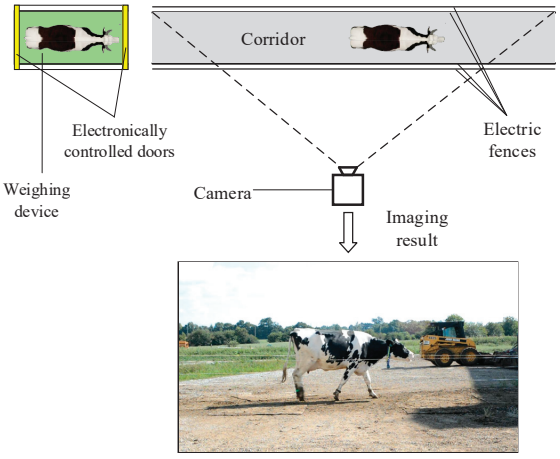


Figure 1. Diagram of the video acquisition system.

2.1.2. Video Decomposition and Processing

The collected videos were analyzed, and the overexposed videos were eliminated to obtain side-looking walking videos of cows. The construction of the dataset mainly comprised the following steps: (1) decomposing the video into image frames; (2) selecting the image frames randomly and quantitatively; (3) classifying the images; (4) normalizing the number of images; and (5) constructing and dividing the subdatasets.

(1) Decomposing the video into image frames. Video decomposition technology was used to decompose the cow side-looking walking videos into frame-by-frame images. The resolution of the cow side-looking image was 1280 pixels (horizontal) × 720 pixels (vertical).
(2) Selecting the image frames randomly and quantitatively. For each walking video, 100 side-looking walking images were randomly selected, and it was ensured that each image contained the complete trunk of the cow.
(3) Classifying the images. The side-view walking images belonging to the same cow were classified and placed into a folder.
(4) Normalizing the number of images. For a folder containing more than 100 images, 100 images were randomly selected as the image dataset corresponding to the cow. The final constructed dataset contained 11,800 images of 118 cows.
(5) Constructing and dividing the subdataset. Due to the large number of samples in the dataset, it is labor-intensive and unnecessary to annotate all the images to train and test the model. Therefore, 10 images of each cow in the dataset were randomly selected to construct a subdataset to train and test the trunk detection and body pattern binarization model. The subdataset contained 1180 images of 118 cows, and the subdataset was divided into a training set, validation set, and test set at a ratio of 5:3:2.

2.1.3. Image Annotation

The cascaded individual cow identification method proposed in this paper needs two annotations during training. One annotation involves labeling the trunk region when training the trunk location model, and the other involves labeling the body pattern in image binarization when training the body pattern binarization model. For trunk region annotation, the Labelme image annotation tool was used. For the body pattern image binarization annotation, the 3D drawing tool of the Windows system was used. The above annotation process was only processed for subdatasets.

2.1.4. Training and Test Platform

The YOLOX detection model, the DeepOtsu binarization model, and the EfficientNet classification model were trained and tested on the same hardware and software platform. The CPU of the platform was an Intel (R) Xeon (R) with 8 G memory. The graphics card of the platform was an NVIDIA Tesla K80(NVIDIA, CA, USA) with 12 G memory. The software environment for training and testing was an Ubuntu 18.04 LTS 64-bit system. The programming language was Python 3.8. CUDA11.0 and cuDNN8.0 were used as the parallel computing architecture and GPU acceleration library for deep neural networks, respectively.

2.2. Detection of the Trunk Area

The body pattern of a cow is mainly concentrated in the trunk area. To eliminate the influence of an irrelevant environment, the trunk area in the side-looking walking image of a cow was located. Existing methods for cow individual location include the frame difference method [14], Gaussian mixture model [29], and YOLO model based on a convolutional neural network (CNN) [25]. The frame difference method uses the difference operation of the adjacent frame images in the video image sequence to obtain the contour of the moving cow target. The Gaussian mixture model obtains the position of the moving cow target by analyzing the change in the gray value of the pixel point in the video. The above two methods need to analyze the continuous sequence of images in the video, and a moving interference object in the background will greatly affect the detection accuracy of the cow target. The YOLO model [30] is a one-step target detection algorithm based on a CNN that uses convolution to extract the features of the image and directly outputs the location and category of the target according to the features. To detect the trunk region in this study with many external interference factors, it is more appropriate to use the detection model based on deep learning. YOLOX was proposed by Ge et al. [31], and its performance exceeds that of the YOLO series of algorithms. YOLOX achieves 50.0% AP on COCO (1.8% higher than YOLOv5 and 2.5% higher than YOLOv4) [31], and the precision of YOLOX is much higher than that of YOLOv3 (33.0% AP). Therefore, we finally decided to use YOLOX to detect the trunk area of dairy cows.

The YOLOX model was built based on YOLOv5 and mainly included four modules: input, backbone, neck, and prediction modules. The structure of YOLOX is shown in Figure 2. When the image to be detected is input into the network, it is first adjusted to a size of 416 × 416 and then sent to the backbone of the network for feature extraction, obtaining three effective feature layers. In the neck module, a series of convolution, upsampling, and downsampling operations and others are carried out on the three effective feature layers to fuse different feature layers and strengthen the feature extraction process. Finally, the prediction module performs a convolution operation on the fused feature layers to obtain the category and position information of the detected target.

After detection, the original image was cropped according to the coordinate information of the detection frame to obtain the body pattern image of the cow. A schematic of the processing method is shown in Figure 2.

The training set in the subdataset was used to train the YOLOX-based trunk detection model. After training, the images of the test set in the subdataset were put into the trained detection model to test its performance. In this paper, AP and $AP^{IoU=0.75}$(AP75) in the index of the COCO dataset were used to evaluate the accuracy of the trunk detection model. These two indicators are defined as follows. The IoU (intersection over union) is a value used to measure the degree of overlap between a prediction box and a groundtruth box, and its formula is

$$\mathrm{IoU} = \frac{S_p \cap S_g}{S_p \cup S_g} \tag{1}$$

where S_p represents the area of the predicted bounding box, and S_g represents the area of the groundtruth bounding box. IoU threshold is used to determine whether the content in the prediction box is a positive sample. For the target detection model, the commonly

used evaluation indices were precision P (Precision) and R (Recall), and their calculation formulas are

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

where TP represents the number of correctly predicted targets. FP represents the number of falsely predicted targets, that is, the background was mistaken for a positive sample. FN represents the number of missed targets, that is, a positive sample was mistaken as the background. For each prediction box, a confidence value was generated, indicating the credibility of the prediction box. Different combinations of P and R were obtained by setting different confidence thresholds. Taking P and R as vertical and horizontal coordinates, respectively, the PR curve could be drawn. When the IoU threshold was set to 0.75, the area under the PR curve was $AP^{IoU=0.75}$ (AP75). AP was averaged over multiple IoU values. Specifically, we used 10 IoU thresholds of 0.50:0.05:0.95. AP and AP75 would comprehensively reflect the performance of the detection model.

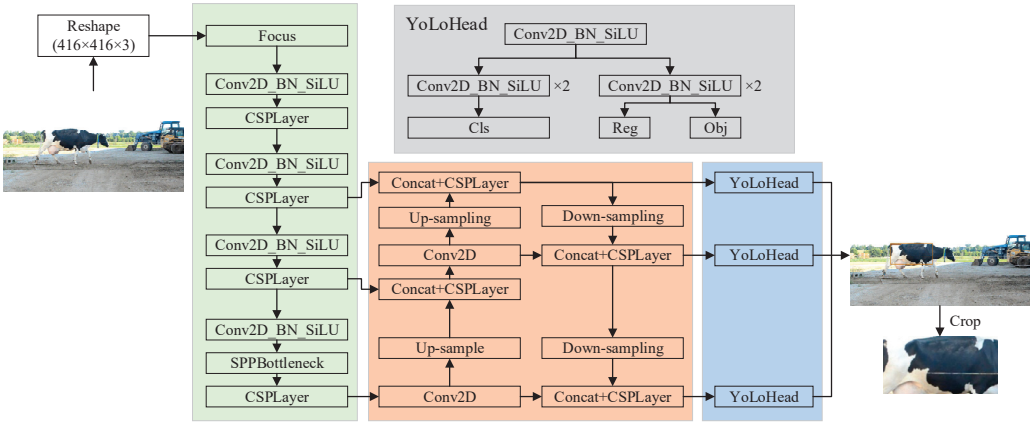


Figure 2. Acquisition of the cow body pattern image based on YOLOX.

After testing, the trained YOLOX model was used to detect the trunk areas of the remaining cow side-looking walking images in the dataset to obtain body pattern images of all cows.

2.3. Binarization of Body Pattern Images

The most prominent feature in the body pattern image of the trunk region is the distribution of black and white patterns. Therefore, in this study, the distribution of black and white patterns was used as the basis for the classification of body pattern images, that is, the identity of individual cows. To highlight the main feature of black and white patterns in the image, the body pattern image was binarized to make the area where black hair is located black and the area where white hair is located white in the image.

2.3.1. Traditional Binarization Method

In this study, due to the obvious color difference between black hair and white hair, two traditional binarization methods—the Otsu method and the color-based binarization method—were used to segment the cow body pattern images. The Otsu method uses the gray characteristics of the image to divide the image into two parts—foreground and background—and when the difference is greatest, the optimum threshold is taken. When using this method for binarization, the image needs to be processed into a gray image first.

In this paper, the weighted average method (Formula (4)) was used to perform grayscale processing on the image, and then, the Otsu method was used to perform binarization.

$$\mathrm{Gray(i,j)} = (R(i,j) + G(i,j) + B(i,j))/3 \tag{4}$$

where R (i, j), G (i, j), and B (i, j) represent the three components of each pixel point of the color image and Gray (i, j) represents the composite value of the three components, that is, the gray value of each pixel point of the processed gray image.

In addition, according to the statistics of the pixel points in the region where the black and white hairs are located in the cow trunk image, an image binarization method based on color feature was designed, as shown in Figure 3. The method determines whether the pixel point is assigned black (0) or white (1) according to the R, G, and B values of each pixel point in the image. In Figure 3, R, G, and B represent the three component values of each pixel point.

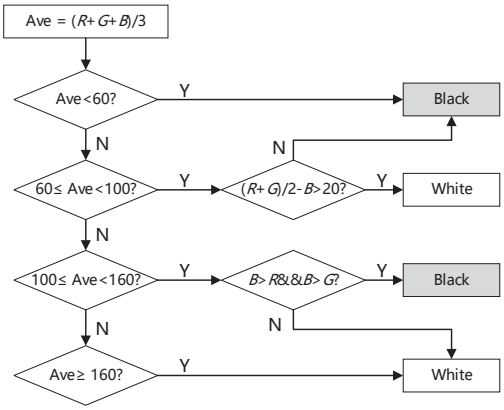


Figure 3. Binarization method based on color features.

2.3.2. DeepOtsu

There were noises from light, stains, occlusion in the background of the cow trunk images, which will lead to wrong binarization results. For example, the reflection caused by strong light makes the black hair area very bright, then the binarization result of this black hair area is easily misclassified as white (value 1); the presence of stains in the white hair area will cause the area to darken, and its binarization result is easily misclassified as black (value 0). Therefore, it is necessary to eliminate the background noise in the image in order to achieve better binarization effect. The binarization method based on simple features such as color and gray distributions may not achieve satisfactory results, because these features cannot eliminate these noises well. CNNs can automatically learn rich and useful features from images and have good performance in image segmentation, classification, target detection, and other tasks. Therefore, in this study, a CNN was used to solve the binary segmentation problem of body pattern images. The DeepOtsu model was proposed by He and Schomaker [28] and mainly solves the document enhancement and binarization problem. Unlike the traditional method of predicting each pixel value, the author proposed a model of learning degradation in images. The model processed the degraded images (x) into uniform images (x_u) using the CNN (Formula (5)), which are noise-free. Then, the images were binarized (Formula (6)) using existing single methods.

$$x_u = \mathrm{CNN}(x) + x \tag{5}$$

$$x_b = \mathrm{B}(x_u) \tag{6}$$

where x_u represents the processed uniform images, x represents the degraded images, B represents an existing binarization method (for example, Otsu), and x_b represents the binarized image.

Because there is also background noise affecting binarization segmentation in the body pattern image, referring to the ideas in the above paper [28], this paper uses U-Net [32] to learn the interference factors in the image and eliminate these negative effects. U-Net is an image segmentation algorithm with a simple convolutional neural network structure, which is also called the encoder–decoder structure. The function of the encoder is to extract the features of different depths of the image, which is realized by convolution and pooling operations. The role of the decoder is to output a segmentation result based on the feature map, which is implemented using upsampling (deconvolution) and feature map concatenation. In the binarization task of cow body pattern images, only two categories are employed, which does not require a very deep or complex network structure. Because the number of images used for training is small, it is easy to cause overfitting by using a large network. U-Net with a simpler structure is sufficient to learn the useful features in cow body pattern images and eliminate noise from the background. The structure of U-Net is shown in Figure 4. After segmentation, a gray image with noise removed is obtained, and then it is processed into a binary image by the Otsu method.

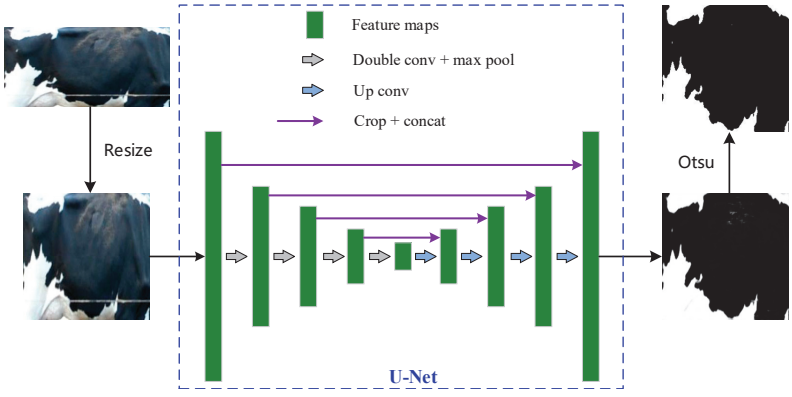


Figure 4. Flowchart of the binarization of a cow body pattern image based on DeepOtsu.

Because the sizes of the body pattern images obtained by the detection model are different, size normalization processing was carried out. The size of all the body pattern images was processed to be 256 (pixels) × 256 (pixels) by using a bicubic interpolation method. The subdataset was used to train and test the DeepOtsu model, the Acc_{seg} index was used to evaluate the segmentation accuracy of the model, and the detection time of a single image was used as the index to evaluate the efficiency of the model. Acc_{seg} is calculated with Formula (7):

$$Acc_{seg} = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

where TP represents the number of correctly segmented white pixels, TN represents the number of correctly segmented black pixels, FP represents the number of incorrectly segmented white pixels, and FN represents the number of incorrectly segmented black pixels. In addition, the two traditional binarization methods were used to binarize the cow body pattern images in the test set, and the accuracy index Acc_{seg} was calculated. By comparing the accuracy of the three methods, we can determine which method is used to binarize the cow body pattern images.

After the completion of the comparative experiment, the remaining body trunk images in the dataset were processed with the best binarization model to obtain the binary body pattern images of all cows.

2.4. Cascaded Classification of Body Pattern Images

For the end-to-end dairy cow individual automatic identification system, the number of dairy cows corresponds to the number of outputs of the individual identification model, and the number of outputs directly affects the quantum parameter and precision of the identification model. In theory, the more output terminals there are, the lower the efficiency and accuracy of the network. In this paper, a cascaded classification method was proposed to reduce the number of outputs of the individual cow identification network. The specific implementation steps are as follows. First, the image was classified according to the proportion of black pixels in the cow body pattern image to realize primary classification. Then, for each subcategory obtained by primary classification, classification was carried out according to the pattern features to realize secondary classification. The cascaded classification method can reduce the number of network parameters without reducing the accuracy, thus improving the efficiency and accuracy of the individual cow identification network.

2.4.1. Primary Classification

As the dataset processed in this study includes 118 cows, it is reasonable to divide the cows into four categories in primary classification. Classification is based on the *B-pro* value, the proportion of black pixels in the binary body pattern image. The images of *B-pro* falling in the interval [0, 0.25) were classified as category I; the images of *B-pro* falling in the interval [0.25, 0.5) were classified as category II; the images of *B-pro* falling in the interval [0.5, 0.75) were classified as category III; and the images of *B-pro* falling in the interval [0.75, 1) were classified as category IV. The primary classification process is shown in Figure 5.

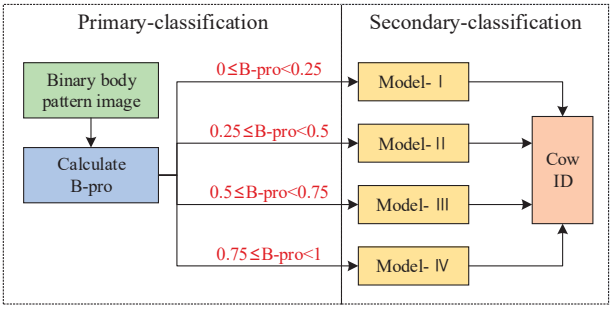


Figure 5. Cascaded classification model.

2.4.2. Secondary Classification

The four subcategories generated by primary classification correspond to the four different secondary classification models. According to the result of primary classification, the image was assigned to the corresponding secondary classification model for individual identification (as shown in Figure 5). Secondary classification was based on the distribution characteristics of black and white patterns in images. Because the binarization process filters out the noise unrelated to classification in the image, secondary classification is relatively simple. The network does not need to determine which features are useful information but only needs to learn and express the features related to classification, such as the distribution area and the boundary trend of the black pattern. However, because the cow is in a state of activity, the position of feature points may change for the same cow's body pattern image, which requires the classification model to have spatial invariance. Therefore, we use EfficientNet to construct the four secondary classification networks.

The basic network architecture of EfficientNet is designed by performing a neural architecture search. EfficientNet consists of three parts. The first part contains a convolution operation, normalization processing, and an activation function whose function is to adjust the number of channels of the input image and to perform preliminary feature extraction. The second part is the main feature extraction structure of EfficientNet, which contains a stack of blocks with seven different parameters. Each block includes several mobile inverted bottleneck Convolution (MBConv) block modules. The MBConv block structure is designed with inverted residuals and ResNet in mind. First, a 1×1 convolution is used to increase the dimension, then a 3×3 or 5×5 depthwise convolution is performed, and an attention mechanism about the channel is added after this structure. Finally, a 1×1 convolution is used to reduce the dimension. The output is connected with the input side to form a residual structure. This is the unique feature extraction structure of EfficientNet, which completes efficient feature extraction in the process of block stacking. The third part of the EfficientNet-B0 network is the prediction head, which contains the convolution layer, pooling layer, and fully connected layer to obtain the final classification results. EfficientNet uses compound scaling to obtain network structures with different depths, widths, and input image sizes. The basic structure of EfficientNet-B0 is shown in Figure 6. Due to the small size of the image, the secondary classification model is selected among EfficientNet-B0, EfficientNet-B1, and EfficientNet-B2, and we determine which model to use based on the training results.

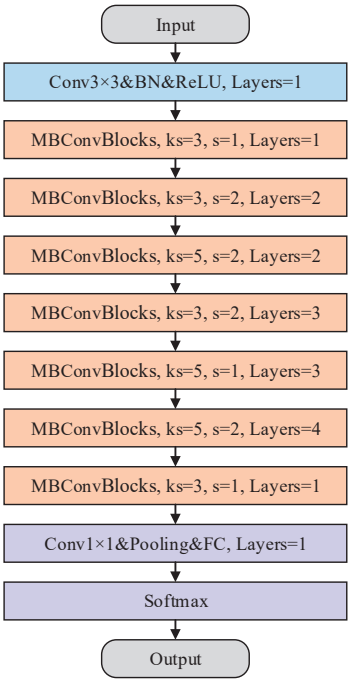


Figure 6. Structure of EfficientNet-B0.

2.4.3. Training and Testing Process

All body pattern images of each cow in the dataset were assigned to the training set, validation set, and test set at a ratio of 5:3:2 to train and test the cascaded classification model. Due to the influence of cow activity, binary segmentation error, and other factors, the proportion of black pixels in the binary body pattern image of the same cow is variable. Therefore, different binary pattern images of the same cow may be assigned to two categories in the process of primary classification. For cows in the above situation, all the

body pattern images of this cow were put into the training set of the corresponding two categories during training to ensure that no matter which category the cow is assigned to, it can be correctly identified. After primary classification, the secondary classification models corresponding to the four categories were trained based on EfficientNet-B0, EfficientNet-B1, and EfficientNet-B2. By comparing their training results, the network structure with higher accuracy was selected as the secondary classification network.

After network training and structure selection, the images in the test set were used to evaluate the performance of the cascaded classification model. The binary body pattern images in the test set were put into the primary classification model first, and then the images were transferred into the corresponding secondary classification model for individual identification. After cascaded classification was completed, the classification accuracy rate Acc_{cls} was used as an index to evaluate the accuracy of the model, and the detection time of a single image was used as an index to evaluate the efficiency of the model. Acc_{cls} is calculated as follows:

$$Acc_{cls} = \frac{true}{true + false} \tag{8}$$

where *true* represents the number of correctly classified samples and *false* represents the number of misclassified samples.

3. Results

3.1. Analysis of Trunk Area Detection Results

The test set in the subdataset was put into the trained YOLOX model to test the performance in cow trunk detection. The results showed that the accuracy evaluation index AP75 value of the detection model reached 0.988, the AP value reached 0.843, and the detection time of a single image was 0.023 s. The YOLOX algorithm can accurately and efficiently obtain the position of the cow trunk from the side-looking walking image of a cow. Figure 7 shows some detection results with different lighting scenes and body patterns. The figure shows that the YOLOX model has good robustness, and that the detection bounding box can contain the trunk area with body patterns, retain the main features used in individual identification, and eliminate interference in the background.

Figure 7. Cow trunk detection results. The red rectangle in the figure represents the detection bounding box of the trunk area, and the text in the upper left of the image represents the category and confidence of the detection bounding box.

3.2. Analysis of the Binarization Results of Body Pattern Images

The test set in the subdataset was put into the traditional binarization models and trained DeepOtsu model, to test the performance in cow body pattern image binarization. The test results of the three methods showed that the DeepOtsu method achieved the highest binarization accuracy of 0.932, the binarization method based on color features achieved an accuracy of 0.877, and Otsu's method based on the gray distribution achieved the lowest accuracy of only 0.827.

Figure 8 shows the binarization results of the three methods for the cow trunk images with interference. The figure shows that the grayscale conversion process reduces the redundant information of the image and filters out some useful information for binarization, resulting in bad body pattern image binarization results. The color feature, as a simple description method, cannot better describe the distribution of black and white body patterns of dairy cows. Therefore, the binarization method based on color features and gray features cannot solve the binarization problem of cow body pattern images in complex scenes. Compared with the other two methods, the DeepOtsu model has obvious advantages and has good robustness to complex interference situations. The DeepOtsu model can remove reflections, stains, shadows, and occlusions in the image through the convolutional neural network to obtain a satisfactory binary image. Therefore, this study used DeepOtsu as a binarization method for cow body pattern images.

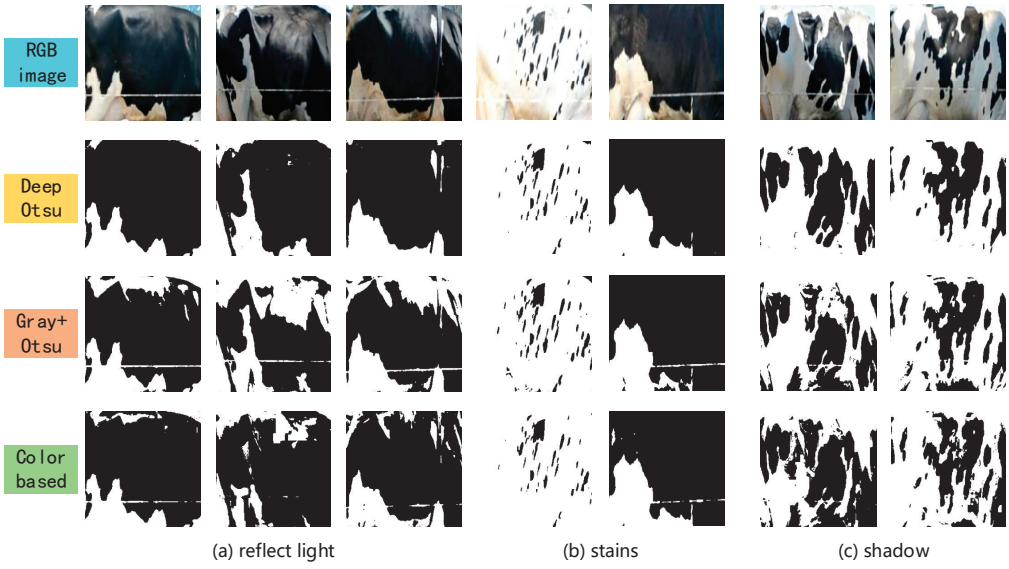


Figure 8. Comparison of three binarization methods under different conditions. In the figure, the images in the first row represent the RGB images to be binarized; the images in the second row represent the images binarized by the DeepOtsu model; the images in the third row represent the images after the grayscale conversion process and Otsu binarization; and the images in the fourth row represent images processed by color-based binarization. (**a**) reflect light (**b**) stains (**c**) shadow.

Figure 9 shows the segmentation results of DeepOtsu model in the presence of interference. The main disturbances that affect the binarization accuracy of the cow body pattern image are as follows.

- The reflection of the black hair area is caused by strong light, which makes the area very bright, as shown in the red rectangle in Figure 9.
- The white electric fence used to limit the walking range of cows leaves a linear white mark on the image of cow body patterns, as shown in the green rectangle in Figure 9.

- The stain in the trunk area makes the area dark, as shown in the yellow rectangle in Figure 9.
- Bright and dark areas are formed by the shadow on the cow, as shown in the blue rectangle in Figure 9.
- Slight overexposure causes the overall image to be brighter, as shown in the last column of Figure 9.

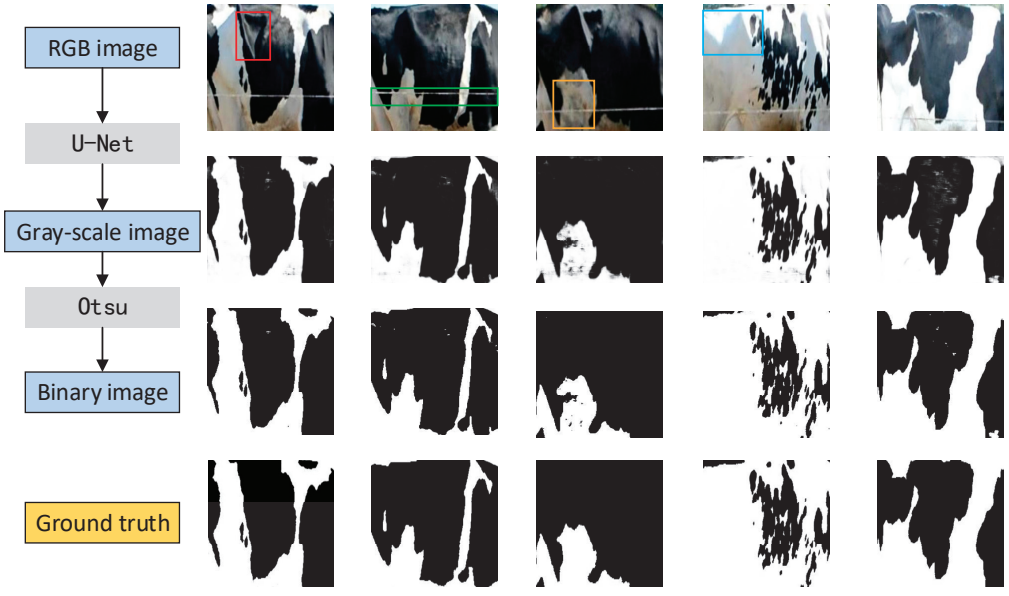


Figure 9. Binarization segmentation results of cow body pattern images. In the figure, the images in the first row are the RGB body pattern images to be processed; the images in the second row are the gray images after U-Net segmentation; the images in the third row are the binary images after Otsu processing; and the images in the fourth row are the ground truths for comparison. The colored rectangular box in the figure marks some areas with interference.

The segmentation results in Figure 9 show that the DeepOtsu model can eliminate different kinds of interferences in the image and output satisfactory binary images of the cow body pattern. By using the convolution neural network U-Net, a relatively 'clean' grayscale cow body pattern image was generated to obtain better binarization results. The binarization process can eliminate the redundant information in the image so that the image only contains the distribution characteristics of black and white patterns. For the individual identification model, the binarization process plays a role in improving the image quality. The binarized cow body pattern image is used as the input of the cascaded classification model, which can make the classification network learn the useful information in the image more quickly and accurately, reduce the complexity of the individual identification model, and make the model adapt to more complex and changeable scenes. Although there are still some small areas that were wrongly segmented in the image, the main features of the black and white pattern distribution were still retained. In the classification process, these misclassified small areas have little effect on the results.

3.3. Analysis of Individual Identification Results of Dairy Cows

3.3.1. Training Results

The proportion values of the black pixels of the binarized cow body trunk images in the training set were counted. According to the proportion values, images were assigned to four categories. The number of cows in each category is shown in Table 1. Different binary pattern images of the same cow may be assigned to two categories due to the changes in *B-pro* values. Therefore, the total number of cows in the four categories is greater than 118. The table shows that the number of cows in categories I and II is less, and the number of cows in categories III and IV is more. Figure 10 shows partial binary cow body pattern images in four categories.

Table 1. Primary classification results.

Index	I	II	III	IV
The number of cows	23	29	49	47

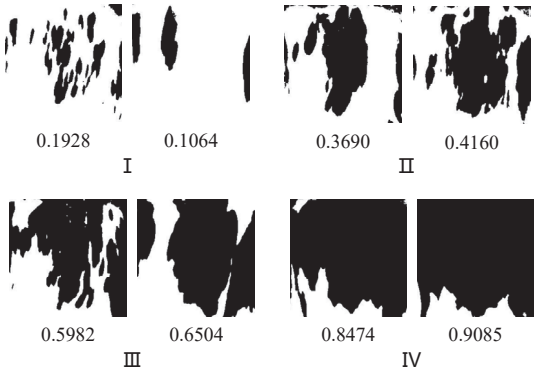


Figure 10. Binarized cow body pattern images in four categories. In the figure, the number below each image represents the proportion of black pixels in that image.

After primary classification was completed, the training sets of different categories were put into EfficientNet-B0, EfficientNet-B1, and EfficientNet-B2 for training. The training results of the four secondary classification models are shown in Table 2. The table shows that for the four secondary classification tasks, the training accuracy of EfficientNet-B1 is better than the training accuracies of EfficientNet-B0 and EfficientNet-B2. At the same time, the training results show that the training accuracy of EfficientNet-B2 is very poor, which may be due to the overfitting of the network caused by the small image size and small amount of data. The depth of the EfficientNet-B1 network is sufficient to extract deep features from the binary cow body pattern image, so EfficientNet-B1 was selected as the secondary classification model.

Table 2. Training accuracy of the four categories.

Model	I	II	III	IV
EfficientNet-B0	1	1	0.985	0.963
EfficientNet-B1	1	1	0.997	0.971
EfficientNet-B2	0.372	0.274	0.125	0.128

3.3.2. Test Results

The images in the test set were put into the cascaded classification model for primary classification and secondary classification, and the classification results and the classification time of a single image were counted. According to the statistics, all the images were classified correctly in primary classification. For secondary classification, the classification results for different categories are shown in Table 3.

Table 3. Test results of secondary classification.

Index	I	II	III	IV	Average
Acc_{cls}	1	1	0.991	0.949	0.985
Classification time for a single image/s	0.389	0.408	0.412	0.412	0.405

The table shows that the classification accuracy rate of categories I and II is 1, the classification accuracy rate of category III is the second highest, and the classification accuracy of category IV is the lowest. The number of output ends of categories I and II is relatively small. Figure 10 shows that the proportion values of black pixels in the body pattern images belonging to categories I and II are relatively low, so the distribution characteristics of black and white patterns are rich. Therefore, the accuracy of these two categories reaches 100%. The number of cows belonging to category III is almost twice that belonging to categories I and II, so the classification accuracy is slightly lower. However, the distribution features of black and white patterns in the binary speckle image are still relatively rich, so its classification accuracy is also very high. The number of cows belonging to category IV is also relatively large. Figure 10 shows that the images in category IV have a relatively high proportion of black pixels, and most of the images have large black areas. The areas with distinguishable feature points are small and generally located at the bottom or corners of the image, so the overall classification accuracy of the images in category IV is slightly low. In addition, the reflection of the black hair area is the main reason for the reduced binarization accuracy. Obviously, the cows belonging to category IV have relatively more black hair area in their body pattern and more binarization errors, which makes the corresponding classification accuracy lower. Overall, the average classification accuracy of the four secondary-classification models is 0.985, which achieved high accuracy in individual identification.

In addition, from the classification results of the four categories, the number of outputs affects the accuracy of the classification model. Reducing the number of outputs of the classification model can improve the process accuracy and speed of the individual cow identification model, and the resulting model has better recognition ability for cows with similar body patterns.

4. Discussion

4.1. Comparison between the Cascaded Method and End-to-End Method

To compare the cascaded identification method with the end-to-end identification method, all RGB body pattern images of each cow in the dataset were used to construct the training set, validation set, and test set at a ratio of 5:3:2, and the end-to-end identification model based on EfficientNet-B1 was trained. Table 4 shows the identification accuracy and speed of the end-to-end method and the cascaded method.

Table 4. Identification accuracies and speeds of different methods.

Index	Cascaded Method	End-to-End Method
Acc_{cls}	0.985	0.987
Identification time of a single image/s	0.405	0.432

Table 4 shows that the end-to-end individual identification method and the cascaded individual identification method achieve the same high accuracy, which is above 0.98. However, because the cascaded individual identification method reduces the number of outputs of each secondary classification model, the number of parameters of the cascaded individual identification model is less than that of the end-to-end individual identification model, so the processing speed of the cascaded individual identification method is slightly higher than that of the end-to-end individual identification method.

In practical applications, when a new cow joins the dairy farm, the recognition model needs to be retrained. Therefore, this paper counts the training time of different individual identification methods, as shown in Table 5. For the cascaded individual identification method, only one or two secondary classification models need to be retrained when a new cow is added (in most cases, only one model needs to be retrained). For the end-to-end recognition method, the entire model needs to be retrained when a new cow is added. According to the comparison of training time in Table 5, the training time of the cascaded individual identification method is shorter than that of the end-to-end individual identification method.

Table 5. Training times of different individual identification methods.

Index	Cascaded Method				End-to-End Method
	I	II	III	IV	EfficientNet-B1
Training time/min	32	39	70	66	132

4.2. Error Analysis

In this paper, the statistics and analysis of the error individual identification results were carried out. Figure 11 shows the two cows with the lowest individual identification rates in the dataset, and their individual identification rates are both 0.75. After analysis, the reasons for the low identification rate include the following two aspects: (1) The two cows belong to category IV, meaning that the distribution area of black and white patterns in the trunk area is very small, and fewer corresponding identification features exist. (2) The distribution of black and white patterns is concentrated in the bottom area of the trunk, and leg movement will change the distribution and shape of the patterns when the cow walks, thus affecting the secondary classification accuracy. Because of the small number of samples in the training set, these changes cannot be learned by the secondary classification model, which is also one of the reasons for the low individual identification rate.

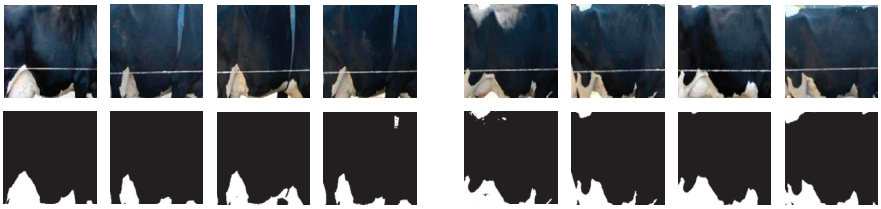


Figure 11. Body pattern images of two cows (**a**,**b**) with the lowest identification rate.

4.3. Comparison of the Proposed Method with Similar Studies

In order to show the advantages of the proposed method more intuitively, a comparison with other identification method based on body pattern images [8,20,25,26,33,34] was conducted as illustrated in Table 6. It can be seen from Table 6 that the number of cows in the dataset of this paper is the largest, and the identification accuracy of the proposed method exceeds most of the references in the table. Although the accuracy in [20,33] is higher than our proposed method in this paper, the dataset of [20] contains only 10 cows,

and the number of cows is very small. It is needed to collect cow images from four perspectives in [33], so the time and labor cost of collecting data are high. In summary, the cascaded individual cow identification method proposed in this paper has obvious advantages over the other publishing similar research and has the potential to be applied to large-scale automated pastures.

Table 6. Comparison between our proposed method and some of state-of-the-art methods in term of image source, identification accuracy, and number of cows in the datasets.

Reference	Image Source	Identification Accuracy	Number of Cows
[8]	Side view images of cow	98.36%	93
[20]	Tailhead images	99.7%	10
[25]	Back images of cow	95.91%	89
[33]	Back image, left side profile image, right side profile image, facial image	99%	51
[26]	Side view images of cow	96.65	105
[34]	Body pattern images (top view)	93.8	46
Our method	Body pattern images (side view)	98.5	118

4.4. Future Research

Although our proposed cascaded method can achieve fast and accurate individual identification of dairy cows, there is still room for improvement. For the binary segmentation of cow trunk images, severely overexposed images were removed when constructing the dataset. However, in an actual production environment, overexposure occasionally occurs. Therefore, in future studies, we can improve the robustness of the binarization model by optimizing the algorithm network so that the cascaded dairy individual cow identification method can adapt to more complex scenes on farms. In addition, due to the limitation of data collection conditions, the number of cows in the dataset constructed in this paper is relatively small, and the number of samples per cow is also relatively small. In future studies, the data can be collected on a large-scale dairy farm with more cows. The proposed method can be applied to farms to further verify the superiority of the method compared with the end-to-end identification method and its potential application on large-scale dairy farms.

5. Conclusions

In this paper, a method of cascaded individual dairy cow identification based on DeepOtsu and EfficientNet was proposed. The body pattern images of dairy cows were binarized and cascaded classified to address the identification difficulty caused by similar body pattern characteristics, poor image quality, and a large number of output terminals in dairy cow group identification. The test results of the method showed that the detection accuracy (AP75) of the cow trunk based on YOLOX is 0.988, and the detection time of a single image is 0.023 s; the binarization accuracy of cow body pattern images based on DeepOtsu is 0.932, and the binarization time of a single image is 0.005 s. The classification accuracy of the cascaded classification model is 0.985, and the classification time of a single image is 0.405 s. The overall individual identification accuracy of the proposed method is 0.985, and the identification time of a single image is 0.433 s. Compared with the end-to-end individual identification method, the proposed method has obvious advantages in identification efficiency and training speed. The proposed method provides a new approach to dairy cattle group individual identification in large-scale dairy farms.

Author Contributions: Conceptualization, R.Z., J.J. and K.Z.; Methodology, R.Z., K.Z. and M.Z.; Software, R.Z., J.W. and M.Z.; Validation, R.Z.; Formal analysis, R.Z. and J.J.; Investigation, K.Z.; Resources, R.Z., J.J. and K.Z.; Data curation, K.Z. and J.W.; Writing—original draft, R.Z.; Writing—review and editing, K.Z. and M.W.; Visualization, R.Z. and J.W.; Supervision, J.J.; Project administration, K.Z.; Funding acquisition, J.J. and M.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by "National Key R&D Plan Key projects of Scientific and technological Innovation Cooperation between Governments", grant number "2019YFE0125600"; "National Natural Science Foundation of China", grant number 32002227"; and "Natural Science Basic Research Plan in Shaanxi Province of China", grant number "2022JQ-175".

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors acknowledge University of Kentucky for facilitation of data acquisition and permission for data use. This study was supported by the National Natural Science Foundation of China (grant No. 32002227), the National Key R&D Plan Key projects of Scientific and technological Innovation Cooperation between Governments (grant No. 2019YFE0125600), and the Natural Science Basic Research Plan in Shaanxi Province of China (grant No. 2022JQ-175).

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

References

1. Rowe, E.; Dawkins, M.S.; Gebhardt-Henrich, S.G. A systematic review of precision livestock farming in the poultry sector: Is technology focussed on improving bird welfare? *Animals* **2019**, *9*, 614. [CrossRef] [PubMed]
2. Tullo, E.; Finzi, A.; Guarino, M. Review: Environmental impact of livestock farming and precision livestock farming as a mitigation strategy. *Sci. Total Environ.* **2018**, *650*, 2751–2760. [CrossRef] [PubMed]
3. Sébastien, F.; Rousseau, A.N.; Laberge, B. Rethinking environment control strategy of confined animal housing systems through precision livestock farming-sciencedirect. *Biosyst. Eng.* **2017**, *155*, 96–123. [CrossRef]
4. Sun, Y.; Wang, Y.; Huo, P.; Cui, Z.; Zhang, Y. Research progress on methods and application of dairy cow identification. *J. China Agric. Univ.* **2019**, *24*, 62–70. [CrossRef]
5. Porto, S.M.C.; Arcidiacono, C.; Giummarra, A.; Anguzza, U.; Cascone, G. Localisation and identification performances of a real-time location system based on ultra wide band technology for monitoring and tracking dairy cow behaviour in a semi-open free-stall barn. *Comput. Electron. Agric.* **2014**, *108*, 221–229. [CrossRef]
6. Gygax, L.; Neisen, G.; Bollhalder, H. Accuracy and validation of a radar-based automatic local position measurement system for tracking dairy cows in free-stall barns. *Comput. Electron. Agric.* **2007**, *56*, 23–33. [CrossRef]
7. Huhtala, A.; Suhonen, K.; Mäkelä, P.; Hakojärvi, M.; Ahokas, J. Evaluation of instrumentation for cow positioning and tracking indoors. *Biosyst. Eng.* **2007**, *96*, 399–405. [CrossRef]
8. Hu, H.; Dai, B.; Shen, W.; Wei, X.; Sun, J.; Li, R.; Zhang, Y. Cow identification based on fusion of deep parts features. *Biosyst. Eng.* **2020**, *192*, 245–256. [CrossRef]
9. Qiao, Y.; Guo, Y.; Yu, K.; He, D. C3D-ConvLSTM based cow behaviour classification using video data for precision livestock farming. *Comput. Electron. Agric.* **2022**, *193*, 106650. [CrossRef]
10. Zhao, K.; Liu, X.; Ji, J. Automatic Body Condition Scoring Method for Dairy Cows Based on EfficientNet and Convex Hull Feature of Point Cloud. *Trans. Chin. Soc. Agric. Mach.* **2021**, *52*, 192–201+73. [CrossRef]
11. Ji, J.; Liu, X.; Zhao, K. Automatic rumen filling scoring method for dairy cows based on SOLOv2 and cavity feature of point cloud. *Trans. CSAE* **2022**, *38*, 186–197. [CrossRef]
12. Zhao, K.; He, D.; Wang, E. Detection of Breathing Rate and Abnormity of Dairy Cattle Based on Video Analysis. *Trans. Chin. Soc. Agric. Mach.* **2014**, *45*, 258–263. [CrossRef]
13. Mahmud, M.S.; Zahid, A.; Das, A.K.; Muzammil, M.; Khan, M.U. A systematic literature review on deep learning applications for precision cattle farming. *Comput. Electron. Agric.* **2021**, *187*, 106313. [CrossRef]
14. Zhao, K.; He, D. Recognition of individual dairy cattle based on convolutional neural networks. *Trans. CSAE* **2015**, *31*, 181–187.
15. Zhao, K.; Jin, X.; Ji, J. Individual identification of Holstein dairy cows based on detecting and matching feature points in body images. *Biosyst. Eng.* **2019**, *181*, 128–139. [CrossRef]
16. Cong, S.; Wang, J.; Zhang, R.; Zhao, L. Cattle identification using muzzle print images based on feature fusion. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *853*, 012051. [CrossRef]
17. Lu, Y.; He, X.; Wen, Y.; Wang, P.S.P. A new cow identification system based on iris analysis and recognition. *Int. J. Biom.* **2014**, *6*, 18–32. [CrossRef]
18. Yang, S.; Liu, Y.; Wang, Z.; Han, Y.; Wang, Y.; Wang, Y.; Lan, X. Improved YOLO V4 model for face recognition of diary cow by fusing coordinate information. *Trans. CSAE* **2021**, *37*, 129–135. [CrossRef]

19. Achour, B.; Belkadi, M.; Filali, I.; Laghrouche, M.; Lahdir, M. Image analysis for individual identification and feeding behaviour monitoring of dairy cows based on Convolutional Neural Networks (CNN). *Biosyst. Eng.* **2020**, *198*, 31–49. [CrossRef]
20. Li, W.; Ji, Z.; Wang, L.; Sun, C.; Yan, X. Automatic individual identification of Holstein dairy cows using tailhead images. *Comput. Electron. Agric.* **2017**, *142*, 622–631. [CrossRef]
21. Okura, F.; Ikuma, S.; Makihara, Y.; Muramatsu, D.; Nakada, K.; Yagi, Y. RGB-D video-based individual identification of dairy cows using gait and texture analyses. *Comput. Electron. Agric.* **2019**, *165*, 104944. [CrossRef]
22. Zin, T.T.; Pwint, M.Z.; Seint, P.T.; Thant, S.; Misawa, S.; Sumi, K.; Yoshida, K. Automatic Cow Location Tracking System Using Ear Tag Visual Analysis. *Sensors* **2020**, *20*, 3564. [CrossRef] [PubMed]
23. Zhang, R.; Zhao, K.; Ji, J.; Zhu, X. Automatic location and recognition of cow's collar ID based on machine learning. *J. Nanjing Agric. Univ.* **2021**, *44*, 586–595. [CrossRef]
24. Zhao, K.; Zhang, R.; Ji, J. A Cascaded Model Based on EfficientDet and YOLACT++ for Instance Segmentation of Cow Collar ID Tag in an Image. *Sensors* **2021**, *21*, 6734. [CrossRef]
25. He, D.; Liu, J.; Xiong, H.; Lu, Z. Individual Identification of Dairy Cows Based on Improved YOLO v3. *Trans. Chin. Soc. Agric. Mach.* **2020**, *51*, 250–260. [CrossRef]
26. Shen, W.Z.; Hu, H.Q.; Dai, B.S.; Wei, X.L.; Sun, J.; Jiang, L.; Sun, Y.K. Individual identification of dairy cows based on convolutional neural networks. *Multimed. Tools Appl.* **2020**, *79*, 14711–14724. [CrossRef]
27. Tan, M.; Le, Q.V. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:11946.
28. He, S.; Schomaker, L. DeepOtsu: Document enhancement and binarization using iterative deep learning. *Pattern Recognit.* **2019**, *91*, 379–390. [CrossRef]
29. Liu, J.; Jiang, B.; He, D.; Song, H. Individual recognition of dairy cattle based on Gaussian mixture model and CNN. *Comput. Appl. Softw. 2018. 35*, 159–164. [CrossRef]
30. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
31. Ge, Z.; Liu, S.; Wang, F.; Sun, J. YOLOX: Exceeding YOLO series in 2021. *arXiv* **2021**, arXiv:2107.08430.
32. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Cham, Switzerland, 2015; pp. 234–241.
33. De Lima Weber, F.; de Moraes Weber, V.A.; Menezes, G.V.; Junior, A.d.S.O.; Alves, D.A.; de Oliveira, M.V.M.; Matsubara, E.T.; Pistori, H.; de Abreu, U.G.P. Recognition of Pantaneira cattle breed using computer vision and convolutional neural networks. *Comput. Electron. Agric.* **2020**, *175*, 105548. [CrossRef]
34. Andrew, W.; Gao, J.; Mullan, S.; Campbell, N.; Dowsey, A.W.; Burghardt, T. Visual identification of individual Holstein-Friesian cattle via deep metric learning. *Comput. Electron. Agric.* **2021**, *185*, 106133. [CrossRef]

agriculture

MDPI

Article

Utility of Deep Learning Algorithms in Initial Flowering Period Prediction Models

Guanjie Jiao [1], Xiawei Shentu [2], Xiaochen Zhu [2,*], Wenbo Song [3], Yujia Song [2] and Kexuan Yang [2]

[1] School of Environmental Science and Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2] School of Applied Meteorology, Nanjing University of Information Science and Technology, Nanjing 210044, China
[3] School of Artificial Intelligence (School of Future Technology), Nanjing University of Information Science and Technology, Nanjing 210044, China
* Correspondence: xiaochen.zhu@nuist.edu.cn

Abstract: The application of a deep learning algorithm (DL) can more accurately predict the initial flowering period of *Platycladus orientalis* (L.) Franco. In this research, we applied DL to establish a nationwide long-term prediction model of the initial flowering period of *P. orientalis* and analyzed the contribution rate of meteorological factors via Shapely Additive Explanation (SHAP). Based on the daily meteorological data of major meteorological stations in China from 1963–2015 and the observation of initial flowering data from 23 phenological stations, we established prediction models by using recurrent neural network (RNN), long short-term memory (LSTM) and gated recurrent unit (GRU). The mean absolute error (MAE), mean absolute percentage error (MAPE), and coefficient of determination (R^2) were used as training effect indicators to evaluate the prediction accuracy. The simulation results show that the three models are applicable to the prediction of the initial flowering of *P. orientalis* nationwide in China, with the average accuracy of the GRU being the highest, followed by LSTM and the RNN, which is significantly higher than the prediction accuracy of the regression model based on accumulated air temperature. In the interpretability analysis, the factor contribution rates of the three models are similar, the 46 temperature type factors have the highest contribution rate with 58.6% of temperature factors' contribution rate being higher than 0 and average contribution rate being 5.48×10^{-4}, and the stability of the contribution rate of the factors related to the daily minimum temperature factor has obvious fluctuations with an average standard deviation of 8.57×10^{-3}, which might be related to the plants being sensitive to low temperature stress. The GRU model can accurately predict the change rule of the initial flowering, with an average accuracy greater than 98%, and the simulation effect is the best, indicating that the potential application of the GRU model is the prediction of initial flowering.

Keywords: *P. orientalis*; recurrent neural network; inverse distance weighting; accumulated air temperature

Citation: Jiao, G.; Shentu, X.; Zhu, X.; Song, W.; Song, Y.; Yang, K. Utility of Deep Learning Algorithms in Initial Flowering Period Prediction Models. *Agriculture* **2022**, *12*, 2161. https://doi.org/10.3390/agriculture12122161

Academic Editor: Maciej Zaborowicz

Received: 24 October 2022
Accepted: 12 December 2022
Published: 15 December 2022

1. Introduction

Flowering is one of the sensitive indicators for assessing climate change [1–6], which reflects changes in surface vegetation and eco-health [7]. Moreover, flowering has tremendous economic value; plants with short flowering time displays have promoted the development of tourism, and tourism activities characterized by flower viewing have gradually become important cultural events, and the market is constantly expanding [8–10].

The climatic conditions have an impact on the initial flowering period of plants, and air and soil temperature are the main factors [5,11–13]. Important progress has been made in phenological research on flowering forecasting based on meteorological data, which has mainly established statistical equations to predict flowering period based on the correlation between meteorological data and phenological data [3,14,15]. In 1974, Richardson et al. [16]

first proposed the application of the chill unit model to research on peach tree dormancy prediction, which calculates the chill unit accumulation coinciding with the completion of plant dormancy to evaluate the impact of low temperature in winter on flowering and to predict the initial flowering period. In 1979, White [17] constructed a linear regression model based on phenological data from 53 species of Montana, which support subsequent flowering studies. In 1986, Anderson et al. [18] further obtained the ASYMCUR GDH model that is an improved normal plant model to fit growing divide hour (GDH) responding to the environment on the basis of the chill unit model, and carried out research on the prediction of the tart cherry flowering period. In 1998, to avoid damage to plants caused by frost and hazards brought by climate change, such as rising temperature, Hakkinen et al. [19] used nearly 60 years of phenological data of birth bud observation in southern Finland from 1896–1955 and meteorological data of light signal and air temperature to predict the bud burst of birch trees by the light and temperature driven model. In recent years, as data work has continued to improve, flowering forecasting has begun to focus on accurate predictive models applied to a wide range of flowers. In 2004, Demeloabreu [20] carried out flowering prediction for different olive varieties in multiple regions to analyze the impact of global warming on olive production. Soil moisture is an important factor affecting spring phenology, so Yashvir et al. [21] utilized soil moisture as a correction factor to improve the accuracy of the original chickpea flowering prediction model in 2019. The research on flowering period in China focuses on analyzing the mechanism of meteorological influence on flowering. In 2019, Wu D et al. [22] conducted analysis through the forecasting model of apple flowering in Shaanxi, which refers to prediction of flowering period at different stations and analysis of the applicability by using the mechanistic models to simulate the growth process of phenology. In 2021, Tan J et al. [23] conducted a fine fitting analysis of cherry flowering and concluded that air temperature and precipitation are the main impact factors of previous cherry period research at Wuhan University.

Current research on flowering forecasting has problems, such as the limited time and space range of accurate predictions and uncertainty around meteorological factors affecting flowering, and currently the demand for initial flowering periods of plants in the Chinese flowering market covers the whole country. A solution for spatial phenology modelling may be to model phenology using herbarium and Citizen Science records and gridded climatic data. Recently, the flowering of Anemone nemorosa was modelled in this way across Europe. However, this approach has some limitations related to the availability of replicated phenological observations and spatial and taxonomic biases [24]. Hence, long-term local monitoring data are still invaluable in phenological studies. With the in-depth integration of artificial intelligence (AI) and meteorological big data, more scholars have begun to pay attention to the application of machine learning (ML) in phenology [25], but detailed research on deep learning (DL) in flowering prediction is lacking.

In our research, we demonstrate the capabilities of deep learning algorithms that have so far been used to a limited extent in phenological research. We believe that the results obtained in our study will find wide application and contribute to a better understanding of the phenological response of plants to meteorological conditions. We also analyze the contribution of each factor via Shapely Additive Explanation (SHAP) to interpret the deep learning model. We expect to provide a scientific basis for nationwide long-term, data-driven flowering prediction models based on our research.

2. Materials and Methods

2.1. Studied Species

P. orientalis (Cupressaceae) is also named tujia or arborvitae. Its initial flowering period is from March to April, and its cones mature in October.

P. orientalis has good stress resistance, which can withstand various extreme environmental conditions [26,27], such as drought, high temperature and low temperature stress, etc. However, the geographical advantages of abundant rainfall and high humidity in southern China can ensure its more healthy growth [28].

P. orientalis is one of the most widely distributed plants produced in southern Inner Mongolia, Jilin, Liaoning, Hebei, Shanxi, Shandong, Jiangsu, Zhejiang, Fujian, Anhui, Jiangxi, Henan, Shaanxi, Gansu, Sichuan, Yunnan, Guizhou, Hubei, Hunan, northern Guangdong and northern Guangxi in China [29].

2.2. Region

The research is to establish prediction models of initial flowering period of *P. orientalis* in China in the Chinese region (73°33′ E-135°05′ E, 3°51′ N-53°33′ N). Due to the large area, the distribution of meteorological elements in China is complex, mainly reflected in the uneven distribution of air temperature, precipitation and humidity, etc.

Temperature regions are divided by accumulated temperature. In China, there are five temperature regions of tropical, subtropical, warm temperate, mesotemperate and cold, whose accumulated temperature value is increasing from north to south, so lower latitude affects the growth process of plants less. There is also a special Qinghai Tibet Plateau region influenced by high altitude of an average 4000 m [30].

According to the humidity index (HI), China can be divided into four regions, namely, arid region (AR), semi-arid region (SAR), semi-humid region (SHR) and humid region (HR) [31]. And HI can reflect the regional humidity, which affects the physiological process of plants through the influence on the water potential, which is the key to the process of plant water absorption. The partition result can be obtained in Figure 1.

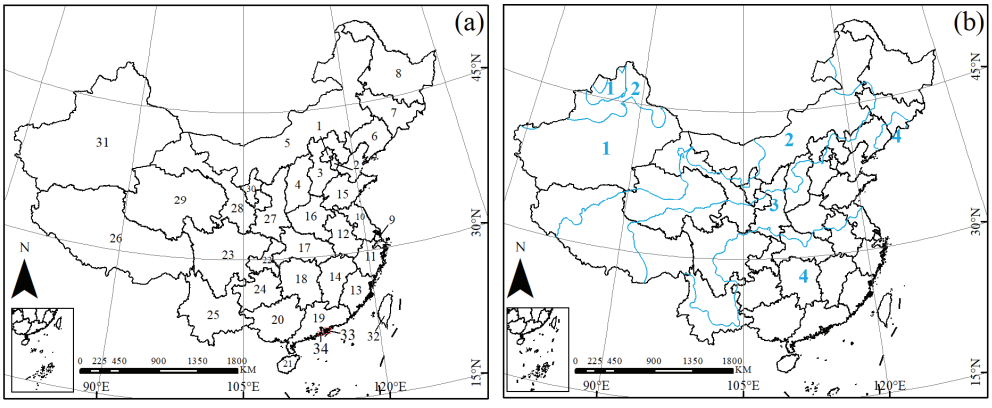


Figure 1. China's regional map of (**a**) administrative division, and (**b**) arid-humid division. Beijing, Tianjing, Hebei, Shanxi, Inner Mongolia, Liaoning, Jilin, Heilongjiang, Shanghai, Jiangsu, Zhejiang, Anhui, Fujian, Jiangxi, Shandong, Henan, Hubei, Hunan, Guandong, Guangxi, Hainan, Chongqing, Sichuan, Guizhou, Yunnan, Tibet, Shaanxi, Gansu, Qinghai, Ningxia, Xinjiang, Taiwan, Hongkong and Macao are denoted by numbers from 1 to 34 in (**a**), respectively. The arid region (AR), semi-arid region (SAR), semi-humid region (SHR) and humid region (HR) are denoted by numbers 1, 2, 3 and 4, respectively.

The analysis of the impact of China's meteorological element conditions on the spatial distribution of *P. orientalis* in the initial flowering period is regional, so we introduced China's administrative division to help spatial analysis. The vector diagram of the division of administrative regions in China is derived from the National Platform for Common Geospatial Information Services (https://www.tianditu.gov.cn (accessed on 8 September 2022)).

2.3. Materials

Phenological data are observational data that reflect periodic biological phenomena including initial flowering period, which refers to the time of one or few flowers fully open. To obtain enough data for DL training, we collected the initial flowering data of *P. orientalis*

from the National Earth System Science Data Center (https://geodata.cn/ (accessed on 13 July 2022)) and the Earth Big Data Science Data Center of the Chinese Academy of Sciences (https://data.casearth.cn/ (accessed on 11 August 2022)) and selected available data that included city stations in Baoding, Beijing, Changde, Guiyang, Hohhot, Shanghai, Foshan, Nanjing, Nanchang, Hefei, Harbin, Kunming, Guilin, Wuhan, Minqin, Shenyang, Tai'an, Xi'an, Chongqing, Yinchuan, Changchun, Changsha and Yancheng from 1961–2015. The spatial distribution can be obtained in Figure 2. A total of 357 valid data points were obtained.

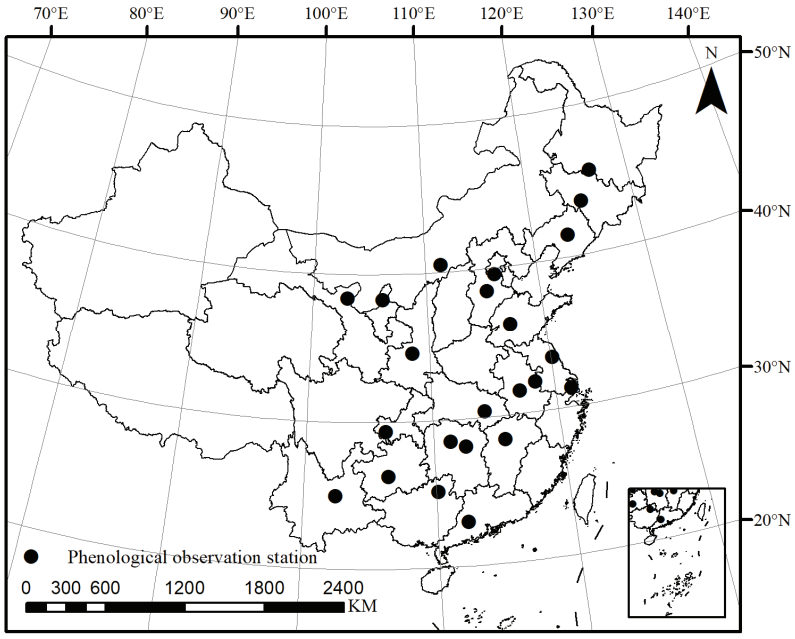


Figure 2. Geographical distribution map of phenological stations.

The meteorological data were obtained from the China Meteorological Science Data Sharing Network "China Ground Meteorological Data Dataset V3.0". A total of 23 basic city stations were selected, and we obtained the meteorological elements of average temperature (°C), daily minimum temperature (°C), daily maximum temperature (°C), daily average ground temperature (°C), daily average precipitation (mm), daily average sunshine hours (h), daily average relative humidity (%), and daily average pressure (Pa) from 1 January to 30 April 1961 to 2015 in these stations.

2.4. Methods

2.4.1. Selection of Meteorological Factors

The effect of air temperature on the initial flowering period is most pronounced, followed by sunshine and precipitation [11,21]. In ecological research, crop growth and development need to accumulate to a certain sum of temperature, so the air temperature is usually expressed in cumulative amount, which is referred to as the accumulated temperature. According to different time scales, the action time of accumulated temperature is varied. In the process of growth, crops respond to the temperature limit, which is the lower limit temperature. When the temperature is lower than the lower limit temperature, the plants will not grow and develop. The accumulated amount of temperature above the lower limit temperature is the active accumulated temperature, and the accumulated difference between the temperature and the lower limit temperature is the effective accumulated temperature, which can be applied to air temperature and ground temperature.

$$Effective\ accumulated\ temperature = \sum (T_i - C_0) \quad (1)$$

$$Accumulated\ temperature = \sum T_i \quad (2)$$

$$Active\ accumulated\ temperature = \sum T_i \quad T_i \geq C_0 \quad (3)$$

where T_i is the daily average temperature, and C_0 is the lower limit temperature.

Since the initial flowering period of Platycladus orientails is mainly in the middle of April, we focused on the meteorological data from January to April. During data processing, we read the meteorological data from each station and used 0 °C, 3 °C, 5 °C AND 10 °C as the lower limit temperatures to calculate the effective accumulated temperature and counted the accumulated temperature and average temperature from January to early April for ten days and the average ground temperature monthly and other factors, as detailed in Table 1.

Table 1. Table of meteorological factors affecting the initial flowering of *P. orientalis*.

Meteorological Elements	Meteorological Factors	Number of Factors
Temperature	1. The effective cumulative temperature of 0 °C, 3 °C, 5 °C, 10 °C (°C); 2. Active temperature (°C); 3. Accumulated temperature (°C); 4. Accumulated temperature for ten days (°C); 5. Average temperature for ten days (°C); 6. Days when the minimum/maximum temperature is less than 0 °C, 5 °C, 10 °C (d); 7. Days when the minimum/maximum temperature is more than 0 °C, 5 °C, 40 °C (d); 8. Average monthly minimum/maximum temperature from January to April (°C).	46
Ground temperature	1. Accumulate ground temperature (°C); 2. Average monthly ground temperature from January to April (°C); 3. Days when the ground temperature is less than 0 °C (d); 4. Days when the ground temperature is more than 40 °C (d).	7
Precipitation	1. Cumulative precipitation (mm); 2. Average precipitation (mm); 3. Accumulated monthly precipitation from January to April (mm); 4. Average monthly precipitation from January to April (mm).	10
Hours of sunshine	1. Total hours of sunshine (h); 2. Monthly hours of sunshine from January to April (h).	5
Relative humidity	1. Average relative humidity (%); 2. Average relative humidity for ten days (%).	11
Pressure	1. Average pressure for ten days (hPa).	10

Because different meteorological data have different degrees of influence [23], we considered different time resolutions when establishing meteorological factors. For example, we mainly deal with accumulated temperature for ten days when doing accumulated temperature calculation through Equation (2), which is a method to calculate ten days of the month. Therefore, each month will have three different accumulated temperatures for ten days values, which are divided into an early value, middle value and late value.

2.4.2. Data Processing

For the convenience of comparison between two different years, we use the data of ordinal number from 1 January to the current date as phenological data of flowering.

With each meteorological factor as the independent variable and ordinal number as the dependent variable, a phenological-meteorological dataset is constructed, and the dataset is normalized to facilitate weight distribution in the deep learning model. At a ratio of 7:3, we divided the training dataset and test dataset for model training and modelled effect evaluation to ensure sufficient samples during training, whose distribution is the same and not repeated, to evaluate the quality of model training.

In order to make each factor value dimensionless in the process of DL training, we normalized the data by max–min method, which will limit each data point to 0–1.

$$y' = \frac{y - \min}{\max - \min} \tag{4}$$

where y' is normalized value, y is value to be normalized, min is the minimum value of the same value and max is the maximum value of the same value.

2.4.3. Deep Learning Model

In current prediction research, such as Southern Oscillation, local evaporation and drought prediction, the deep learning algorithm has a better fitting ability and can improve the spatial resolution of prediction [32,33]. Compared with other common networks such as convolutional neural network (CNN) and artificial neural network (ANN), recurrent neural networks have a significant role in time series processing. The initial flowering period is predicted by three common deep learning prediction models, namely, the recurrent neural network (RNN), long short-term memory (LSTM), and the gated recurrent unit (GRU).

- Compared with other neural networks, the RNN can predict the current input value by combining the input values of the first N time series, that is, it has correlation in the time series.
- LSTM can learn the long-term dependence between two variables and retain the error, which can be maintained at a constant level when backpropagation is carried out along the time layer [34,35]. LSTM is equipped with three gating devices to filter the input data, namely, the input gate, forget gate and output gate. The forget gate will generate a value between 0 and 1 according to the output and current input of the previous time to decide whether to retain the information of the previous time [35]. The time function of the forget gate is mainly controlled by the sigmoid activation function:

$$f_t = \sigma(W_f \cdot [h_t - 1, x_t] + b_f) \tag{5}$$

  where f is the forget gate, W is the weight matrix, b_f is the offset term, and σ is the sigmoid activation function. The closer the value of f_t is to 0, the more items will be forgotten.
- Compared with the LSTM model, the GRU simplifies the calculation steps and substantially increases the training speed, while the GRU also uses a gate device to filter information, namely, the reset gate and update gate. In the process of training, the input information will not be cleared by the gate device, but the necessary information will be retained in the next cycle, and the information will be saved to avoid the problem of gradient disappearance. Since there are only two gate structures, the actual running time of the GRU model is substantially less than that of LSTM with fewer network parameters, so the risk of GRU model overfitting is smaller under the condition of ensuring accuracy.

2.4.4. Training Effect Indicators

The mean squared error (MSE) is used as a loss function, and the mean absolute error (MAE), mean absolute percentage error (MAPE) and coefficient of determination (R^2) are utilized as the training effect indicators to evaluate the model performance.

$$\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{6}$$

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m}|(y_i - \hat{y}_i)| \tag{7}$$

$$\mathrm{MAPE} = \frac{100\%}{m} \times \sum_{i=1}^{m}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{8}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \overline{y}_i)^2} \tag{9}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, m is the number of samples, and $\overline{y}_i$ is the mean of the prediction.

MSE has high robustness, and it can effectively converge with a fixed learning rate, so the model with MSE as the loss function can maintain the accuracy in the process of convergence compared with the model with MAE as loss function [36]. MAE and MAPE are commonly employed indicators to reflect the degree of deviation between the predicted value and the true value. R^2 is mainly used to judge the linear relationship between the model prediction and the true value. Therefore, when the value of R^2 is near 1, the simulation degree of the model is accurate. The above four indicators are applied as mathematical definitions in general statistical research, so they are highly recognized.

2.4.5. Interpretability Model Based on SHAP

Shapely Additive Explanation (SHAP) is a method which uses game theory that is used to study the mathematical theory of contribution rate as the ideological carrier to calculate the impact of the characteristic variables of sample data on the results of the prediction model and then to measure the contribution of these characteristic variables. This approach explains the CART-based complex integrated learning model [37].

The core of SHAP is to calculate the Shapley value of variables, which represents the importance of determining the influence of various factors on the prediction.

$$\phi_i = \sum_{S \subseteq M \setminus \{i\}} \frac{|S|!(|M|-|S|-1)!}{|M|!}[f_x(S \cup \{i\}) - f_x(S)] \tag{10}$$

where M denotes all feature sets S represents subsets of i, $f_x(S \cup \{i\})$ is the predicted value of the characteristic variable containing only $S \cup \{i\}$ in the sample data, and has a Shapley value of i.

As the complexity of using the Shapley value to traverse all subsets exponentially increases, this leads to an excessively long computing time and increases the computational burden, Lundberg and Lee proposed the Tree SHAP model based on the tree model in machine learning combined with the Shapley value [37]. In this research, we used the Deep SHAP model interpreter to rank the contribution of 89 meteorological factors that affect the initial flowering of *P. orientalis*. The Deep SHAP model avoids heuristic selection of linearized components but enables effective linearization from the SHAP values calculated for each component [37]. Therefore, the contribution of different factors in each sample to the model prediction can be achieved via Deep SHAP.

2.4.6. Overall Process of Predicting the Initial Flowering Period in DL

Based on the phenological observation city network and the meteorological observation data of China, we built a comprehensive dataset of the initial flowering and meteorology, importing the dataset into the RNN, LSTM, and GRU models as input vectors and using MSE as the loss function. When the loss function converges, the model is considered mature. MAE, MAPE and R^2 are selected as evaluation indicators to express the prediction effect. In order to compare the difference between the initial flowering period prediction model based on deep learning algorithms and the traditional flowering prediction models, we selected the multiple linear regression model based on accumulated air temperature as the representative of the traditional initial flowering period prediction model, and compared the prediction effect of this model with DL The interpretability model based on SHAP is adopted to further analyze the interpretability and stability of the model. This process can be obtained in Figure 3.

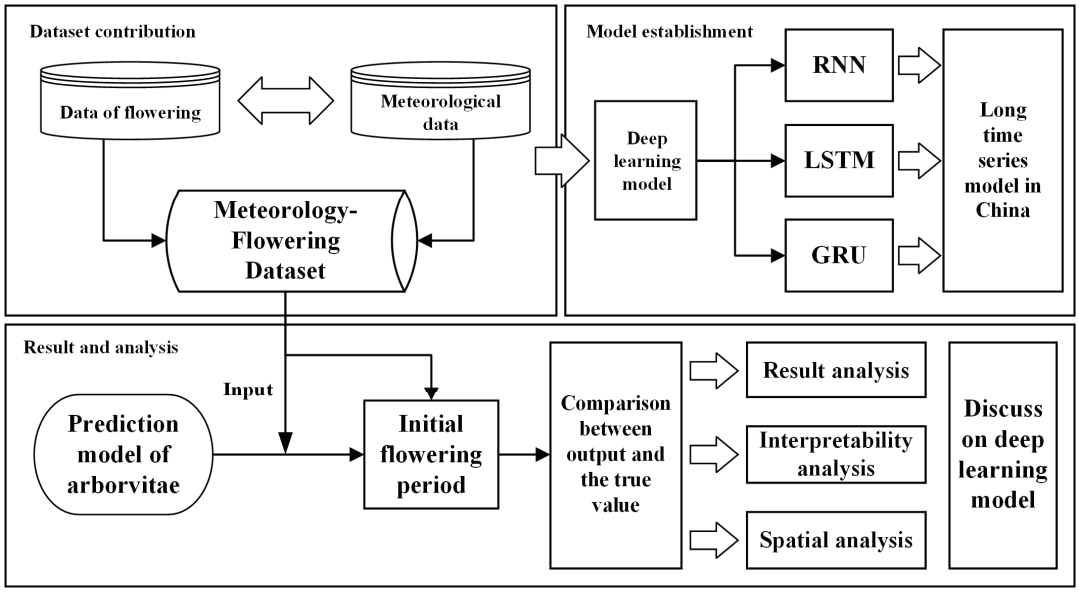


Figure 3. Flow figure of establishing DL models.

3. Results

3.1. Basic Characteristics of P. orientalis during Initial Flowering

As shown in Table 2, the flowering period of *P. orientalis* has obvious regional characteristics: with an increase in latitude, the average initial flowering period is gradually postponed, and the ordinal number of cities in northeast China is nearly 80 d (as a unit representing days), higher than that of coastal cities in south China such as Foshan and Shanghai etc., which is related to the generally high light, temperature and precipitation resources in south China. The dispersion degree of the initial flowering period of different stations can be obtained from the standard deviation. The standard deviation of 23 stations is concentrated at approximately 10 d. The maximum of Kunming station is 23.93 d, and the minimum of Harbin station is 1.50 d. Among all the data, the ordinal number of the earliest flowering period is 5 d observed at Kunming station, and that of the latest flowering period is 136 d at Minqin station. There are obvious interannual fluctuations and spatial differences in the observation data of each station, and the degree of dispersion is large with the range is 131 and normalized standard deviation is more than 0.2. Therefore, it is necessary to establish an accurate prediction model to effectively predict the initial flowering of *P. orientalis* nationwide.

Table 2. Table of ordinal number information of *P. orientalis*' initial flowering period.

Station	Average Value (d)	Minimum Value (d)	Maximum Value (d)	Range (d)	Standard Deviation (d)	Skewness	Kurtosis
Baoding	95.00	76	111	35	10.29	−0.16	−0.28
Beijing	86.97	65	108	43	10.03	0.12	−0.59
Changde	59.88	38	78	40	10.06	−0.27	0.026
Guiyang	57.05	33	86	53	13.89	−0.11	−0.44
Hohhot	108.00	101	121	20	6.31	0.97	0.19
Shanghai	63.38	50	76	26	7.61	−0.04	−0.07
Foshan	48.78	32	65	33	12.27	−0.08	−1.88
Nanjing	44.90	31	55	24	7.30	−0.36	−0.69
Nanchang	55.78	25	76	51	13.43	−0.58	0.34
Hefei	63.93	41	78	37	11.11	−0.67	−0.70
Harbin	130.50	129	132	3	1.50	0.01	0.01
Kunming	40.08	5	98	93	23.96	0.93	0.95
Guilin	43.35	22	74	52	16.51	0.80	−0.33
Wuhan	88.05	52	112	60	18.94	−0.41	−1.17
Minqin	104.93	92	136	44	10.76	1.61	4.07
Shenyang	111.20	104	122	18	7.33	0.69	−2.49
Tai'an	76.25	70	86	16	6.01	1.29	1.78
Xi'an	65.86	46	81	35	8.20	−0.43	0.51
Chongqing	54.62	24	76	52	14.99	−0.39	−1.05
Yinchuan	110.21	84	123	39	12.90	−0.83	−0.67
Changchun	111.96	93	129	36	7.91	0.12	0.89
Changsha	54.00	45	63	18	9.00	0.01	0.01
Yancheng	68.09	44	80	36	8.09	−1.09	1.69

3.2. Model Training Effect

Normalized meteorological data and initial flowering data were imported into the DL models as inputs. It can be seen from Figure 4 that with the increase in the training epoch which represents the number of cycles in the training process, the loss functions of the three deep learning models converge, which shows that the prediction error of the model reaches a small value. Therefore, the training is stopped, and the flowering prediction test is conducted.

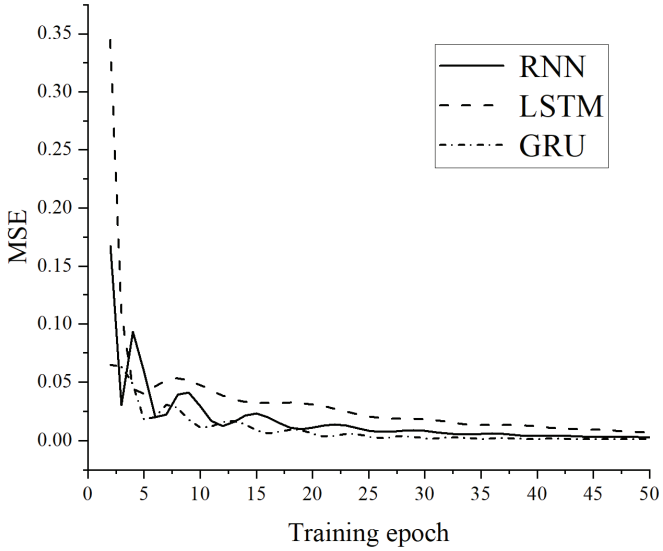


Figure 4. The training process of deep learning models.

In the test dataset, the MAE of the three models is small, and the MAPE of LSTM and the GRU is less than 1%. The R^2 values are greater than 0.99, indicating that there is a significant linear relationship between the true value and the predicted value, which can be obtained in Table 3.

Table 3. Table of prediction effect of DL.

Models and Indicators	RNN	LSTM	GRU
MAE	1.50×10^{-2}	5.18×10^{-4}	2.16×10^{-4}
MAPE	4.56	0.16	0.05
R^2	0.99	0.99	0.99

Typical stations, Yancheng station, Guiyang station and Beijing station, are selected from 23 stations, and prediction analysis is performed. Figure 5 shows that the three deep learning models can better simulate the actual local data of the initial flowering period. The fluctuation trend of the LSTM and GRU models is different from that of the actual extreme years, which is mainly characterized by hysteresis, and the simulated fluctuation change is always smaller than the actual value in the year with obvious changes.

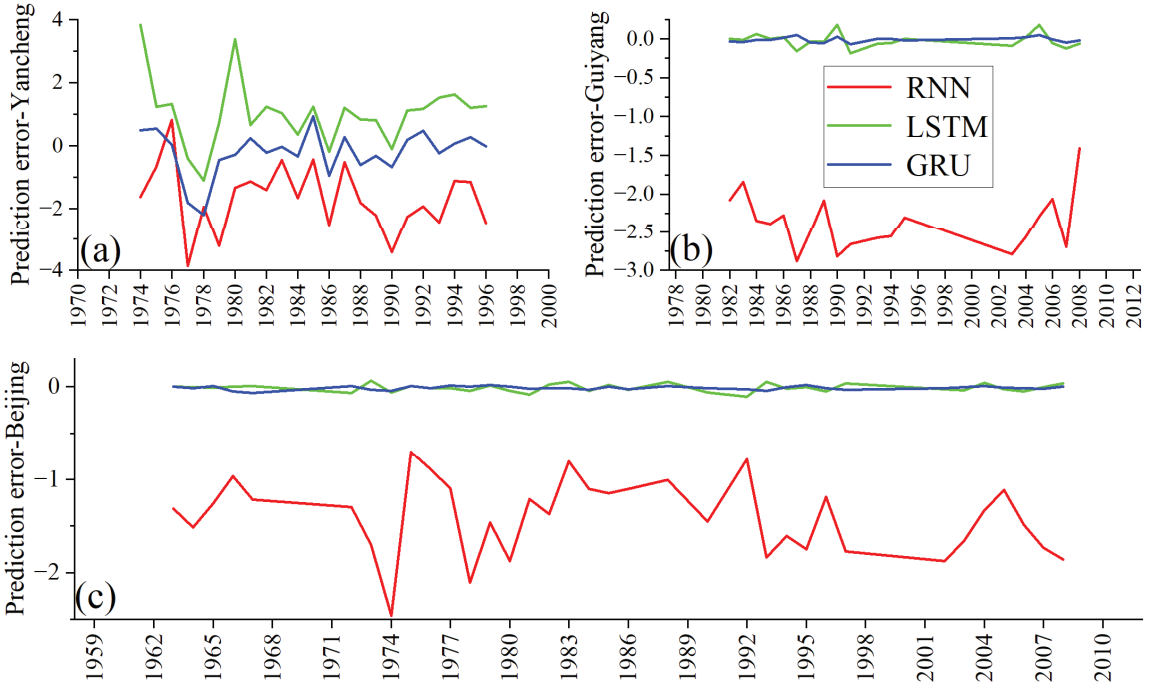


Figure 5. Interannual variation figure of prediction error of (**a**) Yancheng city, (**b**) Guiyang city, and (**c**) Beijing city.

3.3. Interpretability of DL Models

In Deep SHAP, a single sample will output SHAP values of different factors. We used the data of all samples including the training dataset and test dataset, which is a matrix of 89 meteorological factors and ordinal number of initial flowering period. Therefore, a matrix of SHAP values of the same size can be obtained. We explore the importance of different meteorological factors to model prediction by taking the average SHAP value of the whole sample as the factor contribution rate and analyze the stability of different factors

by using the change in the SHAP value of different samples in various meteorological factors as the stability index.

Figure 6a–c shows the analysis thermodynamic diagrams of RNN, LSTM and GRU. Its *x*-axis represents 89 meteorological factors, which are shown by x1-x89, and the order of meteorological factors is from the effective accumulated temperature of 0 °C to early average pressure for ten days in April. The *y*-axis represents 357 samples, which are shown by A1-A357. According to SHAP, the value greater than 0 in the thermodynamic diagrams promotes the prediction effect of the model, while the value less than 0 reduces the prediction effect, and we used color palette to indicate whether the value is greater than 0.

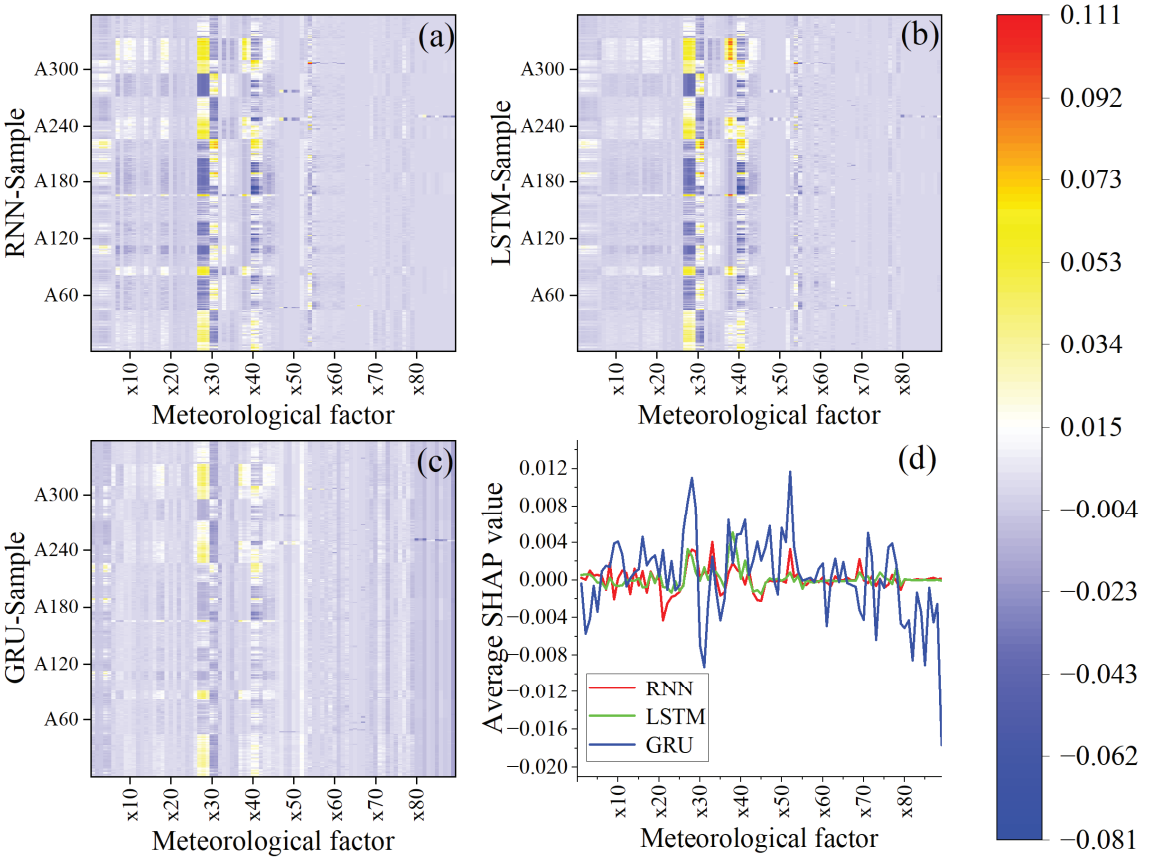


Figure 6. SHAP analysis figure of deep learning models (**a**). RNN contribution analysis thermodynamic diagram; (**b**). LSTM contribution analysis thermodynamic diagram; (**c**). GRU contribution analysis thermodynamic diagram; (**d**). Analysis figure of average contribution of each factor.

Therefore, Figure 6a–c can reflect the contribution rate stability of each meteorological factor though the change of SHAP value of each factor in different samples. In the thermal image, the meteorological factors with obvious fluctuations in the contribution rates of different models are similar and mainly concentrated in various factors related to the minimum temperature. The SHAP value of temperature factors is stable near the positive value, while the SHAP value of pressure factors is stable at the negative value.

According to Figure 6d, among different deep learning models, the average factor contribution rate is different with the range being 0.011 and normalized standard deviation

being 0.2, but in general, temperature factors are more important to the model with 58.6% of temperature factors values being higher than 0. Other factors are less important to the model, and some factors have negative SHAP values, which means they have a negative role in improving model prediction. GRU is more sensitive to input factors, so the absolute value of the contribution rate of GRU factors is higher than that of the other two models, while LSTM is the least sensitive to input factors, among which the absolute value of contribution rate of RNN, LSTM and GRU are 8.22×10^{-4}, 5.87×10^{-4}, 3.25×10^{-3}.

3.4. Comparison between DL and the Traditional Prediction Model

Non-DL flowering prediction methods usually use a few meteorological factors to establish regression models to forecast the initial flowering period, such as a multiple linear regression model, which is a linear regression model with multiple independent variables [38,39]. However, the simple linear models have difficulty accurately predicting flowering period. Chen and others have established a linear mode of multiple linear regression and nonlinear models of polynomial regression between the cherry flowering period and climate factors, and determined that they have a good simulation effect for the nonlinear modes with an average error of prediction less than 1.5 d [23,40]. In the neural network structure of deep learning, there are linear operations such as the convolution layer and nonlinear operations such as the activation function. To test the prediction effect of the deep learning model, we also select the multivariate linear regression model based on the accumulated temperature as the contrast for comparison.

According to the research of most scholars [9,13,19,20,22,23,33], we use the effective accumulated temperature (whose lower limit temperatures are 0 °C, 3 °C, 5 °C and 10 °C), active accumulated temperature and total accumulated temperature as variable factors to establish a multiple linear regression model:

$$y = 166.33x_1 - 261.86x_2 + 59.07x_3 + 38.79x_4 - 0.85x_5 - 1.57x_6 + 0.77 \quad (11)$$

where x_1, x_2, x_3, and x_4 are the effective accumulated temperatures whose lower limit temperature are 0 °C, 3 °C, 5 °C and 10 °C, x_5 is the active accumulated temperature, and x_6 is the total accumulated temperature. The coefficient of each independent variable is its linear relationship with y.

According to the deep learning models and multiple linear regression model, the prediction accuracy of each model is evaluated via MAE, MAPE and R^2, and the results can be obtained from Table 4.

Table 4. Table of comparison between deep learning model and multiple linear regression.

Model / Indicator	Deep Learning Model				Multiple Linear Regression Model
	RNN	LSTM	GRU	Mean	
MAE	1.50×10^{-2}	5.18×10^{-4}	2.16×10^{-4}	5.12×10^{-3}	0.06
MAPE	4.56	0.16	0.053	1.59	15.45
R^2	0.99	0.99	0.99	0.99	0.84

By comparison, the accuracy of the deep learning model was significantly higher than that of the multiple linear regression model with a confidence level of 0.05. And through the multicollinearity analysis, it can be found that in the multiple linear regression model, there is a collinearity problem between the 0 °C effective accumulated temperature and the 10 °C effective accumulated temperature.

3.5. Spatial Distribution and Interpolation of Prediction for DL

Due to the relationship between meteorological elements and space (longitude and latitude), we utilized all phenological data and verified the deep learning model at different stations to show the impact of spatial factors on flowering prediction. We import all the samples into the trained models and calculate the difference with the true value to get

the prediction error. When the error is more than 0 d, it means that the prediction results are ahead of the initial flowering period. The smaller the absolute error, the better the prediction effect. According to the Figure 7, the prediction average error of the RNN model lag behind the true value, mainly focusing on (−2d, −1d) and (−3d, −2d), while the prediction error of LSTM and GRU are mainly focused on (−1d, 0d), but the error of LSTM results exceeds 3d. By comparison, the prediction results of the GRU model are more accurate and stable.

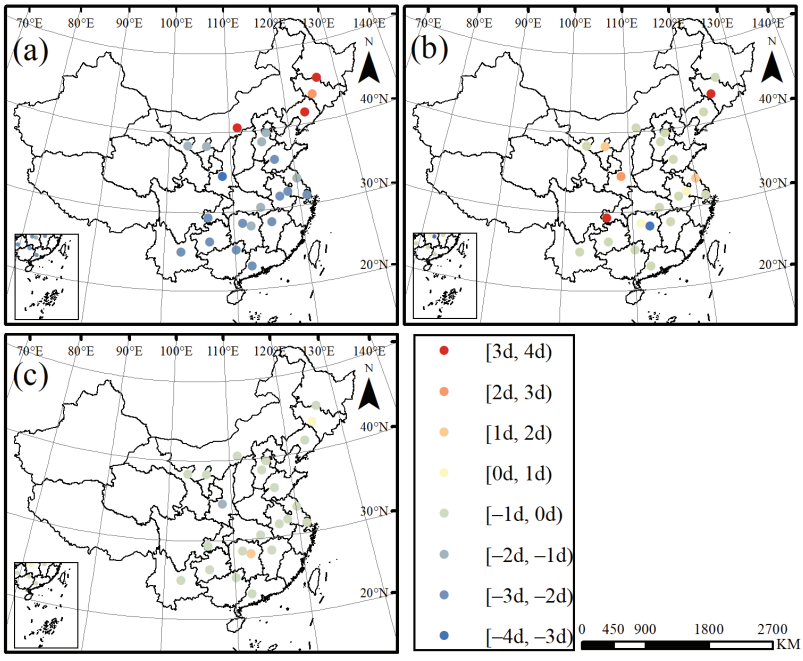


Figure 7. Spatial distribution map of (**a**) RNN, (**b**) LSTM, and (**c**) GRU prediction error.

Since most phenological observation stations in the dataset are concentrated in major urban areas of China and observation data in Northwest and Southwest China are missing, inverse distance weighting (IDW) is used for the average spatial prediction results of deep learning models. According to Figure 8, the interpolation results of the three models show similar characteristics. The similar characteristics are that in terms of latitude, ordinal number of initial flowering period gradually increases from low latitudes 15° N to high latitudes 55° N and present an obvious hierarchical structure, which is the layered structure of early, middle and late initial flowering periods from south China to north. The late flowering area mainly consists of Inner Mongolia and the three eastern provinces of Heilongjiang, Jilin and Liaoning, the middle flowering area mainly consists of central China, and the early flowering area mainly consists of the Yangtze River Delta, including Jiangsu, Zhejiang and Shanghai. The early flowering area and late flowering area have obvious differences in the initial flowering period. A possible main reason is that the late flowering area has a higher latitude, a smaller solar altitude angle, and less radiation, so the accumulated temperature and other resources are insufficient.

The prediction ordinal number of initial flowering period in different regions is similar in the three DL models with an average leaner trend between prediction value and years being −0.01, which means that the initial flowering period of *P. orientalis* in China will advance by about 1.31d each year.

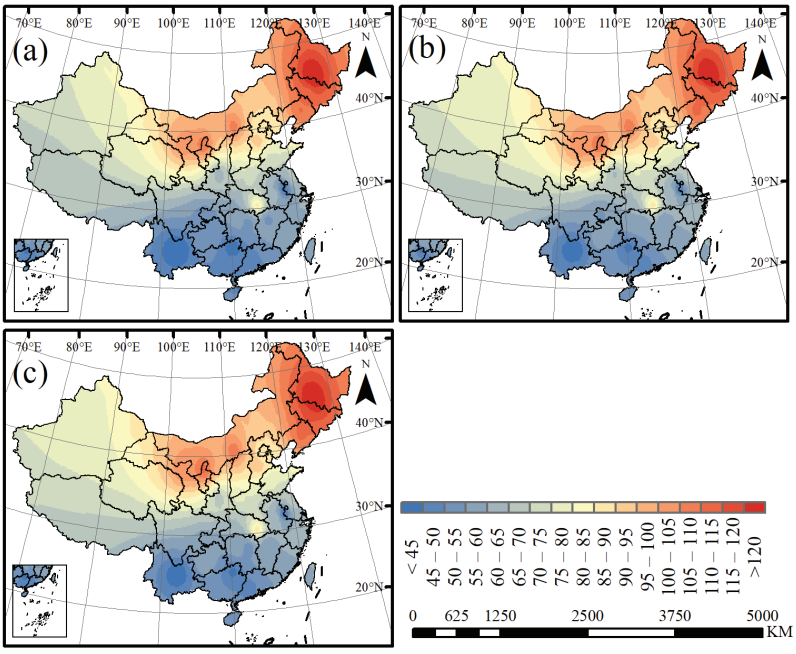


Figure 8. Spatial interpolation map of (**a**) RNN, (**b**) LSTM, and (**c**) GRU's average prediction.

4. Discussion

In this research, we employed deep learning to excavate the deep information relationship between the initial flowering period and phenology and realized a long-term flowering prediction model in China. The accuracy of the RNN, LSTM and GRU deep learning models is significantly higher than that of the traditional flowering prediction models based on multiple linear regression. Via interpretability analysis and spatial analysis, model stability problems such as factor sensitivity and error spatial distribution are explained.

From the viewpoint of some scholars, temperature is the main influencing factor affecting phenology [41–47], because it acts as a signal to regulate the dormancy process of plants [48]. Therefore, the mathematical regression models are built around accumulated air temperature, average air temperature and other factors related to temperature such as effective accumulated air temperature, etc. However, such models may have errors in the prediction effect over a short time and couldn't be applied to nationwide initial flowering period forecasts with the MAE, MAPE being higher than DL models and R^2 being lower.

Due to different meteorological conditions, the flowering period presents diversity in space [30]. In addition, because of the impact of climate change, the change of meteorological conditions in China is also different over time, showing the increase of annual temperature and precipitation [49,50]. This research achieves accurate nationwide prediction of a single species in China with the error of the initial flowering period reduced to less than 1 d, which provides more accurate data support for phenology research. With the development of industrialization, carbon emission might be the main factor affecting the opening process of flowers. Thus, this research provides a model basis for quantitative research on flowering changes in future scenarios.

However, there are some uncertainties in this research. The first uncertainty pertains to the data. We use the data of observation stations in major cities in China, with missing data from western China, which causes serious deviations between the prediction results and the actual value in this region. For DL, the SHAP of different models varies, and there is an obvious difference in the contribution of the three models to some meteorological factors, which makes it difficult to judge the correlation between such factors and the flowering period.

5. Conclusions

We predicted and analyzed the initial flowering period of *P. orientalis* in China through DL model, and the most important results of our study can be summed up as follows:

(1) The initial flowering in China mainly occurs from the beginning of February to the end of April, and it has spatial differences, which are later in northern China than in southern China.
(2) The DL model is suitable for nationwide flowering prediction in China, and the average error of DL is only within 2 d.
(3) Comparing the RNN, LSTM and the GRU, we find that the GRU is more suitable for the prediction model of initial flowering, with higher accuracy and more stable spatial predictions.
(4) The initial flowering period of *P. orientalis* in China presents obvious hierarchical characteristics, which are mainly manifested in the southern region where the flowering period is the earliest. With the increase in latitude, the initial flowering period gradually increases from south to north.

Although the variation in the contribution degree of output in the prediction of the initial flowering period can suggest different mechanisms of meteorological disasters affecting flowering process, our research is still insufficient.

Author Contributions: G.J.: visualization and writing. X.Z.: conceptualization and supervision. X.S., Y.S. and K.Y.: data collection and writing. W.S.: software and formal analysis. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China Project (41805049), National Students' Platform for Innovation and Entrepreneurship Training Program (202210300060Z) and NUIST Students' Platform for Innovation and Entrepreneurship Training Program (XJDC202210300493).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We thank the National Earth System Science Data Center, the Earth Big Data Science Data Center of the Chinese Academy of Sciences and the China Meteorological Science Data Sharing Network for data support. We acknowledge the High Performance Computing Center of Nanjing University of Information Science and Technology for their support of this work.

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

References

1. Root, T.L.; Price, J.T.; Hall, K.R.; Schneider, S.H.; Rosenzweig, C.; Pounds, J.A. Fingerprints of Global Warming on Wild Animals and Plants. *Nature* **2003**, *421*, 57–60. [CrossRef] [PubMed]
2. Walther, G.-R.; Post, E.; Convey, P.; Menzel, A.; Parmesan, C.; Beebee, T.J.C.; Fromentin, J.-M.; Hoegh-Guldberg, O.; Bairlein, F. Ecological Responses to Recent Climate Change. *Nature* **2002**, *416*, 389–395. [CrossRef] [PubMed]
3. Cleland, E.; Chuine, I.; Menzel, A.; Mooney, H.; Schwartz, M. Shifting Plant Phenology in Response to Global Change. *Trends Ecol. Evol.* **2007**, *22*, 357–365. [CrossRef] [PubMed]
4. Parmesan, C.; Yohe, G. A Globally Coherent Fingerprint of Climate Change Impacts across Natural Systems. *Nature* **2003**, *421*, 37–42. [CrossRef]
5. Bandoc, G.; Piticar, A.; Patriche, C.; Roșca, B.; Dragomir, E. Climate Warming-Induced Changes in Plant Phenology in the Most Important Agricultural Region of Romania. *Sustainability* **2022**, *14*, 2776. [CrossRef]
6. García-Mozo, H.; López-Orozco, R.; Oteros, J.; Galán, C. Factors Driving Autumn Quercus Flowering in a Thermo-Mediterranean Area. *Agronomy* **2022**, *12*, 2596. [CrossRef]
7. Linderholm, H.W. Growing Season Changes in the Last Century. *Agric. For. Meteorol.* **2006**, *137*, 1–14. [CrossRef]
8. Sparks, T. Local-Scale Adaptation to Climate Change: The Village Flower Festival. *Clim. Res.* **2014**, *60*, 87–89. [CrossRef]
9. Wang, L.; Ning, Z.; Wang, H.; Ge, Q. Impact of Climate Variability on Flowering Phenology and Its Implications for the Schedule of Blossom Festivals. *Sustainability* **2017**, *9*, 1127. [CrossRef]

10. Tao, Z.; Ge, Q.; Wang, H.; Dai, J. Phenological Basis of Determining Tourism Seasons for Ornamental Plants in Central and Eastern China. *J. Geogr. Sci.* **2015**, *25*, 1343–1356. [CrossRef]
11. Wolkovich, E.M.; Cook, B.I.; Allen, J.M.; Crimmins, T.M.; Betancourt, J.L.; Travers, S.E.; Pau, S.; Regetz, J.; Davies, T.J.; Kraft, N.J.B.; et al. Warming Experiments Underpredict Plant Phenological Responses to Climate Change. *Nature* **2012**, *485*, 494–497. [CrossRef] [PubMed]
12. Chmielewski, F.-M.; Rötzer, T. Response of Tree Phenology to Climate Change across Europe. *Agric. For. Meteorol.* **2001**, *108*, 101–112. [CrossRef]
13. Linkosalo, T.; Hakkinen, R.; Hanninen, H. Models of the Spring Phenology of Boreal and Temperate Trees: Is There Something Missing? *Tree Physiol.* **2006**, *26*, 1165–1172. [CrossRef] [PubMed]
14. Moussus, J.-P.; Julliard, R.; Jiguet, F. Featuring 10 Phenological Estimators Using Simulated Data: Featuring the Behaviour of Phenological Estimators. *Methods Ecol. Evol.* **2010**, *1*, 140–150. [CrossRef]
15. Verbesselt, J.; Hyndman, R.; Zeileis, A.; Culvenor, D. Phenological Change Detection While Accounting for Abrupt and Gradual Trends in Satellite Image Time Series. *Remote Sens. Environ.* **2010**, *114*, 2970–2980. [CrossRef]
16. Arlo Richardson, E.; Seeley, S.D.; Walker, D.R. A Model for Estimating the Completion of Rest for 'Redhaven' and 'Elberta' Peach Trees. *Hortscience* **1974**, *9*, 331–332. [CrossRef]
17. White, L.M. Relationship between Meteorological Measurements and Flowering of Index Species to Flowering of 53 Plant Species. *Agric. Meteorol.* **1979**, *20*, 189–204. [CrossRef]
18. Anderson, J.L.; Richardson, E.A.; Kesner, C.D. Validation of Chill Unit and Flower Bud Phenology Models for "Montmorency" Sour Cherry. *Acta Hortic.* **1986**, *184*, 71–78. [CrossRef]
19. Hakkinen, R.; Linkosalo, T.; Hari, P. Effects of Dormancy and Environmental Factors on Timing of Bud Burst in Betula Pendula. *Tree Physiol.* **1998**, *18*, 707–712. [CrossRef]
20. Demeloabreu, J. Modelling Olive Flowering Date Using Chilling for Dormancy Release and Thermal Time. *Agric. For. Meteorol.* **2004**, *125*, 117–127. [CrossRef]
21. Chauhan, Y.S.; Ryan, M.; Chandra, S.; Sadras, V.O. Accounting for Soil Moisture Improves Prediction of Flowering Time in Chickpea and Wheat. *Sci. Rep.* **2019**, *9*, 7510. [CrossRef] [PubMed]
22. Wu, D.; Huo, Z.; Wang, P.; Wang, J.; Jiang, H.; Bai, Q.; Yang, J. The Applicability of Mechanism Phenology Models to Simulating Apple Flowering Date in Shaanxi Province. *J. Appl. Meteor. Sci.* **2019**, *30*, 555–564. [CrossRef]
23. Tan, J.; Chen, Z.; Xiao, M. Characteristics and forecast of flowering duration of Cherry Blossoms in Wuhan University. *Acta Ecol. Sin.* **2021**, *41*, 38–47. [CrossRef]
24. Puchałka, R.; Klisz, M.; Koniakin, S.; Czortek, P.; Dylewski, Ł.; Paź-Dyderska, S.; Vítková, M.; Sádlo, J.; Rašomavičius, V.; Čarni, A.; et al. Citizen Science Helps Predictions of Climate Change Impact on Flowering Phenology: A Study on Anemone Nemorosa. *Agric. For. Meteorol.* **2022**, *325*, 109133. [CrossRef]
25. Wang, L.; Zhou, X.; Zhu, X.; Dong, Z.; Guo, W. Estimation of Biomass in Wheat Using Random Forest Regression Algorithm and Remote Sensing Data. *Crop J.* **2016**, *4*, 212–219. [CrossRef]
26. Jiang, P.; Shi, J.; Niu, P.X.; Yue, L.U. Effects on Activities of Defensive Enzymes and MDA Content in Leaves of *Platycladus Orientalis* under Naturally Decreasing Temperature. *J. Shihezi Univ. (Nat. Sci.)* **2009**, *127*, 487–493. [CrossRef]
27. Li, X.P.; He, Y.P.; Wu, X.J.; Ren, Q.F. Water Stress Experiments of *Platycladus Orientalis* and Pinns Tablaeformis Young Trees. *For. Res.* **2011**, *24*, 91–96. [CrossRef]
28. Wang, F.; Wu, D.; Haruhiko, Y.; Xing, S.; Zang, L. Digital Image Analysis of Different Crown Shape of *Platycladus Orientalis*. *Ecol. Inform.* **2016**, *34*, 146–152. [CrossRef]
29. An Editorial Committee of Flora of China. *Flora of China*; Science Press: Beijing, China; Missouri Botanical Garden Press: St. Louis, MO, USA, 1999; Volume 4.
30. Yearbook of the People's Republic of China Climate. Available online: http://www.gov.cn/guoqing/2005-09/13/content_2582628.htm (accessed on 3 December 2022).
31. Xu, J.; Wang, D.; Qiu, X.; Zeng, Y.; Zhu, X.; Li, M.; He, Y.; Shi, G. Dominant Factor of Dry-wet Change in China since 1960s. *Int. J. Climatol.* **2021**, *41*, 1039–1055. [CrossRef]
32. Mi, Q.; Gao, X.; Li, Y.; Li, X.; Tang, Y.; Ren, C. Application of Deep Learning Method to Drought Prediction. *J. Appl. Meteorol. Sci.* **2022**, *33*, 104–114.
33. Deo, R.C.; Şahin, M. Application of the Artificial Neural Network Model for Prediction of Monthly Standardized Precipitation and Evapotranspiration Index Using Hydrometeorological Parameters and Climate Indices in Eastern Australia. *Atmos. Res.* **2015**, *161–162*, 65–81. [CrossRef]
34. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
35. Gers, F.A. Learning to Forget: Continual Prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [CrossRef] [PubMed]
36. Amin, M.; Akram, M.N.; Ramzan, Q. Bayesian Estimation of Ridge Parameter under Different Loss Functions. *Commun. Stat. Theory Methods* **2022**, *51*, 4055–4071. [CrossRef]
37. Lundberg, S.; Lee, S.-I. A Unified Approach to Interpreting Model Predictions. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–10.
38. Inouye, D.W. Effects of Climate Change on Phenology, Frost Damage, and Floral Abundance of Montane Wildflowers. *Ecology* **2008**, *89*, 353–362. [CrossRef] [PubMed]

39. Park, I.W.; Mazer, S.J. Overlooked Climate Parameters Best Predict Flowering Onset: Assessing Phenological Models Using the Elastic Net. *Glob. Chang. Biol.* **2018**, *24*, 5972–5984. [CrossRef]
40. Chen, Z.; Xiao, M.; Chen, X. Change in Flowering Dates of Japanese Cherry Blossoms (*P. Yedoensis Mats*.) in Wuhan University Campus and Its Relationship with Variability of Winter Temperature. *Acta Ecol. Sin.* **2008**, *28*, 5209–5217.
41. Menzel, A.; Sparks, T.H.; Estrella, N.; Koch, E.; Aasa, A.; Ahas, R.; Alm-Kübler, K.; Bissolli, P.; Braslavská, O.; Briede, A.; et al. European Phenological Response to Climate Change Matches the Warming Pattern: European Phenological Response to Climate Change. *Glob. Chang. Biol.* **2006**, *12*, 1969–1976. [CrossRef]
42. Abu-Asab, M.S.; Peterson, P.M.; Shetler, S.G.; Orli, S.S. Earlier Plant Flowering in Spring as a Response to Global Warming in the Washington, DC, Area. *Biodivers. Conserv.* **2001**, *10*, 597–612. [CrossRef]
43. Zhou, L. Relation between Interannual Variations in Satellite Measures of Northern Forest Greenness and Climate between 1982 and 1999. *J. Geophys. Res.* **2003**, *108*, 4004. [CrossRef]
44. Fitter, A.H.; Fitter, R.S.R.; Harris, I.T.B.; Williamson, M.H. Relationships Between First Flowering Date and Temperature in the Flora of a Locality in Central England. *Funct. Ecol.* **1995**, *9*, 55. [CrossRef]
45. Krüger, E.; Woznicki, T.L.; Heide, O.M.; Kusnierek, K.; Rivero, R.; Masny, A.; Sowik, I.; Brauksiepe, B.; Eimert, K.; Mott, D.; et al. Flowering Phenology of Six Seasonal-Flowering Strawberry Cultivars in a Coordinated European Study. *Horticulturae* **2022**, *8*, 933. [CrossRef]
46. Bonelli, M.; Eustacchio, E.; Avesani, D.; Michelsen, V.; Falaschi, M.; Caccianiga, M.; Gobbi, M.; Casartelli, M. The Early Season Community of Flower-Visiting Arthropods in a High-Altitude Alpine Environment. *Insects* **2022**, *13*, 393. [CrossRef]
47. Monder, M.J. Trends in the Phenology of Climber Roses under Changing Climate Conditions in the Mazovia Lowland in Central Europe. *Appl. Sci.* **2022**, *12*, 4259. [CrossRef]
48. Tooke, F.; Battey, N.H. Temperate Flowering Phenology. *J. Exp. Bot.* **2010**, *61*, 2853–2862. [CrossRef]
49. Shi, Y.; Shen, Y.; Kang, E.; Li, D.; Ding, Y.; Zhang, G.; Hu, R. Recent and Future Climate Change in Northwest China. *Clim. Chang.* **2007**, *80*, 379–393. [CrossRef]
50. Shi, P.; Sun, S.; Wang, M.; Li, N.; Wang, J.; Jin, Y.; Gu, X.; Yin, W. Climate Change Regionalization in China (1961–2010). *Sci. China Earth Sci.* **2014**, *57*, 2676–2689. [CrossRef]

agriculture

MDPI

Article

Research on Laying Hens Feeding Behavior Detection and Model Visualization Based on Convolutional Neural Network

Hongyun Hao [1], Peng Fang [2], Wei Jiang [1], Xianqiu Sun [3], Liangju Wang [1] and Hongying Wang [1,*]

[1] College of Engineering, China Agriculture University, Beijing 100083, China
[2] College of Engineering, Jiangxi Agriculture University, Nanchang 330045, China
[3] Shandong Minhe Animal Husbandry Co., Ltd., Yantai 265600, China
* Correspondence: hongyingw@cau.edu.cn; Tel.: +86-13-6810-17695

Abstract: The feeding behavior of laying hens is closely related to their health and welfare status. In large-scale breeding farms, monitoring the feeding behavior of hens can effectively improve production management. However, manual monitoring is not only time-consuming but also reduces the welfare level of breeding staff. In order to realize automatic tracking of the feeding behavior of laying hens in the stacked cage laying houses, a feeding behavior detection network was constructed based on the Faster R-CNN network, which was characterized by the fusion of a 101 layers-deep residual network (ResNet101) and Path Aggregation Network (PAN) for feature extraction, and Intersection over Union (IoU) loss function for bounding box regression. The ablation experiments showed that the improved Faster R-CNN model enhanced precision, recall and F1-score from 84.40%, 72.67% and 0.781 to 90.12%, 79.14%, 0.843, respectively, which could enable the accurate detection of feeding behavior of laying hens. To understand the internal mechanism of the feeding behavior detection model, the convolutional kernel features and the feature maps output by the convolutional layers at each stage of the network were then visualized in an attempt to decipher the mechanisms within the Convolutional Neural Network(CNN) and provide a theoretical basis for optimizing the laying hens' behavior recognition network.

Keywords: laying hens; feeding behavior; Faster R-CNN; model visualization

Citation: Hao, H.; Fang, P.; Jiang, W.; Sun, X.; Wang, L.; Wang, H. Research on Laying Hens Feeding Behavior Detection and Model Visualization Based on Convolutional Neural Network. *Agriculture* **2022**, *12*, 2141. https://doi.org/10.3390/agriculture12122141

Academic Editors: Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 10 November 2022
Accepted: 9 December 2022
Published: 13 December 2022

1. Introduction

In recent years, researchers have studied the health and welfare of animals by monitoring their individual behaviors [1,2]. A laying hen's behavioral activities can be divided into feeding, drinking, resting, fighting, etc. Feeding is one of the most important behaviors in the life of laying hens, and it accounts for more than 40% of total activity time [3]. In the large-scale poultry breeding farm, abnormal feeding behavior of laying hens could reflect a health and welfare problem in the long term. For example, the decline in feed frequency and feed intake of some hens may indicate the possibility of disease, while the large-scale deterioration of the feed frequency may indicate that timely feeding is needed. On the contrary, the simultaneous and unexpected occurrence of high feed intake and low egg production may also reflect a health problem of laying hens. Thus, monitoring the feeding behavior of laying hens is significant in the breeding farm.

Traditionally, image processing technology is used to identify or classify poultry behaviors. However, it has the disadvantages of poor model generality, robustness, and difficulty in feature extraction [4–7]. Deep learning technology can learn the characteristics of the data itself through a large number of samples and has the advantages of speed, accuracy, and robustness; it is widely used in image detection and segmentation of animals. Some researchers have utilized deep learning and machine vision methods to detect typical behaviors of livestock and poultry, such as feeding, climbing, drinking, and excretion [8–14]. Wang et al. [15] built a laying hens behavior detection model based on the YOLOv3 network, which could recognize the feeding, mating, standing, and fighting

behaviors of laying hens. To identify broilers' lameness, Nasiri et al. [16] used CNN to extract the key points of the broiler's body and Long Short-Term Memory (LSTM) to classify the lameness of broilers. Fang et al. [17] employed a similar method for pose estimation and behavior classification of broiler chickens, which could identify broiler behaviors such as eating, standing, walking, running, resting, and preening. Geffen et al. [18] detected and counted the laying hens in the battery cages with the Faster R-CNN network and achieved 89.6% accuracy at cage level. Fang et al. [19] constructed a laying hens behavior detection network based on the Faster R-CNN network and knowledge-distillation technology, which significantly improved model performance while reducing the model inference time.

Previous research has proved that CNN could realize the analysis and recognition of image content and effectively solve the problems related to animal behaviors. However, we lack an understanding of its internal implementation mechanism, and the outstanding recognition performance lacks explanation. Therefore, during the model development process, a model with better performance can only be obtained through continuous trial and error [20].

In this research, we developed a feeding behavior detection model for stacked cage hens based on an improved Faster R-CNN network [21]. To solve the problem of loss of low-level features in the network, a feature extraction network based on the path aggregation network was constructed, and the regression loss function was improved, which significantly improved the performance of the feeding behavior detection network. Following this, the convolutional kernel features and the feature maps output by the convolutional layers at each stage of the network were visualized in an attempt to interpret the mechanism within the convolutional neural network and provide a theoretical foundation for the continuous optimization of the hens' behavior detection network.

2. Materials and Methods

2.1. Experimental Setup

The experiment in this research was conducted in Deqingyuan Ecological Park, Yanqing, Beijing, China. Laying hens (Jinghong 1) were reared in a 4 layers-stacked cage breeding house. There was a total of 9200 cages; each cage was 45 cm wide, 60 cm deep, and 50 cm high. A nipple drinker was installed inside the cage, and a feed trough was seated outside, with a light source located directly above the passageway. Six laying hens were reared in a single cage, and usually, 2–4 laying hens were in the feeding position for feeding, and the rest were drinking or resting.

The image acquisition system in this experiment consisted of three digital cameras (XCG-CG240C, SONY, Shanghai, China) with a resolution of 1920 $\times$ 1200 pixels, three fixed focus lenses (Ricoh FLCC0614A 2M, RICHO, Philippines), and a mobile inspection platform. The cameras were mounted on the mobile inspection platform at an angle of 30 degrees downward horizontally and were controlled by a microcomputer (Dell OptiPlex 7080MFF, Dell Inc., Xiamen, China) to capture images of the laying hens. Figure 1 shows the image acquisition system and the housing condition of laying hens. The inspection platform traveled to the front of each cage to collect images of the hens. Images were collected without adding additional light to minimize stress on the hens.

2.2. Data Collection and Labeling

Images were collected from November to December 2021. We selected 100 cages of laying hens for image acquisition and finally selected 1000 images as the original dataset. The data collection followed the Experimental Animal Welfare and Animal Experiment Ethics Committee of China Agricultural University guidelines. As shown in Figure 2, due to the difference in light intensity between hen layers, images collected from the first and second layers were enhanced using the Retinex enhancement algorithm to improve the image readability. After that, the original image set was labeled with the free image label tool "Labelme", in which hens whose heads were near or in the feeding trough were labeled as "feeding" and the others were labeled as "resting". In the detection work, the CNN

does not have scale invariance and rotation invariance due to the fixed characters of the convolution itself. The adaptive ability of the CNN to target changes almost comes from the diversity of data itself. The more and more comprehensive the data, the higher the accuracy of the trained model [10]. Therefore, the dataset was expanded to 2000 images by 90° random rotation, adding Gaussian noise and randomly adjusting image contrast to improve the model's generalization ability. Finally, the dataset contained 4268 samples of hens labeled as "feeding" and 4836 samples of hens labeled as "resting", and was randomly divided into a training set, validation set, and test set (7:2:1).

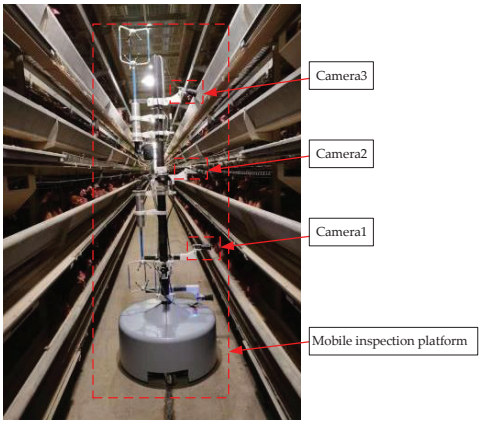


Figure 1. The image acquisition system and housing condition of laying hens.

(**a**) Original image (**b**) Enhanced image

Figure 2. Image sample of hens.

2.3. Faster R-CNN Network

The feeding behavior detection model was constructed based on the Faster R-CNN network in this research. As shown in Figure 3, the Faster R-CNN network can be divided into four parts: feature extraction network, Region Proposal Network (RPN), Region of interest (ROI) pooling network, bounding box regression and classification. The feature extraction network is used to extract the feature maps. The features maps are then shared with the region proposal network and the ROI pooling network, where the region proposal network extracts the candidate bounding boxes to the ROI pooling network, and through the ROI pooling layer, each ROI generates a fixed-size feature map; finally, regression and classification of the bounding boxes are performed.

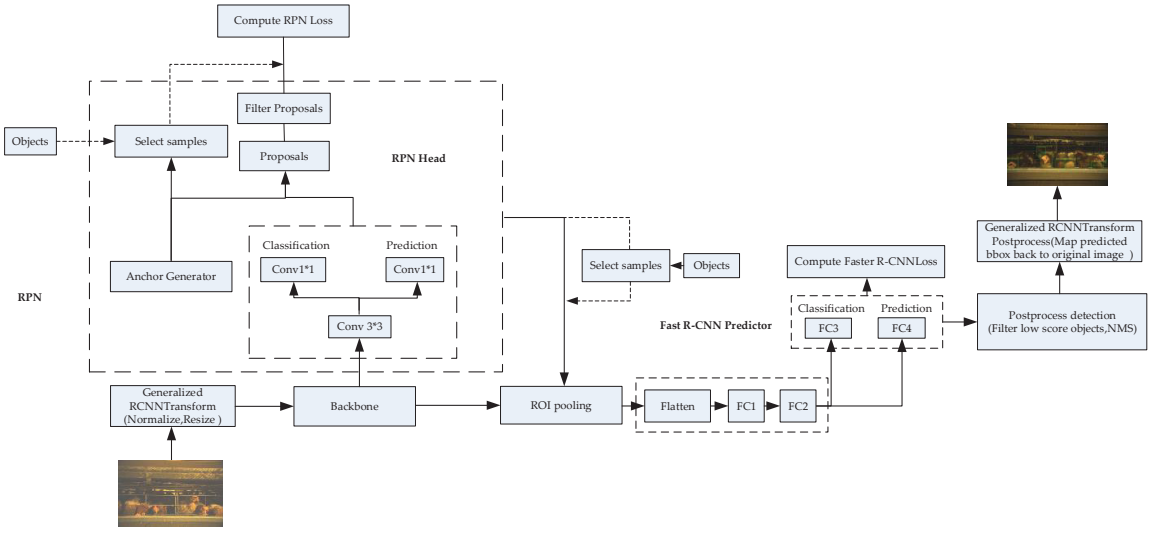


Figure 3. Structure of Faster R-CNN network. FC: fully connected layer, Conv: convolution.

2.4. Construction of Feature Extraction Network Based on Path Aggregation Network

In CNN, low-level layers focus on image details such as edge shape and object position, while deep layers will focus on strong semantic information. The object detection network needs to be concerned about the image's semantic information, position information, and pixel details. Therefore, it is necessary to fully use the features extracted by each level of the backbone network so that the input feature maps of the region proposal network get both semantically vital information and low-level localization information. The Faster R-CNN network achieves this through the Feature Pyramid Network (FPN), which significantly improves the detection ability of the Faster R-CNN network for small objects. However, in the "bottom-up" transmission architecture of FPN networks, the path from the shallow features to the top layer is too long. As shown by the red dotted path in Figure 4, the features extracted from the last convolutional layer of the second stage (stage 2) of the ResNet 101 network pass through hundreds of layers to the top layer (P5). The low-level feature information suffers severe losses through the transmission over long paths, which makes it difficult to preserve accurate target location information in the top-level feature map. Liu et al. [22] proposed a path aggregation network (PAN), for instance segmentation, which significantly improved the performance of an instance segmentation network by creating a bottom-up path augmentation, adaptive feature pooling structure and fully connected fusion method.

In this research, the bottom-up path augmentation of the PAN was introduced into the Faster R-CNN network. The four feature fusion layers were added after the FPN network by lateral connection, the architecture of which is shown in Figure 4b. With the addition of the bottom-up pathway augmentation, the low-level features extracted in the second stage of the ResNet 101 network were transmitted to feature map P2 by a lateral connection and subsequently passed through the feature map N2 to the top feature layer N5 (the path shown by the green dotted line in Figure 4). It took less than ten layers to transmit the low-level features to the top layer, which significantly shortened the information transmission path; the low-level feature information can be better retained in the top feature map, which is conducive to the accurate localization of the targets.

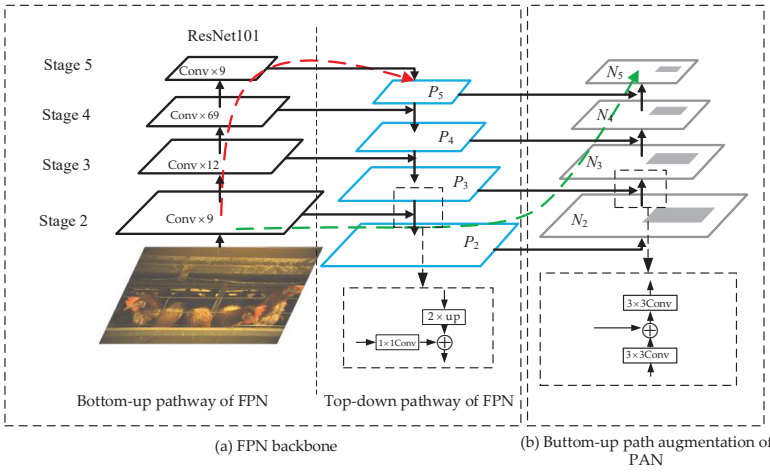


Figure 4. Structure of the improved feature extraction network. Conv: convolution, up: upsampling, ⊕: add.

2.5. Optimisation of the Loss Function

The regression loss and classification loss composed the loss of Faster R-CNN network. Among them, the Faster R-CNN network utilized the $smooth_{L1}$ loss as the regression loss, as shown in Equations (1) and (2).

$$L_{reg} = \lambda \frac{1}{N_{reg}} \sum_i p_i^* smooth_{L1}(t_i, t_i^*) \tag{1}$$

$$smooth_{L1}(t_i, t_i^*) = \begin{cases} 0.5(t_i - t^*)^2 (|t_i - t^*| < 1) \\ |t_i - t^*| - 0.5 (|t_i - t^*| \geq 1) \end{cases} \tag{2}$$

where L_{reg} is the regression loss of the Faster R-CNN, N_{reg} is the number of anchors, p_i^* is 1 if the anchor is positive and is 0 if the anchor is negative, t_i is a vector representing the 4 parameterized coordinates of the predicted bounding box, t_i^* is that of the ground-truth box associated with a positive anchor.

When calculating the regression loss of the network by the $smooth_{L1}$ function, the 4 points of the predicted bounding boxes are treated as independent of each other, and their respective loss values are calculated and then summed up to obtain the total regression loss. In fact, the four points are related to each other. IoU is usually used to evaluate the proximity between the predicted bounding boxes and the ground truth. When multiple predicted bounding boxes get the same $smooth_{L1}$ loss value, their IoU values may vary greatly. Thus, performing regression on the 4 points in isolation is inappropriate, and the predicted bounding boxes composed of the 4 points should be regarded as a whole for the regression. In this research, IoU loss [23], is used to replace the $smooth_{L1}$ loss in the Faster R-CNN network. The IoU loss function is defined as:

$$IoUloss = -\ln(IoU) \tag{3}$$

$$IoU = \frac{I}{U} \tag{4}$$

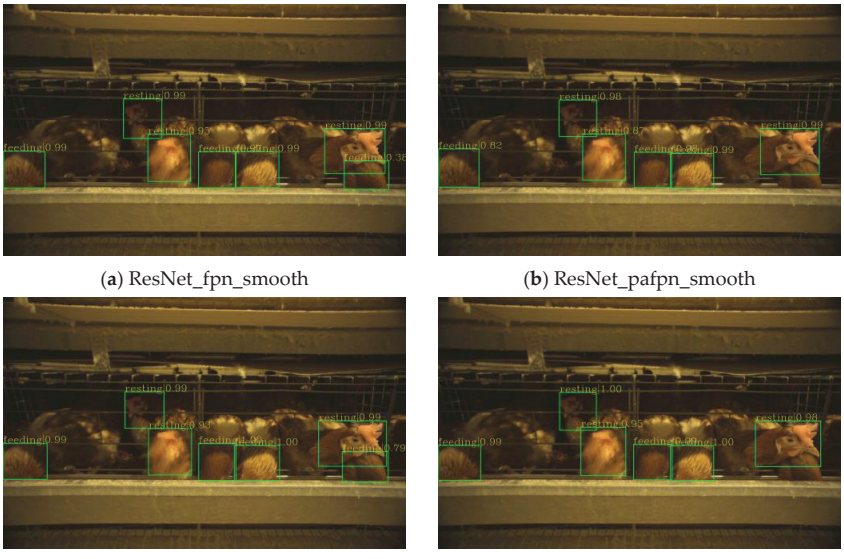
where IoU is the intersection and union ratio of the predicted bounding boxes and the ground truth; I is the area of the intersection region of the predicted bounding boxes and the ground truth; U is the union region of the predicted bounding boxes and the ground truth.

2.6. Model Training

In this research, the training work was performed on a Dell computer with an Intel(R) Core (TM) i7—9700K, an NVIDIA GeForce GTX2080 GPU (11 GB), and 16 GB of memory. The operating environment was Ubantu18.04, CUDA 10.2, cuDNN 8.0.1, and Python 3.7. The model was trained for 16,000 steps, with an initial learning rate of 0.001, a momentum of 0.9, Stochastic Gradient Descent (SGD) optimizer, and a weight decay of 0.0001. The learning rate increased to 0.002 after 8000 steps. In order to obtain the best model, weights were saved every 2000 steps.

3. Results

Different optimization methods of the feeding behavior detection network were tested in this experiment: ① Faster R-CNN network with Resnet101 and feature pyramid network as the feature extraction network, and $smooth_{L1}$ function as the regression loss function (ResNet_fpn_smooth). ② Faster R-CNN network with the Resnet101, path aggregation network, and feature pyramid network as the feature extraction network, and $smooth_{L1}$ function as the regression loss function (ResNet_pafpn_smooth). ③ Faster R-CNN network with the Resnet101 and feature pyramid network as the feature extraction network, and IoU loss as the regression loss function (ResNet_fpn_iou) ④ Faster R-CNN network with the Resnet101, path aggregation network and feature pyramid network as the feature extraction network, and IoU loss as the regression loss function (ResNet_pafpn_iou). The performance of the above four recognition models was tested with the test set, and the same image was input into each of the above four models to obtain the four sets of output results in Figure 5; all models could accurately identify the feeding and resting behaviors of hens.



(**a**) ResNet_fpn_smooth (**b**) ResNet_pafpn_smooth

(**c**) ResNet_fpn_iou (**d**) ResNet_pafpn_iou

Figure 5. Detection results of different models.

The Precision (P), recall (R), and average inference time (t) were used in this experiment to evaluate the feeding behavior detection model performance. As shown in Table 1, the detection precision of all models was above 80%. The accuracy, recall and F1-score of the ResNet_fpn_smooth were 84.4%, 72.67% and 0.781, respectively, while the corresponding values were 87.2%, 71.3% and 0.785 for the ResNet_pafpn_smooth. There was a noticeable improvement in the precision index after adding the path aggregation network to the Faster

R-CNN network and a slight decrease in the recall index. In addition, the inference time of both models was similar, which indicated that the path aggregation network improved the retention rate of low-level feature information and improved the detection precision of the object without increasing the model complexity too much. The precision, recall and F1-score of the ResNet_fpn_iou were 88.73%, 73.49% and 0.804, respectively, higher than that of ResNet_fpn_smooth and ResNet_pafpn_smooth, which means that the IoU loss function could calculate the error between the predicted and true values of the bounding box more accurately, to obtain more accurate prediction results. Finally, the ResNet_pafpn_iou got a precision of 90.12%, a recall of 79.14% and a F1-score of 0.843, which was the best.

Table 1. Performance comparison of different models.

Models	Precision/%	Recall/%	F1-Score	Average Inference Time/s
ResNet_fpn_smooth	84.40	72.67	0.781	0.143
ResNet_pafpn_smooth	87.20	71.31	0.785	0.145
ResNet_fpn_iou	88.73	73.49	0.804	0.143
ResNet_pafpn_iou	90.12	79.14	0.843	0.144

Figure 6 shows the training loss curve of the ResNet_fpn_smooth, ResNet_pafpn_smooth, ResNet_fpn_iou, and ResNet_pafpn_iou. The training loss decreased to a low value within a short time after the training started, then slowly reduced with the training process. The training loss became flat when the iteration was about 14,000 times and no longer declined. When the number of iterations reached 16,000, the training process ended, and the model converged. Based on the training loss curves in Figure 6, ResNet_pafpn_iou achieved the lowest converged loss, which indicated the effectiveness of the optimization process.

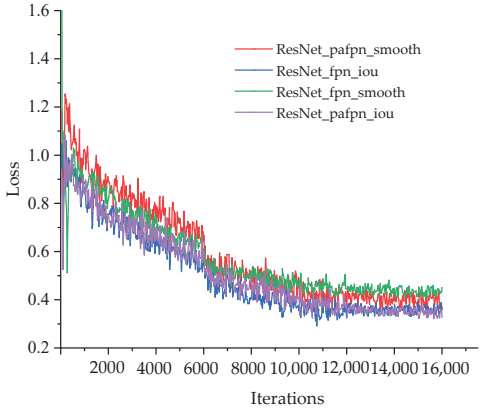


Figure 6. Training loss curves of different models.

4. Discussion

In the CNN, each layer of the network extracts different features through the convolution kernel, and the network will integrate the extracted features to realize the interpretation of the image content. Visualization of the CNN was first proposed by Zeiler et al. [20]. Subsequently, visualization techniques such as Class Activation Maps (CAM) [24] and Gradient-weighted Class Activation Maps (Grad-CAM) [25] were developed.

In this research, taking ResNet101 as an example, the features relating to the convolution kernel and the feature maps generated by the convolution layer of the feeding behavior detection network were visualized. The aim was to understand the internal mechanism of the convolutional neural network, providing a theoretical basis for the optimization of the

behavior recognition network of hens. The ResNet101 network consists of 101 convolution layers and can be divided into 5 stages. In the Faster R-CNN network, the feature maps extracted in the first stage of ResNet101 are not sent to the region proposal network. Therefore, we only visualized the feature maps of the last convolution layers in the second to fifth stages to analyze the differences between the extracted features of the low-level and the top ones.

4.1. Visual Analysis of the Feature Maps

The number of feature maps output in the second, third, fourth, and fifth stages of the ResNet101 network was 256, 512, 1024, and 2048, respectively, and all of the feature maps were single-channel images. In this section, all the single-channel feature maps of each stage were merged into a multi-channel image, and the 4 feature maps with the most significant activation features were output for visualization. Figure 7 shows the visualization results.

The training process of the CNN imitates the cognitive function of the human brain. The human visual system performs image recognition step by step, and people will first understand the color and brightness features in the image, then the simple geometric features such as points, lines, and edges, and after that, the slightly complex features (high-dimensional information) such as texture in the image, finally, forming the concept of the whole image. The CNN similarly processed the image. As shown in Figure 7, low-level layers in the second stage mainly extracted the image's low-level features, such as contour, edge, and color features. It focused more on the image's overall color and line information, not only the contour of the hen. With the deepening of the network, the third and fourth stages focused more on the texture of the image. In the third and fourth stages, the network gradually focused on the contour of hens, and some key features were extracted, including their head and cockscomb. As the network got more profound, the features extracted by the network began to be highly abstract, and the naked eye could no longer recognize the specific content of the extracted features. However, the convolutional neural network can extract essential information from it, and the area of concern of the network is basically focused on the hen's contour, ignoring the background. The subsequent fully connected convolution layers processed the features extracted from the high-level layer to complete the detection and classification of the hens.

4.2. Visual Analysis of the Convolution Kernels

The convolution kernel of the CNN is responsible for extracting features from the image. By visualizing the convolution kernel, we can more intuitively understand the features extracted by the convolution kernel of the image and clearly understand CNN's internal mechanism. The gradient lifting method is used to compute the input image when the convolution kernel of each layer in ResNet101 reaches the maximum activation state, and the input image is the feature extracted by the convolution kernel. This section visualized the first 36 convolution kernels of the last layer in stage 2–stage 5.

From the visualized results in Figure 8, in the second stage of the ResNet101, the convolution kernel extracted some low-level features such as color, line, and texture features. The combination of color and line features formed wavy and long strip textures. With the network getting deeper, the kernels in the fourth and fifth stages extracted more complicated texture features, spiral, circular, and various shape combinations of texture features. The convolution kernel became more and more complex, and the extracted features became more and more refined. A large number of complex and refined texture features gradually depicted the contour of the detection object (hens) as the network got deeper. In summary, the low-level layers of the network mainly extracted general features such as edges, lines, and some simple textures, while deeper layers could extract complex and semantically strong features (feather, eye), which were similar to the target characteristic to be detected.

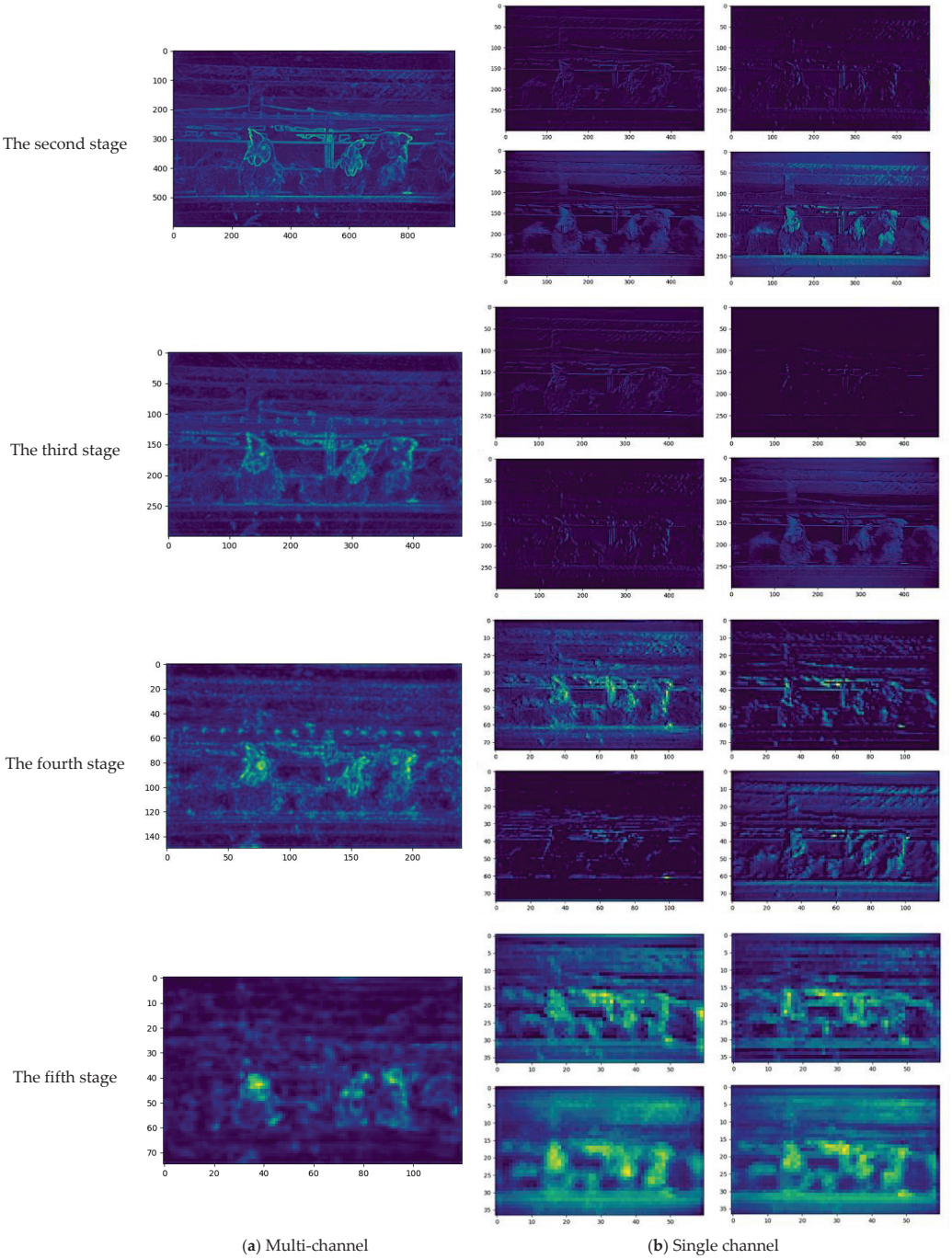


Figure 7. Feature maps from stage 2 to stage 5.

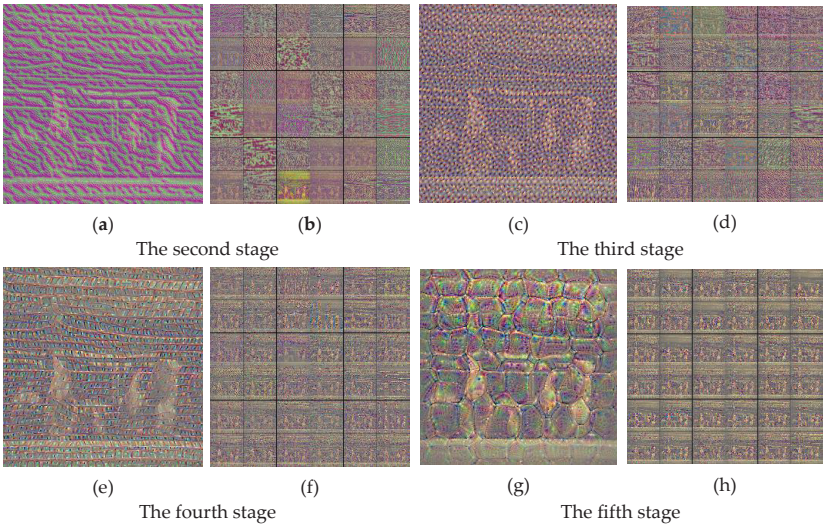


Figure 8. Convolution kernel visualization results. (**a**,**c**,**e**,**g**) are the visualization result of the first convolution kernels at each stage, respectively. (**b**,**d**,**f**,**h**) are the visualization results of the first 36 convolution kernels at each stage, respectively.

4.3. Limits and Future Work

It is worth noting that there were still some limitations to this study. In the detection of the feeding behavior of laying hens, only feeding behavior and resting behavior were taken into consideration; other behaviors, such as fighting, drinking, and egg laying, were not considered in this research. The small cage size and the lighting conditions of the stacked cage breeding house caused this. The drinking and laying behaviors of the hens always occurred inside the cage, while the feeding and resting laying hens would stay close to the front door, blocking the camera's view. Additionally, the low illumination of the house would result in almost no light inside the cage, which means that the camera cannot collect valid images for the detection work. Fighting behavior is often observed during feeding and can be obscured by the trough, making sample collection more complex. In future work, we will attempt to use an infrared camera to capture images and select a better angle.

Furthermore, the Faster R-CNN model was a two-stage object detection network, which was slower in detection speed than other networks studied [26,27]. Thus, a one-stage object detection network such as SSD [28], and YOLOv4 [29] should be considered to further improve the feeding behavior detection model. Lastly, the feeding behavior detection model has been developed for the stacked cage laying hens, but is not suitable for laying hens with other feeding methods. Therefore, the model can be further improved through the collection of more data from laying hens with different feeding patterns.

5. Conclusions

In this work, an improved Faster-RCNN model was constructed to recognize the feeding behavior of stacked caged hens based on a path aggregation network and IoU loss function. The precision, recall and F1-score of the model were improved from 84.40%, 72.67%, 0.781 to 90.12%, 79.14% and 0.843, respectively, and the average detection time was almost unchanged. After that, an ablation experiment was conducted to demonstrate the effectiveness of the improvement and visualize the output feature maps of the convolution layer and the convolution kernel features of the feeding behavior detection network, respectively. Based on the visualization results, the convolutional neural network's internal mechanism was analyzed to explain the CNN 's performance and provide a theoretical basis for further optimization of the detection model. In general, the developed model and

visual analysis method in this research could provide technical support for the subsequent monitoring of the health status and welfare status of laying hens and could also provide a reference for the optimization of other animal detection models. In future work, we will consider using a one-stage object detection network to optimize the feeding behavior detection model further and detect more behaviors, such as drinking and egg laying, to provide further technical support for poultry farm management.

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

Funding: This work was funded by the Ministry of Science and Technology, China under grant number: 2017YFE0122200.

Institutional Review Board Statement: The experiment was conducted following the guidelines of Experimental Animal Welfare and Animal Experiment Ethics Committee of China Agricultural University (Approved number: AW12112202-5-1).

Data Availability Statement: Not applicable.

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

References

1. Aydin, A.; Bahr, C.; Berckmans, D. A Real-Time Monitoring Tool to Automatically Measure the Feed Intakes of Multiple Broiler Chickens by Sound Analysis. *Comput. Electron. Agric.* **2015**, *114*, 1–6. [CrossRef]
2. Yang, X.; Zhao, Y.; Street, G.M.; Huang, Y.; Filip To, S.D.; Purswell, J.L. Classification of Broiler Behaviours Using Triaxial Accelerometer and Machine Learning. *Animal* **2021**, *15*, 100269. [CrossRef] [PubMed]
3. Hansen, I.; Braastad, B.O. Effect of Rearing Density on Pecking Behaviour and Plumage Condition of Laying Hens in Two Types of Aviary. *Appl. Anim. Behav. Sci.* **1994**, *40*, 263–272. [CrossRef]
4. Pereira, D.F.; Lopes, F.A.A.; Filho, L.R.A.G.; Salgado, D.D.; Neto, M.M. Cluster Index for Estimating Thermal Poultry Stress (Gallus Gallus Domesticus). *Comput. Electron. Agric.* **2020**, *177*, 105704. [CrossRef]
5. Neves, D.P.; Mehdizadeh, S.A.; Tscharke, M.; de Alencar Nääs, I.; Banhazi, T.M. Detection of Flock Movement and Behaviour of Broiler Chickens at Different Feeders Using Image Analysis. *Inf. Process. Agric.* **2015**, *2*, 177–182. [CrossRef]
6. de Alencar Nääs, I.; da Silva Lima, N.D.; Gonçalves, R.F.; Antonio de Lima, L.; Ungaro, H.; Minoro Abe, J. Lameness Prediction in Broiler Chicken Using a Machine Learning Technique. *Inf. Process. Agric.* **2021**, *8*, 409–418. [CrossRef]
7. Del Valle, J.E.; Pereira, D.F.; Mollo Neto, M.; Gabriel Filho, L.R.A.; Salgado, D.D. Unrest Index for Estimating Thermal Comfort of Poultry Birds (Gallus Gallus Domesticus) Using Computer Vision Techniques. *Biosyst. Eng.* **2021**, *206*, 123–134. [CrossRef]
8. Jia, N.; Kootstra, G.; Koerkamp, P.G.; Shi, Z.; Du, S. Segmentation of Body Parts of Cows in RGB-Depth Images Based on Template Matching. *Comput. Electron. Agric.* **2021**, *180*, 105897. [CrossRef]
9. Qiao, Y.; Truman, M.; Sukkarieh, S. Cattle Segmentation and Contour Extraction Based on Mask R-CNN for Precision Livestock Farming. *Comput. Electron. Agric.* **2019**, *165*, 104958. [CrossRef]
10. Lamping, C.; Derks, M.; Groot Koerkamp, P.; Kootstra, G. ChickenNet—An End-to-End Approach for Plumage Condition Assessment of Laying Hens in Commercial Farms Using Computer Vision. *Comput. Electron. Agric.* **2022**, *194*, 106695. [CrossRef]
11. Xiao, D.; Lin, S.; Liu, Y.; Yang, Q.; Wu, H. Group-Housed Pigs and Their Body Parts Detection with Cascade Faster R-CNN. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 203–209. [CrossRef]
12. da Silva Santos, A.; de Medeiros, V.W.C.; Gonçalves, G.E. Monitoring and Classification of Cattle Behavior: A Survey. *Smart Agric. Technol.* **2023**, *3*, 100091. [CrossRef]
13. Liu, L.; Zhou, J.; Zhang, B.; Dai, S.; Shen, M. Visual Detection on Posture Transformation Characteristics of Sows in Late Gestation Based on Libra R-CNN. *Biosyst. Eng.* **2022**, *223*, 219–231. [CrossRef]
14. Cheng, M.; Yuan, H.; Wang, Q.; Cai, Z.; Liu, Y.; Zhang, Y. Application of Deep Learning in Sheep Behaviors Recognition and Influence Analysis of Training Data Characteristics on the Recognition Effect. *Comput. Electron. Agric.* **2022**, *198*, 107010. [CrossRef]
15. Wang, J.; Wang, N.; Li, L.; Ren, Z. Real-Time Behavior Detection and Judgment of Egg Breeders Based on YOLO V3. *Neural Comput. Appl.* **2020**, *32*, 5471–5481. [CrossRef]
16. Nasiri, A.; Yoder, J.; Zhao, Y.; Hawkins, S.; Prado, M.; Gan, H. Pose Estimation-Based Lameness Recognition in Broiler Using CNN-LSTM Network. *Comput. Electron. Agric.* **2022**, *197*, 106931. [CrossRef]
17. Fang, C.; Zhang, T.; Zheng, H.; Huang, J.; Cuan, K. Pose Estimation and Behavior Classification of Broiler Chickens Based on Deep Neural Networks. *Comput. Electron. Agric.* **2021**, *180*, 105863. [CrossRef]

18. Geffen, O.; Yitzhaky, Y.; Barchilon, N.; Druyan, S.; Halachmi, I. A Machine Vision System to Detect and Count Laying Hens in Battery Cages. *Animal* **2020**, *14*, 2628–2634. [CrossRef]
19. Fang, P.; Hao, H.; Wang, H. Behavior Recognition Model of Stacked-cage Layers Based on Knowledge Distillation. *Trans. Chin. Soc. Agric. Mach.* **2021**, *52*, 300–306. [CrossRef]
20. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision(ECCV)*; Springer: Cham, Switzerland, 2014; p. 8689. [CrossRef]
21. 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]
22. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768. [CrossRef]
23. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. UnitBox: An Advanced Object Detection Network. In Proceedings of the 24th ACM international conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 516–520. [CrossRef]
24. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]
25. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning Deep Features for Discriminative Localization. *arXiv* **2015**, arXiv:1512.04150. [CrossRef]
26. Jiang, K.; Xie, T.; Yan, R.; Wen, X.; Li, D.; Jiang, H.; Jiang, N.; Feng, L.; Duan, X.; Wang, J. An Attention Mechanism-Improved YOLOv7 Object Detection Algorithm for Hemp Duck Count Estimation. *Agriculture* **2022**, *12*, 1659. [CrossRef]
27. Yang, J.; Zhang, T.; Fang, C.; Zheng, H. A Defencing Algorithm Based on Deep Learning Improves the Detection Accuracy of Caged Chickens. *Comput. Electron. Agric.* **2023**, *204*, 107501. [CrossRef]
28. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; Volume 9905, pp. 21–37. [CrossRef]
29. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLO v4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.

agriculture

MDPI

Article

Frequency-Enhanced Channel-Spatial Attention Module for Grain Pests Classification

Junwei Yu [1,*], Yi Shen [2], Nan Liu [3] and Quan Pan [2,4]

[1] School of Artificial Intelligence and Big Data, Henan University of Technology, Zhengzhou 450001, China
[2] College of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China
[3] Basis Department, PLA Information Engineering University, Zhengzhou 450001, China
[4] School of Automation, Northwestern Polytechnical University, Xi'an 710129, China
* Correspondence: yujunwei@126.com

Abstract: For grain storage and protection, grain pest species recognition and population density estimation are of great significance. With the rapid development of deep learning technology, many studies have shown that convolutional neural networks (CNN)-based methods perform extremely well in image classification. However, such studies on grain pest classification are still limited in the following two aspects. Firstly, there is no high-quality dataset of primary insect pests specified by standard ISO 6322-3 and the Chinese Technical Criterion for Grain and Oil-seeds Storage (GB/T 29890). The images of realistic storage scenes bring great challenges to the identification of grain pests as the images have attributes of small objects, varying pest shapes and cluttered backgrounds. Secondly, existing studies mostly use channel or spatial attention mechanisms, and as a consequence, useful information in other domains has not been fully utilized. To address such limitations, we collect a dataset named GP10, which consists of 1082 primary insect pest images in 10 species. Moreover, we involve discrete wavelet transform (DWT) in a convolutional neural network to construct a novel triple-attention network (FcsNet) combined with frequency, channel and spatial attention modules. Next, we compare the network performance and parameters against several state-of-the-art networks based on different attention mechanisms. We evaluate the proposed network on our dataset GP10 and open dataset D0, achieving classification accuracy of 73.79% and 98.16%. The proposed network obtains more than 3% accuracy gains on the challenging dataset GP10 with parameters and computation operations slightly increased. Visualization with gradient-weighted class activation mapping (Grad-CAM) demonstrates that FcsNet has comparative advantages in image classification tasks.

Keywords: grain pest classification; visual attention mechanism; discrete wavelet transform; deep learning; computer vision

Citation: Yu, J.; Shen, Y.; Liu, N.; Pan, Q. Frequency-Enhanced Channel-Spatial Attention Module for Grain Pests Classification. *Agriculture* **2022**, *12*, 2046. https://doi.org/10.3390/agriculture12122046

Academic Editor: Yanbo Huang

Received: 23 October 2022
Accepted: 27 November 2022
Published: 29 November 2022

1. Introduction

Grains including cereals and legumes provide food for humans and livestock. Insect infestation is one of the leading factors affecting the quantity, quality, nutrition and market value of stored grains. Insect infestation during storage accounts for about 6–10% of postharvest grain losses, which poses serious challenges to food security in many countries [1]. In the European standards of Storage of Cereals and Pulses, ISO 6322-3 gives guidance on controlling attacks by 23 insect and mite pests. In the Chinese Technical Criterion for Grain and Oil-seed Storage (GB/T 29890-2013) [2], ten primary insect pests are specified to be identified. The species of ten primary insect pests are araecerus fasciculatus (AF, coffee bean weevil), bruchus pisorum (BP, pea weevil), bruchus rufimanus boheman (BRB, broadbean weevil), callosobruchus chinensis (CC, azuki bean weevil), plodia interpunctella (PI, Indian meal moth), rhizopertha dominica (RD, lesser grain borer), sitophilus oryzae (SO, rice weevil), sitophilus zeamais (SZ, maize weevil), sitotroga cerealella (SC, angoumois

grain moth) and tenebroides mauritanicus linne (TML, cadelle beetle). Furthermore, the unprocessed grain can be graded into basically clear grain ($\leq$2 insects per kg), regular occurrence of insect grain (3–10 insects per kg), and intense occurrence of insect grain (>10 insects per kg), according to the population density of these ten primary insect pests. Therefore, grain insect identification and population destiny estimation are necessary for applying proper control actions.

The popular methods of insect detection and identification are visual inspection, probe sampling, acoustic detection, electronic nose and imaging methods [3]. Among them, the conventional methods such as visual inspection, trap methods and probe sampling are time-consuming and labor-intensive. Modern methods such as acoustic detection and electronic nose are costly and unreliable in noisy and complex environments. With the advancement of computer vision, image processing-based methods are proved to be more suitable for identification and classification of grain insects.

Traditional image processing methods utilize color, edge, corner, key point or other low-level features to recognize the grain pests [4–7]. For example, the United States Department of Agriculture (USDA) used visual reference images for insect detection and grain grading since 1997. Ridgway et al. [8] developed a non-touching method based on machine vision to detect saw-toothed grain beetles. Wen et al. [9] proposed a hierarchical model that combined both local features and global features to identify orchard insects.

Thanks to huge volumes of image data, convolutional neural networks (CNN) achieve great success in image classification, object detection, image segmentation and other visual tasks. CNN-based deep learning models such as ResNet [10] and VGGNet [11] have already surpassed human-level accuracy in image classification. Albeit the progress has been made in common object classifications, grain insect pest classification is still a challenging task in the practical application. As ten primary insect pests specified in GB/T 29890-2013 occur in three groups: grain weevils, grain borers and grain moths, among each group, the insects are difficult to distinguish. On the other hand, the attributes of different shapes, small sizes, multi-colors and cluttered backgrounds also pose challenges on grain insect classification. Motivated by the fact that humans and birds can find the insects in grains effectively, we introduce frequency, channel and spatial attention mechanisms into the image classification models.

This paper focuses on the frequency-enhanced attention mechanism, which integrates more clues to improve the accuracy of grain insect classification. The main contributions of this paper can be summarized as follows.

(1) We collect a challenging dataset of 10 species of stored-grain insects specified by the standard GB/T 29890-2013.

(2) We construct a novel triple-attention network (FcsNet) combined with frequency, channel and spatial attention modules. The frequency information of discrete wavelet transform (DWT) and discrete cosine transform (DCT) are involved in the convolutional neural network. FcsNet can be plugged into classic backbone networks as an efficient add-on module.

(3) Extensive experiments and ablation studies are carried out on the proposed dataset GP10 and open dataset D0. More insights into the frequency-enhanced attention mechanism can be found in the visualization results of the confusion matrix and Grad-CAM.

2. Related Works

In order to process the information received visually more efficiently, people are used to paying attention to some of the information while ignoring other visible information. Inspired by human vision, a new method for data processing is proposed, called attention mechanism. The attention mechanism is essential to add different weights to each part of the input information, so that the model could pay attention to areas which are more significantly weighted and thus improves the accuracy of model judgment.

To solve the problems caused by pests, Cheng et al. [12] established a system that can identify agricultural pests in a complex background using a convolutional neural

network (CNN) and residual network. This system has 98.67% accuracy for classifying the images of 10 species of agricultural pests, which is better than the ordinary deep neural network AlexNet [13]. Nanni et al. [14] proposed an automatic pest classification model by combining CNN and significance methods, but these methods [12,14] do not introduce an attention mechanism. Xie et al. [15] published a large field crop pest dataset (D0). The dataset contains about 4500 images of 40 species of field crop pests. However, the background of this dataset is single and the pose of pests is similar, which makes it easy to extract pest features. Ung et al. [16] followed a residual attention network (RAN), feature pyramid network (FPN) and a multi-branch multi-scale attention network (MMAL-Net) to improve the accuracy of the final pest classification based on integration technology and in accordance with the prediction results of the above three networks. However, they used only one attention mechanism. Zhou et al. [17] proposed an efficient small-scale convolutional neural network for pest identification, which is composed of a double fusion with a squeeze-and-excitation-bottleneck block (DFSEB block) and a max feature expansion block (ME block). Li et al. [18] developed a multi-scale insect detector (MSI_Detector) by constructing a feature pyramid to extract stored-grain insect image features with different spatial resolutions and semantic information. Shi et al. [19] proposed a multi-class stored-grain insect object detection network based on R-FCN (Region-based fully convolutional network) which achieves both high classification accuracy and speed.

In the development of attention in computer vision, common attention mechanisms can be divided into spatial attention and channel attention. Spatial attention can be viewed as an adaptive spatial region selection mechanism, and using it can directly predict the most relevant spatial locations [20,21] or select important spatial regions [22]. Hu et al. [23] captured long-range spatial context information by gather and excite operations, and they designed the GENet model, which not only emphasizes on important features, but also suppresses noise. Li et al. [24] viewed self-attention in terms of expectation maximization (EM) and proposed EM attention. Huang et al. [25] treat the self-attention operation as graph convolution and proposed cross-attention. Compared with the previous self-attention-based spatial attention [22], it improves the speed and generalization capability. Channel attention adaptively recalibrates the weight of each channel, and can be viewed as an object selection process, thus determining to what to pay attention. Hu et al. [26] proposed a new architecture unit based on ResNet [10], which is called a squeeze-and-excitation network (SENet) block. They compared the performance of global average pooling (GAP) and global maximum pooling (GMP) as squeeze operators, and finally adopted GAP to calculate the channel attention. Gao et al. [27] proposed the global second-order pooling (GSoP) block to address the limited ability of the SE block to capture global information. To overcome the high model complexity, Wang et al. [28] proposed an efficient channel attention (ECA) block. This block introduces one-dimensional convolution to reduce the redundancy of fully connected layers and obtain more efficient results. Moreover, Woo et al. [29] found that the combination of two kinds of attention has better performance through ablation experiments, and proposed the convolutional block attention module (CBAM). From another perspective, Qin et al. [30] regarded the channel representation problem in SENet as a compression process using frequency analysis, and proposed a new multi-spectral channel attention method (FcaNet) with the performance superior to that of SENet. Guo et al. [31] surveyed attention models in deep neural networks and encouraged various studies to improve deep learning results by using attention mechanisms.

3. Materials and Methods

3.1. Residual Networks

He et al. [32] proposed a residual network (ResNet) in 2015. This network solved the network degradation problem caused by too many hidden layers in the deep neural network (DNN) (this degradation is not caused by overfitting), abandoned the dropout and used Batch Normalization (BN) for training acceleration. In addition, it introduced the shortcut connection between the input and output to avoid gradient disappearance and

explosion in the DNN training. After these problems are solved, the depth of the network rose by several orders of magnitude.

The structure of ResNet can not only speed up the training of neural networks very quickly and improve the accuracy of the model, but it is also easy to optimize. Therefore, ResNet has become the basis for many research tasks, including classification, detection and segmentation. In other words, ResNet is suitable for backbone networks.

3.2. Channel Attention Module

The channel attention mechanism was proposed by Hu et al. [26] in 2017. It can reallocate the feature weight on the channel based on a new "feature recalibration" strategy, which has improved effective features and suppressed invalid feature information. Moreover, Woo et al. [29] noted that the global maximum pooling (GMP) also plays a role in channel attention, and has modified it, as shown in Figure 1. All above can be summarized as follows:

$$C = \mathrm{F}_{\mathrm{cbam}}\left(\mathrm{X}, \theta\right) = \sigma(\mathrm{W}_2\delta(\mathrm{W}_1\mathrm{GAP}(\mathrm{X})) + \mathrm{W}_2\delta(\mathrm{W}_1\mathrm{GMP}(\mathrm{X}))) \quad (1)$$

where X represents the input, GAP and GMP represent the global average pooling and global maximum pooling operations, respectively, $\mathrm{W_i}$ represents the weight of the full connection layer, and the δ and σ distribution represents ReLU and Sigmoid functions.

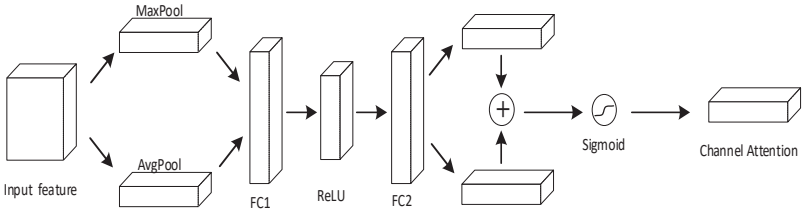


Figure 1. Diagram of channel attention module (CAM). As illustrated, the channel attention module utilizes both max-pooling outputs and average-pooling outputs and forward them to the fully connected layer, which finally generates channel attention through the sigmoid function.

3.3. Spatial Attention Module

At the same time, Woo et al. [29] noted the importance of spatial attention and proposed a convolutional block attention module (CBAM). They found that spatial attention and channel attention are complementary. Unlike channel attention, the spatial attention focuses on "where" the information part lies. In the study of spatial attention, they compared the convolution kernels of different sizes and found that a larger convolution kernel can produce better accuracy. This shows that a wider receptive field is needed in spatial attention. As shown in Figure 2, it can be written as follows:

$$S = \sigma(\mathrm{Conv}([\mathrm{GAP}(\mathrm{X}); \mathrm{GMP}(\mathrm{X})])) \quad (2)$$

where Conv(·) represents a convolution operation.

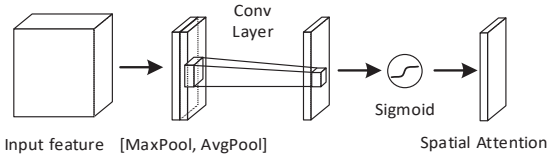


Figure 2. Diagram of spatial attention module (SAM). As illustrated, the spatial attention module forwards max pooling outputs and average pooling outputs to the convolution layer and generates spatial attention through the sigmoid function.

3.4. Frequency Attention Module

In addition to the channel and spatial attention modules, Qin et al. [30] also proposed a frequency domain channel attention network (FcaNet). Based on SENet, they regarded the channel representation problem as a compression process using frequency analysis, and analyzed GAP in the frequency domain. They mathematically proved that GAP is a special case of characteristics in the frequency domain and proposed a new multi-spectral channel attention method based on such discovery.

GAP is used to calculate the mean value of all spatial elements in each channel. However, different channels may have the same mean value, but have different semantics, which leads to poor diversity of features obtained through GAP. The discrete cosine transform (DCT) is a kind of Fourier transform and is often used to compress signals and images, and the two-dimensional DCT contains more frequency components, including the lowest frequency component GAP.

Specifically, it first divides the input images into several groups and then conducts two-dimensional DCT processing for each group. Finally, similar to SENet processing, the final weight is obtained by using the full connection layer, ReLU and Sigmoid functions. This can be written as follows:

$$S = F_{fca}(X, \theta) = \sigma(W_2\delta(W_1[(DCT(Group(X)))])) \tag{3}$$

where DCT represents 2D discrete cosine transform while Group represents dividing the input into several groups.

Li et al. [33] found that the down-sampling (max-pooling, average-pooling and strided-convolution) in deep learning often amplifies random noise and destroys the basic results of the target. They used Discrete Wavelet Transform (DWT) to replace the down-sampling operation in the network to improve the robustness of model classification.

DWT can decompose the one-dimensional signal $s = \{s_j\}_{j\in\mathbb{Z}}$ into low-frequency components $s_1 = \{s_{1k}\}_{k\in\mathbb{Z}}$ and high-frequency components $d_1 = \{d_{1k}\}_{k\in\mathbb{Z}}$, which can be written as follows:

$$\begin{cases} s_{1k} = \sum_j l_{j-2k} s_j \\ d_{1k} = \sum_j h_{j-2k} s_j \end{cases} \tag{4}$$

where $l = \{l_k\}_{k\in\mathbb{Z}}$ and $h = \{h_k\}_{k\in\mathbb{Z}}$ are respectively low-pass and high-pass filters of the orthogonal wavelet.

If expressed by vectors and matrices, the formula (4) can be written as:

$$s_1 = Ls, d_1 = Hs \tag{5}$$

where L and H are, respectively:

$$L = \begin{pmatrix} \cdots & \cdots & \cdots & & & & \\ \cdots & l_{-1} & l_0 & l_1 & \cdots & & \\ & & \cdots & l_{-1} & l_0 & l_1 & \cdots \\ & & & & & \cdots & \cdots \end{pmatrix} \tag{6}$$

$$H = \begin{pmatrix} \cdots & \cdots & \cdots & & & & \\ \cdots & h_{-1} & h_0 & h_1 & \cdots & & \\ & & \cdots & h_{-1} & h_0 & h_1 & \cdots \\ & & & & & \cdots & \cdots \end{pmatrix} \tag{7}$$

For a 2D signal X, DWT usually performs one-dimensional DWT on each row and column, namely:

$$X_{ll} = LXL^T \tag{8}$$

$$X_{lh} = HXL^T \tag{9}$$

$$X_{hl} = LXH^T \tag{10}$$

$$X_{hh} = HXH^T \tag{11}$$

DWT decomposes an image X into high-frequency components X_{lh}, X_{hl} and X_{hh} and low-frequency component X_{ll}. X_{ll} is the low-resolution version of the image it keeps the most energy and basic structure of the image. While X_{lh}, X_{hl} and X_{hh} represent the image details that include edges and noise. Therefore, the DWT coefficients can be integrated into the convolution neural network to extract useful features for object classification.

3.5. Proposed Method

In this work, we believe that channel attention, spatial attention and frequency domain attention focus on the target area in the image from different dimensions. We speculate that if these three attention modules are combined, the network's overall performance will be improved by mutual complementation. Based on the three attention modules and DWT down-sampling operation, we proposed a novel triple-attention network (FcsNet). Figure 3 shows the schematic diagram of the network we proposed.

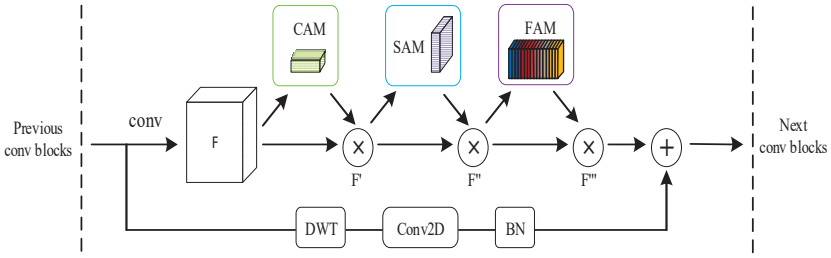


Figure 3. FcsNet integrated with a ResBlock in ResNet. This figure shows the exact position of our module when integrated within a ResBlock. We apply FcsNet on the convolution outputs in each block. Therein, the condition for DWT operation is Stride equal to 2.

To compare the network structures of ResNet and FcsNet (ours), we list their details in Table 1, where DWT^1 represents the wavelet transform substituting max-pooling operation and DWT^2 represents the wavelet transform substituting convolution operation with stride 2. CAM, SAM and FAM represent channel, spatial and frequency attention modules, respectively.

Table 1. Network structure of ResNet-50 and Fcs-ResNet-50(ours). The shapes and operations with specific parameter settings of a residual block are shown in brackets, with the numbers of blocks stacked. The right side shows different down-sampling performed by conv3_1, conv4_1, and conv5_1 with a stride of 2.

Layer Name	Output Size	ResNet-50		Fcs_ResNet-50	
conv1	112 × 112	conv, 7 × 7, 64, stride 2			
conv2_x	56 × 56	max pool, 3 × 3, stride 2		DWT[1]	
		$\begin{bmatrix} \text{conv}, 1 \times 1.64 \\ \text{conv}, 3 \times 3.64 \\ \text{conv}, 1 \times 1.256 \end{bmatrix} \times 3$		$\begin{bmatrix} \text{conv}, 1 \times 1.64 \\ \text{conv}, 3 \times 3.64 \\ \text{conv}, 1 \times 1.256 \\ \text{CAM} + \text{SAM} + \text{FAM} \end{bmatrix} \times 3$	
conv3_x	28 × 28	$\begin{bmatrix} \text{conv}, 1 \times 1.128 \\ \text{conv}, 3 \times 3.128 \\ \text{conv}, 1 \times 1.512 \end{bmatrix} \times 4$	Conv2D BN	$\begin{bmatrix} \text{conv}, 1 \times 1.128 \\ \text{conv}, 3 \times 3.128 \\ \text{conv}, 1 \times 1.512 \\ \text{CAM} + \text{SAM} + \text{FAM} \end{bmatrix} \times 4$	DWT^2 Conv1 × 1 BN
conv4_x	14 × 14	$\begin{bmatrix} \text{conv}, 1 \times 1.256 \\ \text{conv}, 3 \times 3.256 \\ \text{conv}, 1 \times 1.1024 \end{bmatrix} \times 6$		$\begin{bmatrix} \text{conv}, 1 \times 1.256 \\ \text{conv}, 3 \times 3.256 \\ \text{conv}, 1 \times 1.1024 \\ \text{CAM} + \text{SAM} + \text{FAM} \end{bmatrix} \times 6$	

Table 1. *Cont.*

Layer Name	Output Size	ResNet-50	Fcs_ResNet-50
conv5_x	7×7	$\begin{bmatrix} \text{conv}, 1 \times 1.512 \\ \text{conv}, 3 \times 3.512 \\ \text{conv}, 1 \times 1.2048 \end{bmatrix} \times 3$	$\begin{bmatrix} \text{conv}, 1 \times 1.512 \\ \text{conv}, 3 \times 3.512 \\ \text{conv}, 1 \times 1.2048 \\ \text{CAM} + \text{SAM} + \text{FAM} \end{bmatrix} \times 3$
	1×1	global average pool, 10-d fc, softmax	

4. Experiments and Results

In this section, firstly we explained our experiment. Secondly, in order to better compare our dataset (GP10) and D0 dataset [15], we rebuilt all evaluated networks [10,26,29,30] in the PyTorch framework, and used standard evaluation indicators to compare with the performance of previous methods. Finally, we studied the effectiveness of our method in the classification of grain pest images.

4.1. Datasets

We evaluated our proposed method on two datasets. We collected the first dataset (GP10), including 1082 pictures of 10 species of stored grain pests, namely, araecerus fasciculatus (AF, coffee bean weevil), bruchus pisorum (BP, pea weevil), bruchus rufimanus boheman (BRB, broad bean weevil), callosobruchus chinensis (CC, azuki bean weevil), plodia interpunctella (PI, Indian meal moth), rhizopertha dominica (RD, lesser grain borer), sitophilus oryzae (SO, rice weevil), sitophilus zeamais (SZ, maize weevil), sitotroga cerealella (SC, angoumois grain moth) and tenebroides mauritanicus linne (TML, cadelle beetle). Figure 4 shows some sample images of our dataset.

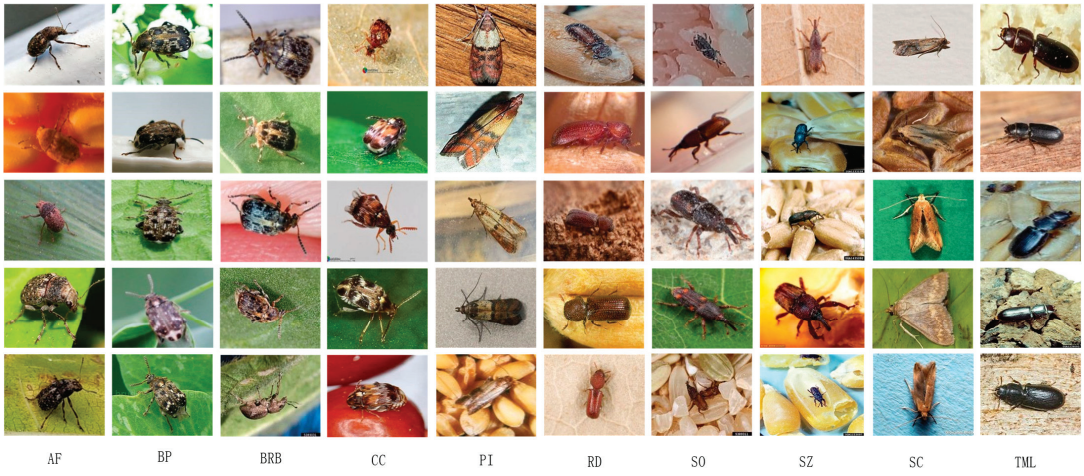


Figure 4. Sample images collected in GP10.

While collecting these samples, we relied on common image specimen search engines, including iNaturalist and Bugwood Images, etc. iNaturalist is a global community containing biodiversity data, whose goal is to promote biodiversity discipline and conservation. Bugwood Images is a funded project launched by the Center for Invasive Species and Ecosystem Health of the University of Georgia in 1994. It provides an accessible high-quality image archive and focuses on species related to economy, including insects, plants, agriculture and integrated pest management, etc.

We used the English name and corresponding synonyms of each subcategory as query keywords to search and download samples of the corresponding category. Secondly, we

searched and learned the structural characteristics of each type of stored grain pests on professional insect science websites to screen and verify each type of sample. Thirdly, we cut each type of picture according to size requirement for convenient model training later.

The second dataset is D0 (4500 pictures in all), including 40 different pests. Some are shown in Figure 5.

Figure 5. Example images in D0.

4.2. Experiment Settings

Our dataset is divided into three subsets: training set images (876 pcs), verification set images (103 pcs) and test set images (103 pcs), subject to the ratio of 8:1:1. See Table 2 for detailed classification. In order to obtain sufficient target features, we first expanded the training set to 2628 images by flipping horizontally and adding Gaussian noise. To make the experiment more normal and impartial, we first used python script to divide the three subsets at random, with no duplicate images present in these three subsets. The same set of division data was used in the subsequent experiments. Similarly, the same settings were used on dataset D0.

Table 2. Composition of the D0 dataset.

Species	Abbreviations	Number of Samples	Train	Val	Test
Araecetus fasciculatus	AF	115	93	11	11
Bruchus pisorum	BP	110	88	11	11
Bruchus rufimanus Boheman	BRB	97	79	9	9
Callosobruchus chinensis	CC	83	67	8	8
Plodia interpunctella	PI	129	105	12	12
Rhizopertha dominica	RD	69	57	6	6
Sitophilus oryzae	SO	176	142	17	17
Sitophilus zeamais	SZ	83	67	8	8
Sitotroga cerealella	SC	115	93	11	11
Tenebroides mauritanicus Linne	TML	105	85	10	10
Total		1082	876	103	103

We processed the input images in advance. Firstly, we applied random clipping to the training set and adjusted its size to 224 × 224. Then, we used the method of randomly changing brightness, contrast and saturation to enhance the generalization of the model and solve the problem of overfitting. In the verification set, firstly, we adjusted the minimum edge of the image to 256, with the aspect ratio of the original image maintained. Then, we used the center clipping method to cut the image size to 224 × 224. Finally, we applied the center clipping method with the same size as the training window in the test phase. For more convenient training, we converted the data into Tensor format and standardized the data accordingly.

In the phase of training, we used the multi-class cross entropy as the cost function. Then, we used the Adam optimizer with a learning rate of 10^{-4} to optimize the network parameters. Next, we set the small batch to 32 and conducted 200 epochs of training. Finally, we saved the optimal training parameters and tested their predictions.

4.3. Evaluation Metrics

Because of the imbalanced class distribution of our dataset, we employed several comprehensive metrics for the classification task, including parameters (params), floating point operations (FLOPs), accuracy (acc), average precision (MPre), average recall (MRec), average F1-score (MF1), receiver operating characteristic (ROC) curve and area under the roc curve (AUC).

FLOPs are mainly used to describe the computation of a model, which is similar to the time complexity of an algorithm.

For the convolution kernel, we compute FLOPs as follows:

$$\text{FLOPs\_c} = 2\text{HW}\left(\text{C}_{\text{in}}\text{K}^2 + 1\right)\text{C}_{\text{out}} \tag{12}$$

where H, W and C_{in} are the respective height, width and number of channels of the input feature map, K is the kernel width (assumed to be symmetric), and C_{out} is the number of output channels.

For fully connected layers, we compute FLOPs as follows:

$$\text{FLOPs\_fc} = (2\text{I} - 1)\text{O} \tag{13}$$

where I is the input dimension and O is the output dimension.

Params is mainly used to describe the size of a model, which is similar to the spatial complexity of an algorithm.

The parameter number of the convolution layer is calculated as follows:

$$\text{params\_c} = C_o \times \left(k^2 \times C_i\right) \tag{14}$$

where C_o is the number of output channels, C_i is the number of input channels, and K is the kernel width (assumed to be symmetric). If the convolution kernel has a bias term, it will be added by one, and if not, it will not be added.

The number of parameters of the full connection layer is calculated as follows:

$$\text{params\_fc} = (I + 1) \times O = I \times O + O \tag{15}$$

where I is the length of the input vector and O is the length of the output vector.

Acc is the proportion of the true positive value to the total predicted value among all classes as follows:

$$\text{Acc} = \frac{TP}{N} \tag{16}$$

where N is the number of samples and TP is true positive. Pre is the proportion of positive values in the total number of categories. To treat the classes as being equally important, we computed the precision for each category, then took an average of them to obtain MPre as follows:

$$\text{Pre}_c = \frac{TP_c}{TP_c + FP_c} \tag{17}$$

$$\text{MPre} = \frac{\sum_{c=1}^{C} \text{Pre}_c}{C} \tag{18}$$

where C is the number of classes. FP_c and TP_c stand for the false positive and the true positive of the $c-$ th class, respectively. Similarly, we computed Rec and MRec as follows:

$$\text{Rec}_c = \frac{TP_c}{TP_c + FN_c} \tag{19}$$

$$\text{MRec} = \frac{\sum_{c=1}^{C} \text{Rec}_c}{C} \tag{20}$$

where FN_c stands for the false negative of the $c-$ th class. The F1 combines the MPre and MRec as a trade-off as follows:

$$\text{MF1} = 2\frac{\text{MPre} \cdot \text{MRec}}{\text{MPre} + \text{MRec}} \tag{21}$$

In addition, the ROC (receiver operating characteristic) curve is used to compare the classification performance of the models. The vertical axis of the ROC curve represents the true-positive rate (TPR), and the horizontal axis represents the false-positive ratio (FPR). The higher the TPR and the lower the FPR, the better the performance of the model. In other words, the closer the ROC curve is to the upper left corner, the higher the model prediction results. TPR and FPR are defined as follows:

$$\text{TPR} = \frac{TP}{TP + FN} \tag{22}$$

$$\text{FPR} = \frac{FP}{TN + FP} \tag{23}$$

where TP, FP, FN and TN refer to true positive, false positive, false negative and true negative, respectively. The ROC curve is difficult to distinguish the performance gap between models, so we choose AUC (area under the roc curve) as the evaluation metric. The

AUC is between [0, 1], and the closer its value to 1, the better the classification performance of the model. The AUC definition is as follows:

$$AUROC = \int TPRd(FPR) \tag{24}$$

4.4. Experimental Results

4.4.1. Verification on Private Dataset

In accordance with the evaluation criteria in Section 4.3, we first compare the performance and efficiency of the proposed model with existing attention mechanisms on the dataset GP10 and D0, then report the results in Table 3. We observed that our method performs best on Acc, MPre, MRec and MF1. FcsNet achieves 11.65%, 9.71%, 5.83% and 3.89% accuracy gain than ResNet, SENet, CBAM and FcaNet, respectively. This means that our method is effective. This method can combine the attention of frequency domain, channel and space, and use DWT for down-sampling to improve the accuracy significantly.

Table 3. The performance comparison of different networks on GP10 and D0 datasets.

Architecture	Backbone	Params	FLOPs	GP10				D0			
				Acc	MPre	MRec	MF1	Acc	MPre	MRec	MF1
ResNet	ResNet-50	23.53 M	4.12G	62.14	64.74	61.17	61.71	96.08	96.50	95.61	95.82
SENet	ResNet-50	26.04 M	4.13G	64.08	69.30	64.62	63.93	97.00	97.49	96.79	97.00
CBAM	ResNet-50	26.05 M	4.14G	67.96	71.37	67.16	67.45	97.47	97.76	97.28	97.40
FcaNet	ResNet-50	26.04 M	4.13G	69.90	69.88	68.77	68.06	97.63	98.19	97.62	97.81
FcsNet(ours)	ResNet-50	28.56 M	5.18G	73.79	74.38	72.79	71.99	98.16	98.49	98.33	98.34

Furthermore, we analyzed the complexity of this method from two aspects such as learnable parameters (Params) and floating point operations per second (FLOPs). For parameters, our method increased by 9.6% and 9.7%, respectively, compared with CBAM and FcaNet. For the FLOPs, our method increased by 25.4% and 25.1%, respectively, compared with CBAM and FcaNet.

Our method (FcsNet) achieved a confusion matrix as shown in Figure 6. It can be found that obvious errors are caused by several similar categories which belong to the same genus and have many common features. For example, BP and BRB belong to the same genus of bruchus, SO and SZ belong to the same genus of sitophilus.

Figure 7 shows the prediction probability of SO and SZ. Because of the similar morphology of SO and SZ, there are two prediction probabilities much bigger than the other categories. This means these two categories are often misclassified. If the top-2 error rate is considered, the accuracy will be greatly improved on the proposed dataset GP10. This also confirms that the above-mentioned categories of the same genus have common featuresand pose challenges to our research.

In order to eliminate the influence of sample imbalance, we draw the ROC curve of each model to intuitively represent the prediction ability of each model. We also calculated the AUC to make it clear which model performed better. This is shown in Figure 8. By comparison, it is easy to find that, although our model is slightly inferior to FcaNet and CBAM in the beginning, the performance of our model is slightly higher than other models in general.

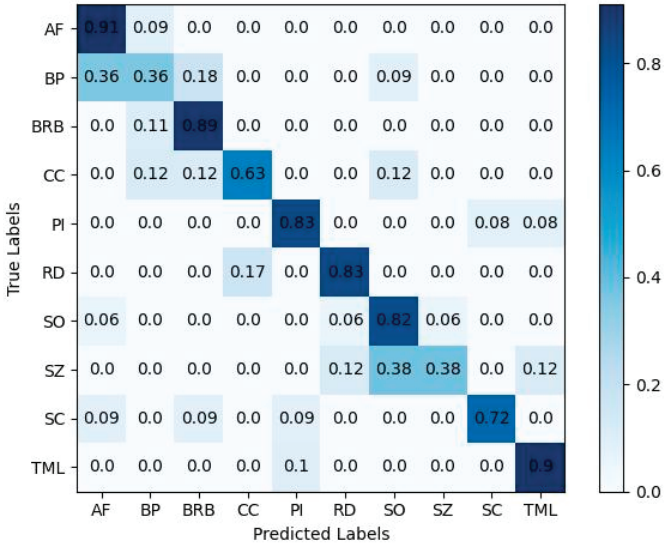


Figure 6. Confusion matrix of the proposed method. The vertical axis is the true label, and the horizontal axis represents the predicted label. The values in the diagonal area in the figure are the proportion of correct predictions, and the other values are the proportions of wrong predictions. The darker the color, the larger the proportion.

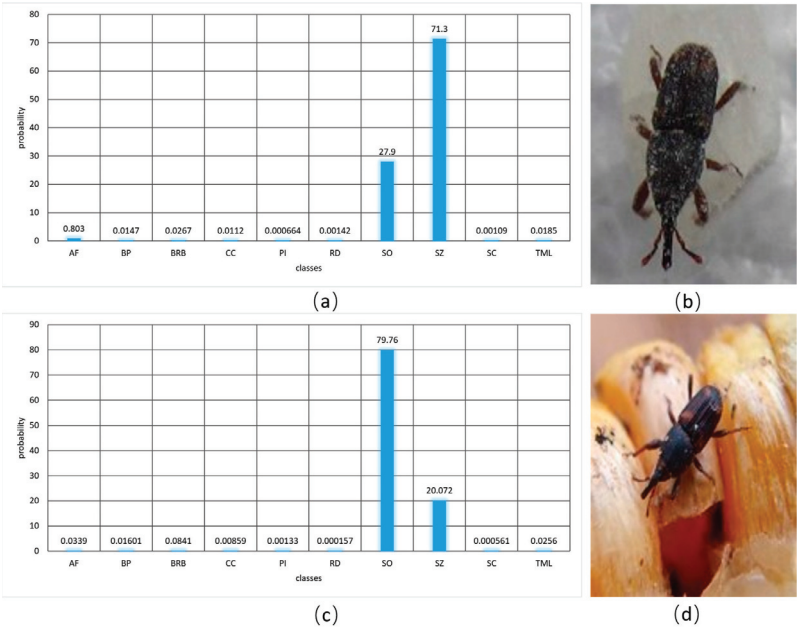


Figure 7. Comparison of SO and SZ prediction results, where (**a**,**c**) are bar charts of the prediction probability of SO and SZ (in percentage). The horizontal axis is the class name and the vertical axis is the probability. Examples of images for SO and SZ are shown in (**b**,**d**), respectively.

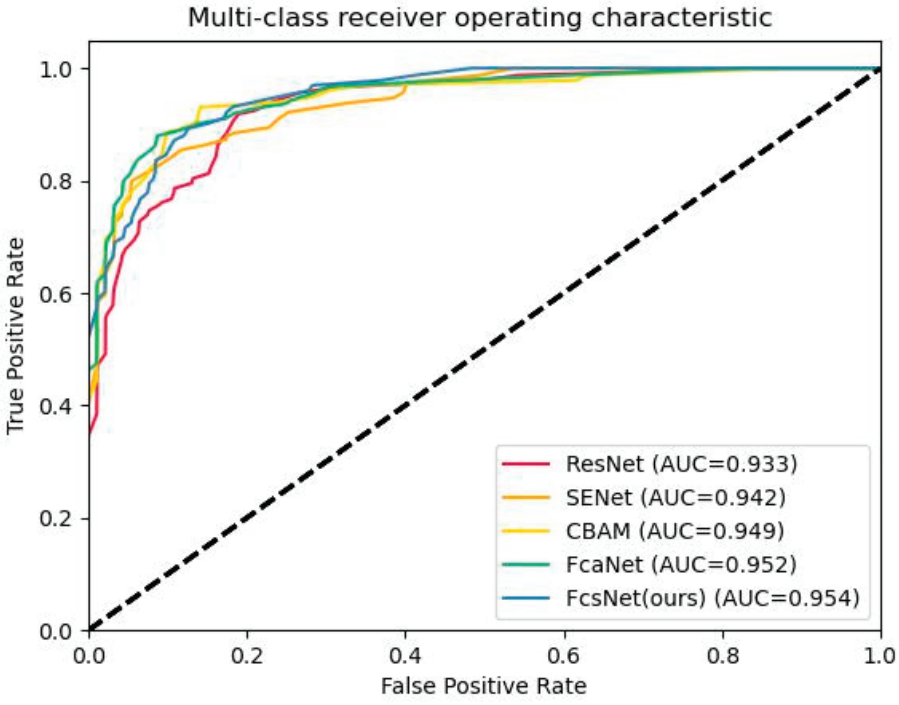


Figure 8. ROC comparison of different models. The horizontal axis is the false positive rate, the vertical axis is the true positive rate, and the lower right corner is the color and AUC value corresponding to the model.

4.4.2. Verification on Open Dataset

In the field of pest images, the open dataset D0 of Xie et al. [15] is often used as a standard dataset to verify proposed methods for classification. In order to further verify the performance of the proposed method, we used this dataset as supplementary proof. We observe that FcsNet is superior to other architectures in every comparison, which indicates that the benefits of FcsNet are not limited to our dataset (GP10). See Table 3 for details.

Through comparison, it is not difficult to find that the accuracy on the dataset GP10 is not as high as that on D0. Based on analysis, we concluded the following two reasons. Firstly, the images on dataset D0 have a similar background and the pest postures change slightly. In Figure 5, we give images of some categories. Secondly, our dataset (GP10) has a complex background and a high degree of similarity exists in appearance between different categories. Therefore, classification on the GP10 dataset is more challenging.

4.5. Visualization with Grad-CAM

This section shows the visualization of our proposed model. Previously, it was believed that the deep learning network was a black box and lacked some explanatory power, for example, in classification network models (such as VGGNet [11], ResNet [10] and MobileNet [34]), and it was unclear why the network predicted like this and where the concerns were for each category. Zhou et al. [35] proposed a kind of category activity mapping technology, which can draw a thermodynamic chart to show to which areas the network pays attention, and also where the network structure needs to be changed and retraining carried out. Moreover, Selvaraju et al. [36] upgraded and improved it based on category activity mapping to make the existing most advanced deep model interpretable without changing its architecture, thus avoiding the tradeoff between interpretability and accuracy.

Figure 9 shows the Grad-CAM [36] generated by ResNet, SENet, CBAM, FcaNet and FcsNet based on the input images of our test set. As can be seen, FcsNet includes the focus of other models in the focus input image, and it seems to focus more on the whole area of the grain pests. This also confirms the effectiveness of our proposed method.

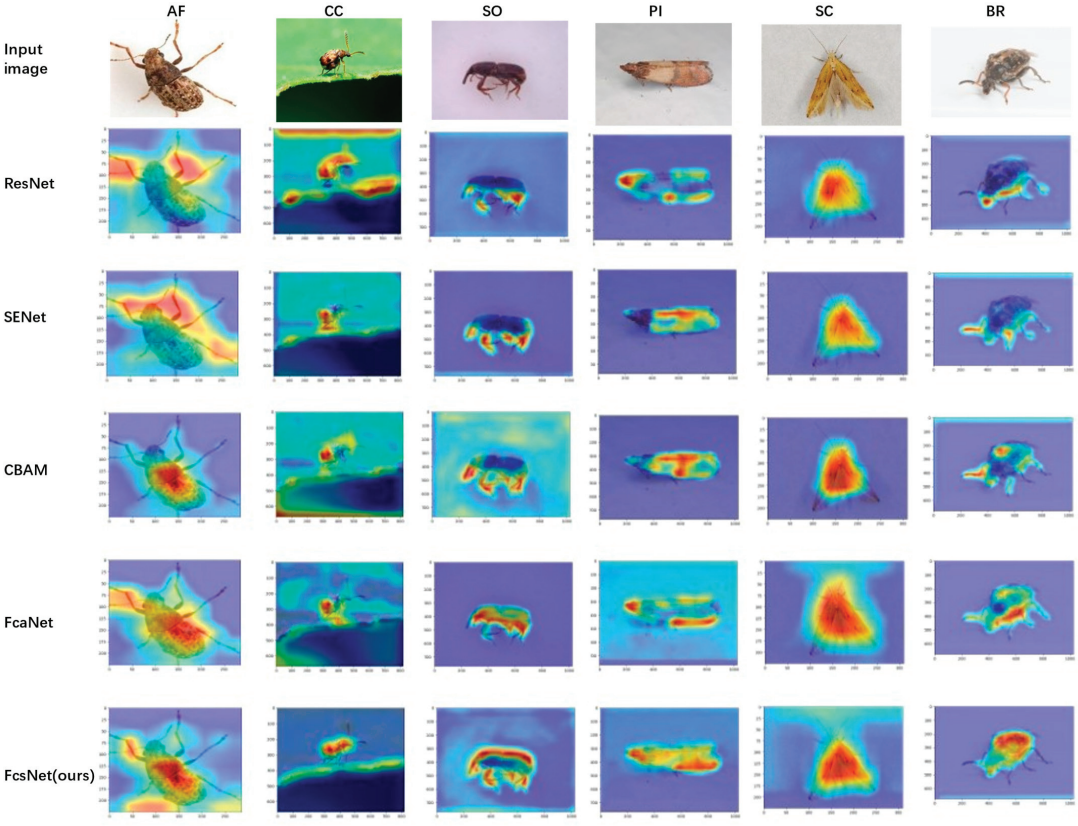


Figure 9. The Grad-CAM visualization results. We compared the visualization results from ResNet, SENet, CBAM, FcaNet and FcsNet, and calculated the gradient CAM visualization of the final convolution output.

5. Conclusions

In this paper, we propose a stored grain pest identification method based on a triple-attention module (FCS), namely, frequency domain attention (FAM), channel attention (CAM) and spatial attention (SAM). We combine the three domains and use wavelet transform for down-sampling to achieve considerable improvement in performance while maintaining a low overhead, and verified on our dataset (GP10) and D0, with the accuracy rates being 73.79% and 98.16%, respectively. FcsNet has good performance and can provide a new idea and method for the rapid detection and identification of pests. In the future, our work will focus on using multi-domain attention mechanisms to solve pest detection and segmentation tasks.

Author Contributions: Conceptualization, J.Y. and N.L.; Methodology, Y.S. and J.Y.; software, Y.S. and Q.P.; data curation, Y.S.; writing, Y.S. and J.Y.; writing—review and editing, Y.S. and J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Key R&D and Promotion Projects of Henan Province (Science and Technology Development, 212102210152); the Innovative Funds Plan of Henan University of Technology(2021ZKCJ14); the Young Backbone Teacher Training Program of Henan University of Technology (2015006).

Institutional Review Board Statement: The authors are grateful to the editors and anonymous viewers for their valuable and insightful comments and suggestions.

Data Availability Statement: The data are not publicly available because the data need to be used in future work.

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

References

1. Abdullahï, N.; Dandago, M.A. Postharvest Losses in Food Grains—A Review. *Turk. J. Food Agric. Sci.* **2021**, *3*, 25–36. [CrossRef]
2. *GB/T 29890-2013*; Chinese Technical Criterion for Grain and Oil-Seeds Storage. Standards Press of China: Beijing, China, 2013. (In Chinese)
3. Banga, K.S.; Kotwaliwale, N.; Mohapatra, D.; Giri, S.K. Techniques for Insect Detection in Stored Food Grains: An Overview. *Food Control* **2018**, *94*, 167–176. [CrossRef]
4. Bay, H.; Ess, A.; Tuytelaars, T.; Van Gool, L. Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.* **2008**, *110*, 346–359. [CrossRef]
5. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
6. Oliva, A.; Torralba, A. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *Int. J. Comput. Vis.* **2001**, *42*, 145–175. [CrossRef]
7. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
8. Ridgway, C.; Davies, E.R.; Chambers, J.; Mason, D.R.; Bateman, M. Rapid Machine Vision Method for the Detection of Insects and Other Particulate Bio-Contaminants of Bulk Grain in Transit. *Biosyst. Eng.* **2002**, *83*, 21–30. [CrossRef]
9. Wen, C.; Guyer, D. Image-Based Orchard Insect Automated Identification and Classification Method. *Comput. Electron. Agric.* **2012**, *89*, 110–115. [CrossRef]
10. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity Mappings in Deep Residual Networks. In Proceedings of the Computer Vision—ECCV, Amsterdam, The Netherlands, 11–14 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 630–645.
11. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
12. Cheng, X.; Zhang, Y.; Chen, Y.; Wu, Y.; Yue, Y. Pest Identification via Deep Residual Learning in Complex Background. *Comput. Electron. Agric.* **2017**, *141*, 351–356. [CrossRef]
13. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
14. Nanni, L.; Maguolo, G.; Pancino, F. Insect Pest Image Detection and Recognition Based on Bio-Inspired Methods. *Ecol. Inform.* **2020**, *57*, 101089. [CrossRef]
15. Xie, C.; Wang, R.; Zhang, J.; Chen, P.; Dong, W.; Li, R.; Chen, T.; Chen, H. Multi-Level Learning Features for Automatic Classification of Field Crop Pests. *Comput. Electron. Agric.* **2018**, *152*, 233–241. [CrossRef]
16. Ung, H.T.; Ung, H.Q.; Nguyen, B.T. An Efficient Insect Pest Classification Using Multiple Convolutional Neural Network Based Models. *arXiv* **2021**, arXiv:2107.12189.
17. Zhou, S.-Y.; Su, C.-Y. An Efficient and Small Convolutional Neural Network for Pest Recognition—ExquisiteNet. *arXiv* **2015**, arXiv:1409.1556.
18. Li, J.; Zhou, H.; Wang, Z.; Jia, Q. Multi-Scale Detection of Stored-Grain Insects for Intelligent Monitoring. *Comput. Electron. Agric.* **2020**, *168*, 105114. [CrossRef]
19. Shi, Z.; Dang, H.; Liu, Z.; Zhou, X. Detection and Identification of Stored-Grain Insects Using Deep Learning: A More Effective Neural Network. *IEEE Access* **2020**, *8*, 163703–163714. [CrossRef]
20. 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, Venice, Italy, 22–29 October 2017; pp. 764–773.
21. Mnih, V.; Heess, N.; Graves, A. Recurrent Models of Visual Attention. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2014; Volume 27.
22. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-Local Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.

23. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Vedaldi, A. Gather-Excite: Exploiting Feature Context in Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31.
24. Li, X.; Zhong, Z.; Wu, J.; Yang, Y.; Lin, Z.; Liu, H. Expectation-Maximization Attention Networks for Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9167–9176.
25. Huang, Z.; Wang, X.; Wei, Y.; Huang, L.; Shi, H.; Liu, W.; Huang, T.S. CCNet: Criss-Cross Attention for Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019. [CrossRef]
26. 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, 18–23 June 2018; pp. 7132–7141.
27. Gao, Z.; Xie, J.; Wang, Q.; Li, P. Global Second-Order Pooling Convolutional Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3024–3033.
28. 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 (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
29. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
30. Qin, Z.; Zhang, P.; Wu, F.; Li, X. FcaNet: Frequency Channel Attention Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 783–792.
31. Guo, M.-H.; Xu, T.-X.; Liu, J.-J.; Liu, Z.-N.; Jiang, P.-T.; Mu, T.-J.; Zhang, S.-H.; Martin, R.R.; Cheng, M.-M.; Hu, S.-M. Attention Mechanisms in Computer Vision: A Survey. *Comp. Vis. Media* **2022**, *8*, 331–368. [CrossRef]
32. 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, 27–30 June 2016; pp. 770–778.
33. Li, Q.; Shen, L.; Guo, S.; Lai, Z. Wavelet Integrated CNNs for Noise-Robust Image Classification. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 7243–7252.
34. 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.
35. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning Deep Features for Discriminative Localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
36. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]

agriculture

MDPI

Review

The Path to Smart Farming: Innovations and Opportunities in Precision Agriculture

E. M. B. M. Karunathilake [1], Anh Tuan Le [1], Seong Heo [2], Yong Suk Chung [1,*] and Sheikh Mansoor [1,*]

[1] Department of Plant Resources and Environment, Jeju National University, Jeju 63243, Republic of Korea; bhagya@office.jejunu.ac.kr (E.M.B.M.K.)
[2] Department of Horticulture, Kongju National University, Yesan 32439, Republic of Korea
* Correspondence: yschung@jejunu.ac.kr (Y.S.C.); mansoorshafi21@gmail.com (S.M.)

Abstract: Precision agriculture employs cutting-edge technologies to increase agricultural productivity while reducing adverse impacts on the environment. Precision agriculture is a farming approach that uses advanced technology and data analysis to maximize crop yields, cut waste, and increase productivity. It is a potential strategy for tackling some of the major issues confronting contemporary agriculture, such as feeding a growing world population while reducing environmental effects. This review article examines some of the latest recent advances in precision agriculture, including the Internet of Things (IoT) and how to make use of big data. This review article aims to provide an overview of the recent innovations, challenges, and future prospects of precision agriculture and smart farming. It presents an analysis of the current state of precision agriculture, including the most recent innovations in technology, such as drones, sensors, and machine learning. The article also discusses some of the main challenges faced by precision agriculture, including data management, technology adoption, and cost-effectiveness.

Keywords: precision farming; smart farming; agricultural technology; Internet of Things (IoT); big data analytics; machine learning; artificial intelligence (AI)

Citation: Karunathilake, E.M.B.M.; Le, A.T.; Heo, S.; Chung, Y.S.; Mansoor, S. The Path to Smart Farming: Innovations and Opportunities in Precision Agriculture. *Agriculture* **2023**, *13*, 1593. https://doi.org/10.3390/agriculture13081593

Academic Editors: Francesco Marinello, Xiuguo Zou, Zheng Liu, Xiaochen Zhu, Wentian Zhang, Yan Qian and Yuhua Li

Received: 19 June 2023
Revised: 4 August 2023
Accepted: 9 August 2023
Published: 11 August 2023

1. Introduction

Precision agriculture (PA) is a management strategy for addressing geographical and temporal variabilities in agricultural fields [1–3] that involves data and contemporary technologies. With a forecasted human population of between 9 and 10 billion by 2050 [3–5], precision agriculture is becoming more and more important to contemporary agricultural research. By 2050, the amount of food produced worldwide must grow by at least 70% [1,5–7]. This is a difficult endeavor [4] because it puts further strain on already-scarce resources and the environment [1–3]. Therefore, precision agriculture is essential to maximize output while using fewer inputs of all sorts in more effective ways, reducing adverse impacts on the environment, and assuring sustainability [2,3]. Precision farming was born with the introduction of GPSs (global positioning systems), GISs (geographic information systems), yield monitors, and other data generators in all three crucial phases of agricultural operations in the 1990s [2,8,9]. In precision agriculture, motorized equipment was only used for performing agricultural processes [2,10], and the problem-recognizing and decision-making steps were authorized by humans. The technological advancement during the Third Industrial Revolution, known as Industry 3.0 [8], led precision agriculture to digitalization by integrating information technologies and improved automation capabilities in precision farming. As a result of this digitalization, "farm practices" with manual tools moved to "agriculture" from animal traction, then to motorized mechanization, and now to digital equipment [2].

Precision agriculture so far mainly consists of variable rate technologies (VRTs), electronic maps, yield monitors, and guidance farming systems [2,8]. Variable rate applications

were firstly demonstrated in northern Germany and Denmark in 1988 after global positioning systems (GPSs) were available for civil services [11]. GPS services were opened for general use in U.S. farms in 1983 [2]. In the next decade, GPS technology facilitated farmers to precisely locate and map their fields [10,12], empowering them to manage their farmlands according to site-specific conditions and field variabilities. At the beginning of the second millennium, yield monitors were developed, enabling farmers to monitor crop yield in real-time via best matching [13]. Advancement of remote-sensing technology, such as satellites, drones, ground-based sensors, and crews, authorized farmers to collect high-resolution data on their fields, allowing them to make informed decisions about crop management [3]. Precision agriculture is not only focused on crop farming but also on other agricultural production systems: agronomics, livestock farming, aquaculture, and agroforestry [2,3,9,14].

In the current status of precision agriculture, there are several issues, such as unsustainable resource utilization, long-term monoculture, intensive animal farming [8], environmental compromises, uneven distribution of digitization [15], food safety issues, inefficient agri-food supply chain [13,16], and lack of awareness of and inertia toward novel changes. These issues prevent achieving efficiency, productivity, and sustainability from agricultural production and escalate unintended impacts on ecosystems [17]. The fourth industrial revolution, which is known as Industry 4.0, occurred in 2011 with the Internet of Things (IoT), big data, artificial intelligence (AI), robotics, and blockchain technology [8,18]. In 2017, these advanced technologies were integrated into agriculture in order to overcome the above-mentioned issues, transforming precision agriculture to Agriculture 4.0, or smart farming [8,16]. With this transition, there is a growing focus on sustainability in agriculture, with many farmers adopting precision agricultural technologies to reduce the environmental impacts of farming and promote long-term sustainability. As a result, agricultural-manufacturing processes and supply chains have become more autonomous and intelligent [18], including the automation of various tasks such as planting, seeding, harvesting, and soil sampling. This is making farming more efficient while reducing labor costs.

Smart agriculture is an evolving field that leverages technological innovations to transform traditional farming practices. The integration of digital technologies into agriculture has opened up new opportunities and possibilities, revolutionizing the way farmers manage their crops, resources, and operations. It is a rapidly evolving field that encompasses a wide array of approaches, applications, and impacts. The broader objective of this review is to delve into the essential aspects of precision agriculture, exploring its key components and highlighting its potential for sustainable farming practices. One of the critical aspects of precision agriculture is data collection and acquisition planning, which plays a fundamental role in optimizing farm management decisions. Through efficient data gathering, farmers can make informed choices regarding crop health, resource allocation, and yield optimization. Decision making and execution are also vital components of precision agriculture, where the integration of cutting-edge technologies is pivotal. Leveraging machine vision technology, the Internet of Things (IoT), and artificial intelligence (AI) can lead to enhanced precision and efficiency in agricultural processes, benefiting both farmers and the environment. Throughout this review, successful precision agriculture proposals and real-world implementations are analyzed to gain insights into their achievements and challenges. By identifying future developments required in precision agriculture, we aim to provide a comprehensive understanding of how this field can continually evolve to support sustainable farming practices and address global food security challenges. The amalgamation of scientific research and technological innovations holds great promise for the future of precision agriculture and its positive impact on agriculture and society as a whole.

2. Precision Agriculture Approaches, Applications, and Impacts

Precision agriculture involves data-driven management decisions that improve resource use efficiency, resulting in reduced agricultural costs while lowering the environmental impacts from agriculture [19]. Hence, data and data collection systems, decision support tools, and data-driven equipment and input adjustments are major components of precision agriculture [2], engaging in three key agricultural steps: diagnosis, decision making, and performing [20], respectively. Before the integration of smart technologies, ICT (information and communication technology) was incorporated into agricultural devices and machinery to capture real data. Here, remote sensing, automated hardware and software, telematics, drones, autonomous vehicles, GPSs, and robotic technologies were incorporated into agricultural practices. As an example, the agro-tech company John Deere introduced GPSs for tractors, expecting increased yield and decreased input wastage [19]. The previous status of precision agriculture before smart farming can be summarized as follows.

2.1. Data Collection and Acquisition

Data, data collection, and decision support tools are important for the identification and diagnosis of various aspects in agriculture. In precision agriculture, data on individual fields and crops are gathered by observing, measuring, and sensing with different kinds of sensors, yield and soil monitors, and remote-sensing tools, such as imaging from drones, crews, aircraft, or satellites [1–3,13]. Thus, "sensing" is a fundamental management tool of precision agriculture [3,13], which is observing detailed information and providing data on climate conditions, soil conditions, fertilizer requirements, water availability, pest and disease stresses, and other field parameters [3]. A range of sensors are used in precision agriculture. Biomass parameters are important in making decisions to monitor the fertilization and caring for crops. Sensors for mass flow and moisture content are components of yield monitors, together with a differential global positioning system (DGPS) receiver. Properly calibrated yield monitors can generate accurate real-time information for decision making, such as underperforming areas leading to site-specific crop fertilization designs [13]. Precision livestock farming uses sensors and monitoring technology to collect data on animal health and welfare, enabling farmers to make informed decisions about feed, waste, and other inputs with improved efficiency and productivity. Colter position sensors combined with ultrasonic soil surface sensors are employed in dynamic Colter depth control systems [3].

Remote-sensing technologies, such as drones, crews, aircraft, satellites, and other ground-based sensors, are used to collect data on crops and soil conditions [2,3]. Remote sensing supports the identification of spatial patterns of signatures of plants that are coincidental with soil characteristics, as well as pest or disease stresses [11]. Imagery is one kind of remote-sensing data that can reveal ground truthing [2,3,11]. Previously, aircraft have been used not only for many farming imagery operations that generate data, but also chemical- or fertilizer-spraying activities. Moreover, satellite images have been available for farm management for many years. As an example, the US-LANDSAT satellites were available for this purpose in 1970 [2]. Unmanned aerial vehicles (UAVs) equipped with global navigation satellite system (GNSS) technology have been recently employed for mapping, gathering imagery data, land surveying, crop spraying, and livestock monitoring [2,3]. Geocoded sampling is a requisite component of precision agriculture and ground truthing when spatial images are used for decision making [11]. Real-time and cost-effective remote sensing, such as LASSIE (low-altitude stationary surveillance instrumental equipment), are crucial in precision agriculture, as it enables continuous and automatic recoding of real-time images of crops and soil with GIS reference [11]. This information can be used to make informed decisions about crop management resource allocation [3].

Sensor data and other data associated with geospatial coordinates from a global navigation satellite system (GNSS) provide information to create maps, especially yield maps and soil maps for site-specific management decisions [2,3]. Yield maps are used

to characterize field production quantitatively and qualitatively [21], which is crucial to make management decisions. Analyzing variabilities depicted on maps enables the identification of factors that influence productivity, facilitating the implementation of site-specific field management strategies [3]. Soil maps offer valuable insights into the spatial distribution of the physical and chemical properties within a given field [21], serving as indispensable decision-making tools in precision agriculture [13]. This significance stems from the fact that soil's physical and chemical characteristics, such as water availability, nutrient-holding capacity, bulk density, porosity, nutrient availability, and topography, typically exert an influence on crop yield [21]. Weather and climate trends can also be predicted using sensor data, which are important in all farming practices. Harvesting time is an affecting factor of grain loss in paddy rice farming, which is also able to be monitored with data observation [1].

2.2. Planning, Decision Making, and Execution

After creating decisions by analyzing gathered information, actions are performed according to the decisions created using data-driven equipment. Most fields are not homogenous in terms of soil and climate properties, as well as diseases [22]. Conventional agriculture did not take this into account; therefore, rigorous use of limited resources and excess use of chemicals and synthetic fertilizers resulted in unsustainable conventional agricultural practices. This also drove lots of wastage, even in the amounts of resource inputs and yield. Nonetheless, precision agriculture itself has proved that the application of technologies to manage the spatial and temporal variabilities in agricultural fields is possible to improve performance and environmental quality [9]. Variable rate technologies in precision agriculture involve applying inputs such as fertilizers, water and seeds, and crop protection chemicals (pesticides and weedicides) at varying rates, depending on the specific needs of each area of a field [23]. In this approach, residual issues of chemicals, as well as wastage of input resources, can be reduced. Also, net profit can be improved with increased crop yield and reduced input costs, as farmers can use resources according to the field requirements rather than full-coverage application in fields at uniform rates [2,24,25].

According to the identified heterogeneity of a field, amounts of water, fertilizer, herbicides, pesticides, and liming can be determined and applied. When considering the irrigation practices in precision agriculture, technology-driven, more sustainable smart irrigation systems are there to apply precise amounts of water at precise times. When soil moisture sensor data give an estimation of a required amount of water, irrigation systems can be diverted into variable rate irrigation to apply irrigation water until moisture content returns to the ideal level [26]. Most of the time, these effective and efficient water management systems are automatically controlled, increasing irrigation water use efficiency (IWUE). Monteiro et al. in 2021 described the use of satellite LANDSAT data and remote-sensing data to develop a feasible operational irrigation water model [3]. Likewise, tillage depth can be determined via matching with variabilities of soil physical properties [27]. Chemical spraying and seeding are also performed according to variable rate approaches. Previously, agricultural aircraft were used for chemical spraying, where a pilot controlled the spray [23]. In the present, aircraft are employed with an auto-adjusting ability for the application rate of chemicals based on a prescription map, whilst UAVs are also used as fertilizer spreaders [3]. Precision seeding can control sowing depth, densities, and distances effectively while saving seeds, time, and labor costs. Studies estimated that precision seeding based on variable rate technologies was 10% to 30% more efficient than conventional practices [3].

This site-specific management increases the number of correct decisions per unit area per unit time related to net benefits [9] while supporting the conservation of agricultural inputs and reducing costs together with environmental impacts [2,13,24]. Another management tool, grid sampling, also involves the division of fields into a grid and collecting data at each intersection of the grid. This approach provides representative information of the entire variation within a field [11], where such data are able to be used for site-

specific management to optimize management practices precisely [7]. For small-scale variabilities of soil and crop features, a local resource management (LRM) system was developed with computer-aided farming (CAF), which translated information into variable rate applications [11].

Thus far, humans have used digital tools to enhance diagnosis and decision making while adding automated machines for precise performing [14]. The accelerating changes of Industry 4.0 plus these digital technologies have granted the gradual automation of the diagnosis and decision-making steps, limiting human involvement to only monitoring (Figure 1) [6]. This revolution mostly targets optimal farming and variability management in order to enhance production. However, fulfilling the food demand should not rely solely on "more production". At the same time, it should be consider "less wastage" of both the inputs and outputs of agricultural production [3].

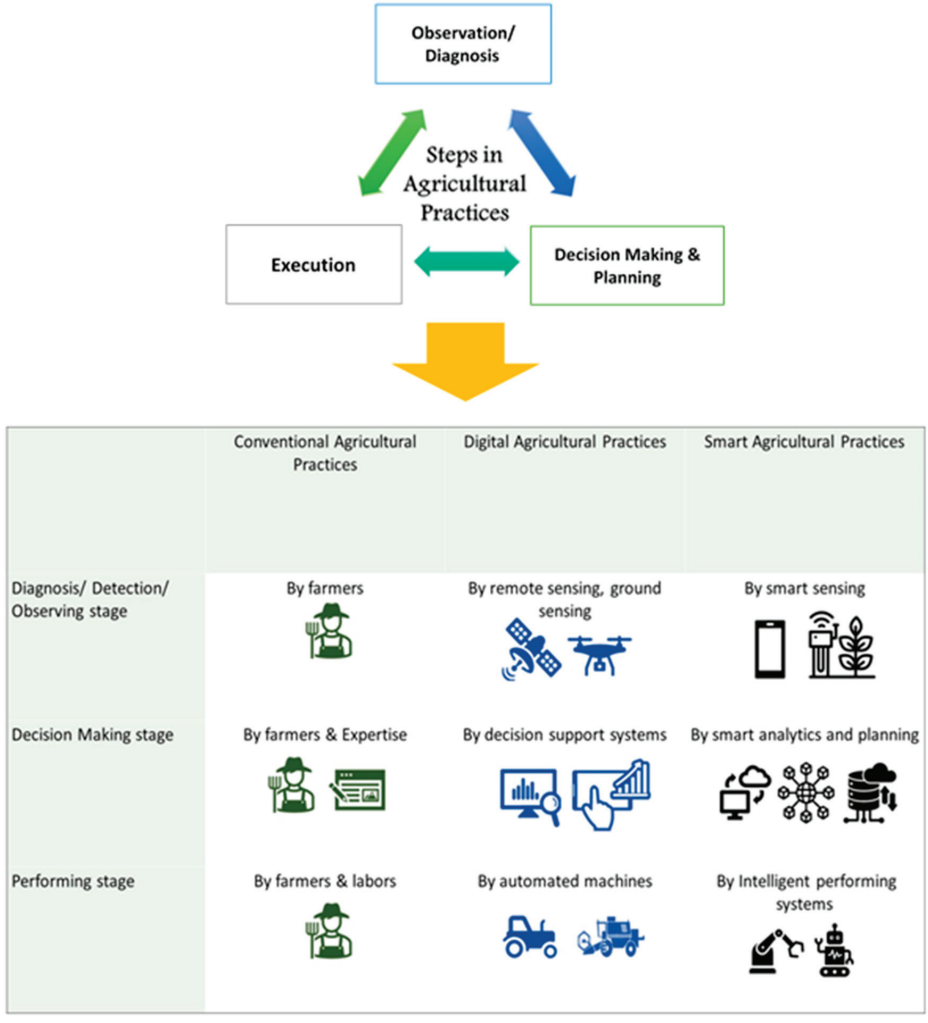


Figure 1. Three-phase cycle of an automation system and the evolution of automation of those phases in agriculture with emerging advanced technologies.

3. Precision Agriculture: The Next Frontier for Sustainable Farming

In the present, we are in the early stage of a new agricultural revolution with data-intensive approaches [2,6,16], which deploy machines at each and every step in agriculture (Figure 1), namely diagnosis, decision making, and performing. Human power is only involved in monitoring and maintaining [20]. Apart from the gradual modification of agricultural practices by the three previous industrial revolutions, the ongoing fourth industrial revolution is shaping the current status of agriculture, leading to Agriculture 4.0. This new discipline is characterized by data-driven management; new tool-based production, sustainability, professionalization; and the reduced environmental footprint of farming with modern smart technologies [24], such as robot technology (including drones), big data, artificial intelligence, computer vision, 5G, cloud computing, the Internet of Things, and blockchain technology [4,5,8,16]. This makes agricultural production systems more autonomous and intelligent [18,28]. Therefore, the following involvements can be identified as new trends and precision agriculture (Figure 2), where new capabilities are introduced to smart farming.

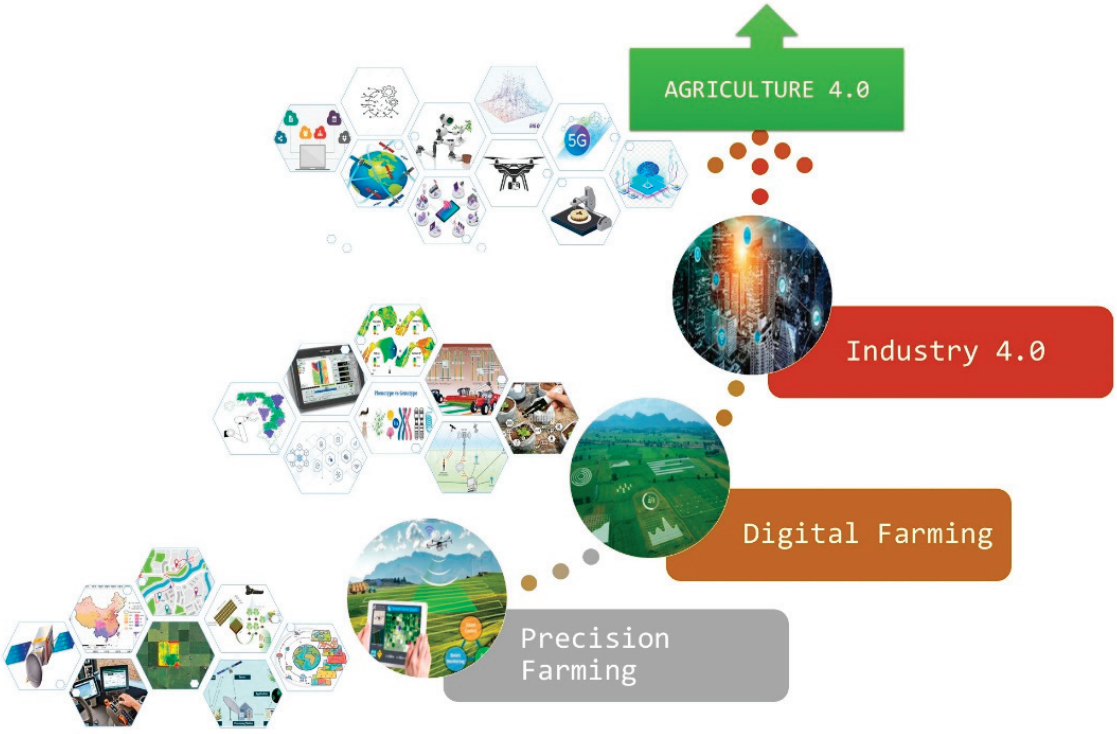


Figure 2. Different integrated technological contexts to form the fourth agricultural revolution: novel trends in precision agriculture.

3.1. Big Data

Precision agriculture systems are highly related to data and information [5]. Generally, unstructured and vast amounts of data are used by big business industries, like social-networking sites, to learn or predict customer behaviors accurately [4]. Similarly, in precision agriculture, big data analytics are applied to understand data-intensive agricultural processes for decision making [6], where analytic tools operate enormous data sets [4]. These analytic tools consist of data mining, statistics, AI, predictive analytics, neural language processing, etc. [4]. Big data science usually functions either with ML, cloud computing, image processing, modeling and simulation, statistical analysis, NDVI

vegetation indices, or GIS. These conjugations can discover correlations, patterns, and trends from large quantities of data via capturing, storing, exchanging, analyzing, and marketing features of this high-performance informatics technology [6]. These predictions and recommendations assist farmers with handling the upcoming outcomes, risks, and challenges in the agricultural industry [4]. Combining the data in agricultural production processes creates traceability of product while increasing product quality, including safety and taste. As customers are now aware of the ecological footprint of agri-products, the above combination supports the increase in the demand for agricultural commodities [29], adding high market value. Recent advancements of high-resolution remote sensing and intelligent information and communication technologies, including social media (Facebook, Twitter, Amazon, Instagram, etc.), have contributed to big data analytics in many sectors, as well as in many stages in farming, including decision making, weather forecasting, weather management, disaster management, smart management of resources, disease and pest interruption, and harvesting time predictions [4,6,30]. Moreover, big data analytics aid in implementing real-time forecasting in precision agriculture [4]. However, data updating, device security, correctness of data, accuracy of data, availability of data, and security elements, such as encryption, are still barriers when combining big data with smart farming [31]. Invalid data can lead farmers to make costly, disruptive decisions and actions [5].

3.2. Machine Vision Technology

Precise and accurate data and information are the driving components of precision agriculture. Recently, image analysis has become a more reliable data source than manual, labor-intensive, costly data-collecting methods [22,32]. Here, machines can read and understand the real world through pixel images and produce accurate site-specific information [31]. Machines with 'eyes' in agricultural activities are called machine vision (MV). This, also known as agro-vision or the 'eyes' of robots, provides non-destructive, robust, rapid, and steady methods to monitor cultivation processes. MV systems give machines their vision and judgement capabilities in image processing and data extraction [10]. Although MV technologies have already been applied successfully for crop species identification, crop stress detection, crop seed quality assessment, weed detection, disease detection, etc., they are still at the prototype stage. Currently, emerging deep-learning (DL) techniques in growing machine-learning (ML) technologies are integrated with MV applications in order to develop intelligent robots for multispectral imagery analysis and real-time analysis in field variable rate applications [10,25]. Commercial smartphones, which are ubiquitous among the human population, are able to be used in monitoring crop health and stress based on MV systems [33].

3.3. Internet of Things (IoT)

The IoT refers to a network of interconnected items and technologies [16]. The IoT is one of the most important technological advancements in precision agriculture and smart farming [5]. IoT architecture for agriculture, such as agricultural sensors with ICT and UAV, collects data for precision agriculture [31]. Also, the burgeoning IoT and mobile data are the core of the fourth industrial revolution [10]. Meanwhile, advancements in communication technologies and wireless networks (5G, LoRaWAN, NB-IoT, Sigfox, ZigBee, and Wi-Fi) have broadened the application of the IoT in diverse fields, such as real-time remote control and high-throughput phenotyping, while giving better coverage, bandwidth, connection density, and end-to-end latency (Table 1) [8]. When it consolidates in agriculture together with cloud computing, it results in smart farming [6] for various scopes of livestock monitoring, smart greenhouses, fishery management, and weather tracking [8]. The IoT can be widely used in all areas of precision agriculture with the development of sensors with independent intellectual property rights and the development of smart devices, such as intelligent tractors, UAVs, and robots that can replace high levels

of manual labor input, performing high-quality operations while adjusting to challenging working conditions [31].

Table 1. Main specifications of prominent wireless technologies of fifth-generation communication paradigm: [34–38].

	Sigfox	**LoRaWAN**	**NB-IoT**	**Zigbee**	**Wi-Fi**	**5G**
Bandwidth	Low bandwidth	Low to moderate bandwidth	Low to moderate bandwidth	Low to moderate bandwidth	High bandwidth	Very high bandwidth
Maximum Data Rate	Up to 100 bps	Up to 27 kbps	Up to 250 kbps	Up to 250 kbps	From a few Mbps to several Gbps (varies based on the version)	High data rates from several hundred Mbps to multi-Gbps
Payload Length	Limited to 12 bytes per message (140 messages per day)	Up to 51 bytes per message (varies depending on the region)	Up to 1600 bytes per message (varies depending on the network operator)	Up to 128 bytes per message (varies depending on the network layer)	Up to several kilobytes per message (varies based on the version)	Supports large payload sizes ranging from several kilobytes to several megabytes
Coverage	Several kilometers in rural areas and up to a few hundred meters in urban areas from a Sigfox base station	Varies from a few kilometers in urban area and tens of kilometers in rural areas depending on antenna height and line of sight	Wide area of coverage up to several kilometers or more from a base station by leveraging existing cellular infrastructure (similar to 2G/3G cellular networks)	Up to tens of meters (can be extended by utilizing mesh networking, allowing devices)	Limited to indoor around 30–50 m or local area environments (can be extended)	A few hundred meters to several kilometers from a base station (varies depending on the frequency band and deployment strategy)
Cost	Relatively low cost due to its simple infrastructure requirements	Cost-effective due to shared infrastructure and low-power devices	Affordable due to utilizing existing cellular infrastructure	Reasonably priced, especially for small-scale deployments	Cost-effective for local area networks, but infrastructure costs can vary	Higher infrastructure costs compared to other technologies
Advantages	Low power consumption, long-range coverage, low-cost infrastructure	Long-range coverage, low power consumption, low-cost infrastructure	Wide network coverage, secure, supports voice and mobility	Low power consumption, mesh networking, supports large networks	High bandwidth, widespread availability, support for various applications	Very high bandwidth, ultra-low latency, massive device connectivity, high reliability
Disadvantages	Limited bandwidth, low data rate	Limited bandwidth, shared spectrum, higher latency	Higher power consumption compared to other LPWAN technologies	Limited range, interference from other devices, complex network setup	High power consumption, shorter range, limited scalability	Higher infrastructure cost, limited coverage in some areas, higher power consumption

Different IoT sensors for temperature, humidity, light intensity, pressure, CO_2 levels, insect infestations, foliage, sunlight intensities, and wind speed are there to collect and receive data, which are then uploaded to cloud information support systems to man-

age [4,13,16,28]. Those sensors can directly combine with agricultural robots, autonomous platforms, machines, and weather stations for real-time monitoring [4]. With the IoT, UAVs can respond promptly, leading to high-quality, high-resolution, and exceptionally reliable observations through high-throughput 3D monitoring at different geographical areas. At the same time, various kinds of agricultural sensor nodes, autonomous farm vehicles, and mobile crowd sensing have been put forward based on the IoT for ground and undersurface cognition [8]. Most IoT sensors in precision agriculture are in wireless frameworks [13] or low-power wide-area networks [8] and, hence, can be used for on-site analysis [3], as well as mass data transfer, without any interruptions [29,31]. Still, there are cost, operational, technical, and data management difficulties in implementing the IoT in agricultural operations [13]. Designing low-cost, energy-efficient, wireless IoT technologies in autonomous applications is affected by the following dependencies: data latency on power consumption, data scalability on storage and processing cost, and data interoperability on cloud compatibility to store and process various kinds of data [13].

Different IoT devices are coalesced as networks to achieve high-speed data exchanging [4,30]. Therefore, the development of an IoT framework can also solve problems with big data [31]. With more advancements, agricultural operations like protecting, controlling, monitoring, and detecting can be extended using smart phones with the IoT [25]. As an example, time-consuming cattle status monitoring has also benefited from the IoT, allowing farmers to monitor the health and welfare of animals. Also, weed detection through MV primarily consists of deep learning (DL) and image processing [16].

Edge computing enables affordable real-time data transmission in IoT precision agriculture, reducing data package size and alleviating strain on centralized cloud resources. Internet and communication companies leverage their expertise to extend cloud service capabilities to edge networks, shaping the edge computing landscape. Pioneers like Cisco and Huawei have developed comprehensive frameworks and lightweight computing systems. The IoT connects objects through smart technologies, while research explores aerial edge–IoT systems for improved convergence speed and task completion rates [39–42].

3.4. Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL)

AI has a key role in robotics and autonomous systems (RASs). The development of AI in the IoT has contributed continuous data streams [31]. To make agricultural data into meaningful information in decision-making data, mining techniques are required. Various environmental data and farming historical records in big data are analyzed using AI, which finds patterns that are hidden in big data [29]. These discoveries are important in the pest identification, disease detection, yield prediction, and fertilizing plans [25,31] included in agricultural decision support systems. AI has noteworthy potential to accommodate the reduction of food wastage, the improvement of production hygiene, and the monitoring of machines in many stages of agriculture, such as supply chain, agricultural production pattern, and agricultural production process including soil, crop, and water management, as well as disease and pest control [4,8]. Then, AI has the potential to overcome problems in conventional farming [31].

Both ML and DL are subconcepts of AI (Figure 3) [10]. With ML, a computer learns independently to improve the performance of AI, which goes through explicit feature extraction [6]. ML focuses on the theory, performance, and properties of learning systems and algorithms, as it is a high-performance informatics technology for quantifying and understanding data-intensive farming processes [6]. On the other hand, DL can solve problems with combinations of layers and nonlinear functions [10]. To address limitations in the practical implementation of robots, mobile terminals, and intelligent devices in modern agriculture, the integration of machine-learning algorithms has had significant improvement. With machine-learning models, integration into mobile detection algorithms has paved the way for innovative and more precise detection methods, overcoming certain limitations faced by technology adaptation in plant factories, such as limited computer power, insufficient storage capacity, complexities within the plant factory environment, and

precision issues related to small target detection [25,43]. Furthermore, machine-learning techniques can mitigate the need for large network sizes and improve the operational speeds of these systems [43]. This advancement has wide-ranging applications, including accurate fruit and pest detection, as well as the optimization and prediction of complex conditions in plant tissue cultures and breeding processes [25,28,44]. Notably, a study (referenced as study 13) successfully applied machine-learning models and artificial neural multilayer regression models to enhance the in vitro regeneration of soybeans by tracking simple, observable traits, such as shoot regeneration frequency and shoot length.

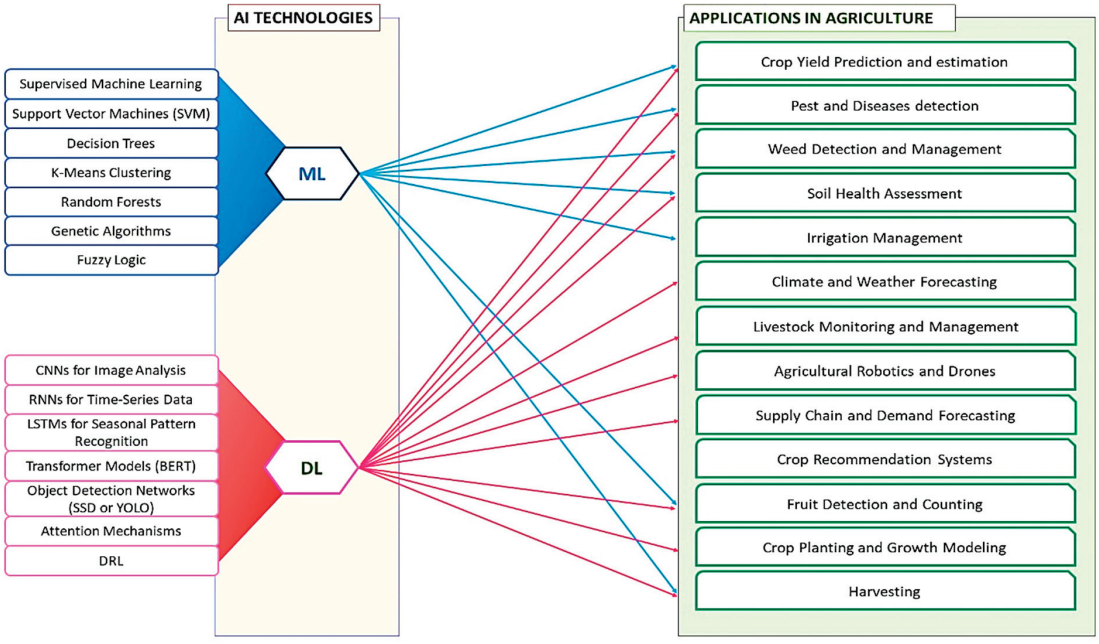


Figure 3. Major AI applications in different practices in precision agriculture.

Also, machine-learning algorithms are employed for data validation, enabling a deeper understanding of dynamic agricultural conditions through data collected from various elements of modern agriculture [6,44]. Despite these advancements, challenges remain in terms of processing speed and the development of efficient information visualization systems for farmers when dealing with big data [6]. Nonetheless, continued research in the fields of big data, the IoT, machine learning, and deep learning holds great potential in overcoming these roadblocks and providing accurate predictions of the dynamic nature of agriculture while identifying new opportunities [1]. Supervised machine-learning techniques, such as support vector machines, decision trees, k-means, random forests, genetic algorithms, deep learning, and fuzzy logic, are several categories of machine-learning models (Figure 3) that play a vital role in agricultural automation, augmenting the intelligence of other technologies, such as smartphones, unmanned aerial vehicles, unmanned ground vehicles, satellite systems, automated machines, agricultural robots, and big data analytics [1,28,31].

Mobile applications have significantly diverted from these AI, ML, DL, and MV technologies [10]. ML algorithms in big data are also critically essential because this integration can learn from data to create decisions, data-based prospects, and predictions. Due to the intricate input data requirements of machine learning (ML) and deep learning (DL), the initial stage of adopting ML models in precision agriculture may encounter significant obstacles in terms of the time and cost involved in gathering the necessary data

from commercial farms [1,6,30]. However, with the continuous advancement of IoT sensors, AI-based autonomous machines or robots together with cloud computing, edge computing, and blockchain can support overcoming this difficulty during the transforming, storing, and processing of data in the creation of ML models [27,36,41]. Accordingly, ML is able to be used to solve diverse issues in agriculture related to yield prediction, crop quality, disease detection, weed detection, species identification, animal welfare, livestock production, water management, and soil management [6,45]. Common principles of ML techniques are clustering, decision trees, instance-based models, regression, artificial and deep neural networks, ensemble learning, support vector machines, and Bayesian models [6]. A study proved that ML was a powerful tool for analyzing data to monitor inputs and outputs aiming to optimize plant tissue culture protocols [44].

Smart farming is technology that relies on its implementation with the use of AI and the IoT in cyber-physical farm management [28]. According to current applications, AI has been involved in soil management, crop management, disease management, weed control, etc. Examples are the fuzzy-logic-based soil risk characterization decision support system (SRCDSS), management-oriented modeling (MOM), artificial neural networks (ANNs), CALEX, PROLOG, computer vision systems, ANN-GIS, invasive weed optimization (IWO), and support vector machines [4]. One key application of AI is a mobile expert system where farmers can use their smartphones for disease diagnosis, species identification, and soil health analysis with the help of mobile apps. In addition, AI is a real-time analyzer of satellite images when the progress of farming is tracked with satellite imagery [24]. With AI applications, precision agriculture now has a scientific background, which helps to make precision agriculture more formalized to perform optimal agriculture outputs [29]. In the future, AI may be improved to deal with the dynamic nature of agricultural microclimates, as it is now facing difficulties finding a single standard solution for that heterogeneity. The existing experience gap between AI researchers and farmers hinders the complete understanding of agricultural problems and solutions. To eliminate this obstacle, the knowledge of farmers, agricultural professionals, and AI researchers should be linked. In spite of this, accessibility and privacy protection problems when working with huge amounts of data should be addressed to deliver more skillful AI [8,16].

3.5. Guidance Systems

Guidance systems use GPS (global positioning system) technology to provide farmers with real-time information about their equipment locations and herd-grazing locations, enabling them to optimize field operations such as planting, harvesting, and herding [1,12]. The limited number of satellites, poor signal strengths, and lack of reliable connectivity were overcome by introducing a GNSS (global navigation satellite system), which then replaced labor-intensive, time-consuming farm operations with more effective methods, such as VRA [11,31]. Previously, agricultural inputs were performed manually, and during Agriculture 3.0, they were performed mechanically using digitalized machines [2]. With rapid commercialization, agricultural machinery services have emerged that require efficient management to prevent overuse or underuse issues. For the understanding of agricultural machinery, GNSS plays a crucial role in optimizing effectivity and efficiency [46]. The new trend of GNSS-enabled devices in the fully automated steering of tractions is saving time, labor costs, and money [2]. Precision agricultural robots require high-resolution navigation solutions [47]. Similarly, agricultural rovers and robots are effective only when precisely guided in their actions [45]. Some studies introduced DL propagation models in GNSS fused with inertial navigation data sets for precision agriculture [47]. One example is electric seeders with optical fiber detection technology that were developed and tested successfully [3]. The new development of software-based farm management solutions for GIS encourage the automation of data collection and analysis of supervising, storing, decision making, and farm management.

3.6. Blockchain Technology

Blockchain is defined as a decentralized, distributed database that maintains a continuously growing list of ordered records or blocks, which was first used in cryptocurrency [15,48]. Blockchain offers data transparency, immutability, and reliability, which improve the mutual trust between various parties in the supply chain [15]. As this technology eliminates the obstacles of corporations, this was introduced to precision agriculture, increasing the easiness of the integration of digital technologies into agriculture. This step provides solutions to some technical challenges in smart farming, furnishing the remote monitoring and controlling of farm equipment through the "IoT applied Greenhouse Monitoring System" [15,48]. One such challenge is an insufficient and insecure infrastructure for data sharing. Another challenge is the delay of remote-sensing satellites in detecting the variability of croplands. Therefore, as a solution for the above decentralization, anonymity, and security problems in the IoT in smart farming, blockchain has been proposed, expecting lightweight, distributed, decentralized, and transparent security and privacy [5,48]. Blockchain can assist with having a reliable, faster, and secure platform to monitor farm operations, although it is still in its early stages of maturity [15,48]. As information can be communicated securely in a distributed network [48], with the help of blockchain this can improve the planning of schedules for various agricultural processes, such as irrigation water sharing, energy consumption, the incorporation of machines and labors, and tasks for robot coalitions and autonomous UAVs [15,28]. Especially in the food supply chain, this is a crucial point because of food safety issues, as well as asymmetric and fragmented information occurring related to the insufficient supply chain [1,8,10].

3.7. Robotics and Autonomous Systems

Most recently, autonomous farming has involved a high degree of the use of robotics, sensors, drones, and remote sensing to perform various agricultural tasks, such as planting, spraying, harvesting, and weeding, while reducing labor costs and improving efficient decision making [3,45]. RASs are a combination of emerging modern technologies that have key applications in both agricultural production processes and production patterns. Mobile robots equipped with various sensors, actuators, and ML algorithms are key enablers to automatically handle variability and uncertainty in farming practices [47]. Key applications of RAS in agricultural patterns are in plant factories, 3D food printing, and biodiverse farming, whereas autonomous farming, aerial monitoring, and automated husbandry have become new applications in agricultural production processes [8]. However, agricultural RASs are required to be improved to fulfill efficient work with accurate guidance, autonomous navigation, and accurate detection of dynamic agricultural environments (changing appearances, growth stages, weather conditions, object overlapping, etc.). Intelligent actions, such as robot-assisted plant phenotyping, fruit counting, fruit harvesting, fruit counting, leaf peeling, selective spraying, and 3D mapping, are demonstrated and currently employed applications of RASs [8]. Auto-steered agricultural vehicles are also used in many field operations [3], such as tilling, planting, chemical applications, and harvesting. These machines, like harvesters, sprayers, tractors, planters, and mechanical weed controls, use guidance systems either with light bars [13] or a GNSS [2,20]. These guidance systems visualize the positions of equipment to prevent skips and overlaps, which is important in variable rate applications.

3.8. Artificial Satellites, Unmanned Aerial Vehicles (UAVs), and Unmanned Ground Vehicles (UGVs)

Artificial satellites, such as American Landsat satellites, the European Sentinel-2 System, the RapidEye constellation satellite system, the GeoEye-1 system, and WorldView-3, for remote sensing help to generate remotely accessible data in multispectral forms [8]. The establishment of these intelligent remote-sensing satellites has provided full coverage for collecting agricultural information [8,31,49]. More recently, ubiquitous and affordable technologies such as drones, crews, and aircraft have allowed images to be captured closer

to the ground and at a higher frequency, increasing detail and functionality [45]. UGVs acquire high-resolution data for weed identification and control, selective pesticide spraying, soil analysis, and crop scouting, while scouting robots accomplish specific targets [49] such as mechanical weeding (Oz robot), spraying (GUSS autonomous sprayer), fertilizing, mapping, and seeding (RowBot system), as well as vineyard management (VineRobots) [4,50]. Information, including imagery data generated by satellites, UAVs, and UGVs, is the paramount thing in precision agriculture, as it supports vegetation patch identification, weed recognition, pest attack detection, observation of environmental stresses, and accurate classification in VRT [18,45]. Not only that, in other agricultural disciplines, such as aquaculture, agroforestry, and forestry, imagery data play a considerable role because they can cover large areas when gathering information, and these data are reproducible [20]. Data from satellites, UAVs, and UGVs are supported by detailed ground survey data processed with ML and DL algorithms in order to make them usable and meaningful information [18].

For example, in forestry, determining forest densities is labor-intensive and time-consuming, although it is an important parameter when combatting climate change. Recently, data of tree type distribution could be achieved over a wide area of forest with the help of hyperspectral images and NDVI and RGB images from UAVs such as Sentinel-2 [13,16]. Likewise, in remote sensing satellites and drones play a big role in monitoring deforestation and obtaining accurate coverage of vegetative types and classification of tree species and are more effective than other UAV or LiDAR data [14,34]. Although there are limitations, drone and remotely piloted aircraft usage is dramatically increasing while providing precise information for precision agriculture through hyperspectral sensors, multispectral cameras, and other novel technologies [14]. This is a cost-effective, promising method for monitoring large-scale farms or crop lands [4], as well as forest areas [14].

3.9. High-throughput Phenotyping

High-throughput phenotyping has emerged as a promising approach to enhance precision agriculture by allowing the rapid and accurate measurement of plant traits [51] quantitatively and qualitatively [22,52]. Accurate and high-throughput plant phenotyping is important for accelerating crop breeding [52]. This technique uses advanced technologies such as remote sensing [40,42], spectral imaging [41], and robotics [53] to collect large amounts of data on plant characteristics, such as growth rate, yield, disease resistance, and morphology [51,54,55]. By collecting and analyzing these data, farmers can gain insights into how their crops are performing and make more informed decisions about things like fertilization, irrigation, harvesting, and pest management [22,54]. High-throughput phenotyping can also help breeders to develop new crop varieties that are better adapted to local growing conditions and can produce higher yields (Figure 4) [51]. A full range of visible and near-infrared hyperspectral data enables ML techniques such as LSR (least squares regression) to predict specific biochemical and physicochemical traits beyond simple vegetative indices [56]. ML-based precision agriculture systems have AI background [52,54], and therefore, when detecting diseases, pests, nutrient deficiency, and weeds, stressed responses are detected using high-quality images generated with UGV or UAV remote sensing, hyperspectral imaging, and satellite imaging to support high-throughput phenotyping [57]. Ultimately, high-throughput phenotyping has the potential to revolutionize agriculture by enabling more the precise, real-time, and efficient monitoring of farming practices that can improve crop productivity, reduce environmental impacts, and increase food security [54].

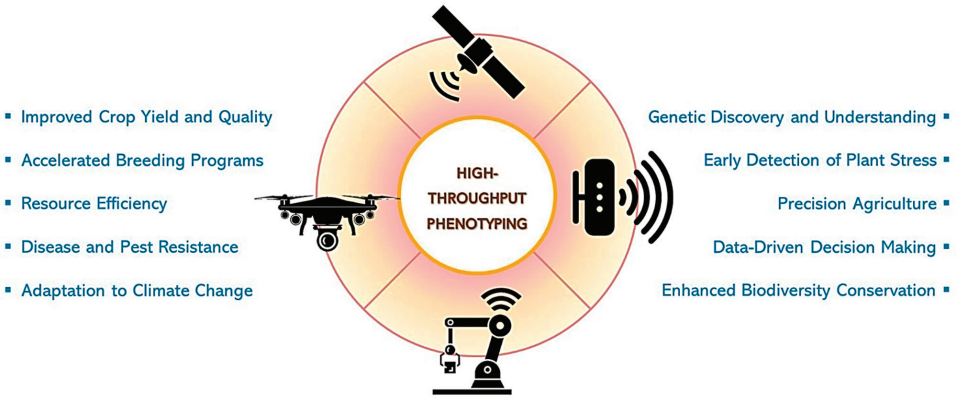


Figure 4. Importance of high-throughput phenotyping in agriculture.

The traditional methods of plant breeding have limitations in terms of time, cost, and accuracy. HTP, on the other hand, uses nondestructive and rapid methods to gather data on a large number of plants, allowing breeders to identify traits of interest more efficiently [58]. A study by Yang et al. (2017) [59] described using high-throughput phenotyping (HTP) and quantitative trait locus (QTL) mapping to investigate the genetic architecture of maize plant growth. The authors collected data on various traits related to plant growth, such as plant height, leaf area, and biomass, using HTP techniques, such as imaging and spectroscopy [59]. Unmanned aerial systems (UASs) have brought about a revolutionary change in field high-throughput phenotyping by providing a platform for different sensors to collect remote-sensing data in field-scale trials. These sensors include regular RGB cameras, multispectral-imaging cameras, hyperspectral-imaging cameras, thermal-imaging sensors, and light detection and ranging (LiDAR) sensors that enable the nondestructive estimation of plant traits, such as yield, biomass, height, and leaf area index. This is a significant advancement in agriculture, allowing for the high-throughput phenotyping of crops. In comparison to ground-based sensors, UASs increase the frequency and throughput for phenotyping, while being cost-effective and providing high-resolution images as compared to satellite-based techniques. The phenotypic traits can be used to select crops with high yield and strong stress resistance, such as disease and salt resistance, ultimately leading to improved production [60].

As technology advances, the future of high-throughput phenotyping (HTP) appears promising. Multiple HTP technologies, such as drones, sensors, and artificial intelligence, can be integrated to facilitate more efficient and accurate phenotyping, which can aid breeders in identifying desirable traits and making better selections. HTP can also be used for precision agriculture, where farmers can leverage data generated using HTP technologies to make informed decisions on inputs such as fertilizers, pesticides, and water to increase efficiency, reduce waste, and improve yield. HTP can also play a crucial role in climate change research by identifying crop varieties that can better adapt to changing climate conditions, thereby ensuring food security. Lastly, HTP can be used in developing countries to enhance food security and improve crop productivity, but it requires the development of affordable and accessible HTP technologies that can be easily adopted by farmers in those countries [56,58,59].

3.10. Telematics

Broadband connectivity is required when addressing challenges in the adoption, cost, and environment of smart technologies. Inadequate connectivity leads to inefficiencies, impacting machine downtime, human error, and real-time information availability. Limited connectivity not only affects profitability but also hampers the adoption of real-time-reliant precision agriculture. Producers with adequate connectivity are expected to be more effi-

cient, highlighting the importance of connectivity in agriculture [61]. The transformative potential of 5G and beyond mobile networks in driving business and societal change is being recognized. Considering environmental concerns and climate change, the role of mobile networks in fostering sustainability and innovation is questioned. Sectors like smart agriculture, forestry, biodiversity monitoring, and water management are crucial for sustainable resource utilization. Evaluating the capabilities of 5G and 6G networks, including current and future support, is essential for identifying use cases and the requirements in these domains [34]. As an example, a study in Thailand designed telematics-equipped tractors to assist farmers in efficiently managing their machinery, optimizing performance and enhancing overall productivity. In addition to improved management capabilities, these tractors offered features such as theft prevention, effective maintenance monitoring, and machine operation tracking [62].

4. Studies of Successful Precision Agriculture Proposals and Implementations

The article [63] reviewed advancements in automated fruit-harvesting robots for sweet peppers and apples, highlighting the successful implementation of a sweet-pepper-harvesting robot called 'Harvey', which effectively addressed detection, grasp selection, and manipulation challenges. Similarly, the apple-harvesting robot utilized a picking manipulator and a catching manipulator, along with machine vision and prioritization algorithms, for efficient harvesting. The article emphasized interdisciplinary collaboration for further advancements in automated harvesting systems and the importance of intelligent systems like deep learning and crop management software for enhancing productivity and sustainability in modern agriculture. Field trials were conducted with Harvey for sweet peppers in Australia, while robotic picking systems for apples were tested in a Washington orchard in the U.S. [64,65]. Israel has successfully implemented autonomous robotic technology in their crop fields, paving the way for the commercial use of AI harvesters. Tevel Aerobotics Technologies developed an autonomous fruit-picking system that utilized flying robots tethered to an autonomous vehicle, enabling accurate fruit picking, extended work hours, and additional tasks like tree thinning and pruning. This system addressed labor shortages, reduced fruit production costs by approximately 30%, provided real-time updates to farmers via a mobile app, and aimed to tackle challenges faced by the agriculture industry. Tevel plans to introduce its innovative solution to the global market, catering to fruit farmers worldwide and contributing to the growing agricultural robotics sector [66].

Senapathy et al. introduced the IoTSNA-CR model from their study, which leveraged IoT technology to classify soil nutrients and provide crop recommendations, aiming to optimize fertilizer usage and maximize productivity for farmers. The implementation of AI harvesters in Israel showcases the potential of artificial intelligence, machine learning, cloud services, sensors, and automation for delivering real-time information and support to farmers. The proposed IoTSNA-CR model incorporated IoT sensors, cloud storage, machine-learning techniques, and an optimized algorithm (MSVM-DAG-FFO) to achieve high accuracy in soil analysis. The model allowed farmers to maintain soil information in the cloud, reducing costs and improving productivity. Experimental validation confirmed the effectiveness of the model for crop prediction and soil health maintenance, emphasizing the importance of real-time data collection and expanding data sets and regular application use for informed decision making and soil quality enhancement [49]. The use of unmanned aerial systems (UASs) and unmanned ground vehicles (UGVs) in precision agriculture for inspecting insect traps in olive groves was proposed by [49], with a cooperative robot architecture using UAS and UGV systems evaluating vision-based navigation algorithms and augmented reality tags for return and landing. The results demonstrated the feasibility of the architecture for automating inspections and improving pest control policies. Challenges remain in addressing real-world conditions and optimizing image capture. Future work includes real-world scenarios and long-term mission capabilities of UAS vehicles [49].

Two studies from University Tenaga National, Malaysia, present autonomous and robotic machineries to deal with fertilizer and pesticide spraying. The authors of [67]

presented a low-cost agricultural robot for fertilizer and pesticide spraying, crop monitoring, and pest detection. The prototype system operated autonomously, reducing labor costs, although productivity slightly lagged behind human workers. An autonomous organic fertilizer mixer was developed in [68] based on IoT technology to reduce labor costs and enhance efficiency. The improved mixer allowed remote monitoring, updates, and alerts, aiming to further streamline the organic fertilizer-mixing process. A harvesting robot system for cherry tomatoes in greenhouses was developed by the Beijing Research Center of Intelligent Equipment for Agriculture. This new harvesting robot system for cherry tomatoes was designed featuring a railed-type vehicle, a visual servo unit, a manipulator, and picking end-effectors. Field tests demonstrated an average picking time of 12 s per bunch of tomatoes with a success rate of 83% [69]. Also, X. Jin et al. [70] designed a small-sized vegetable seed electric seeder with power drive and optical fiber detection technology, providing high efficiency and precision by monitoring sowing conditions in real-time for different seed sizes (Table 2).

The Cooperative Heterogeneous Robots for Autonomous Insects Trap Monitoring System experiment in Portugal proposed a cooperative UAS and UGV system for olive grove inspection that verified the feasibility and robustness of the multiple-cooperative robot architecture in an olive inspection scenario [49]. Russian researchers Filipe et al. [7] proposed an approach for dynamic robot coalition that combined fuzzy coalition games and smart contracts to form a dynamic and trusted coalition. It enabled the collection and dissemination of information from robot sensors in a shared space. Integration of the IoT with blockchain allows the continuous tracking of food in precision agriculture tasks, ensuring transparency and verification at each stage. Precision agriculture is a strategy that uses advanced technologies, like sensors, remote sensing, and data analytics, to improve agricultural management decisions and increase productivity, profitability, and sustainability. Machine-learning models have been integrated with IoT sensors to develop intelligent sensors for generating of big amount of data. In the study of Smolka et al. [71], a microchip capillary electrophoresis sensor was used for soil nutrient analysis, demonstrating its general sensitivity to ions in liquids, particularly NO3, NH4, K, and PO4. The sensor exhibited strong linearity and detected important plant nutrients, which could contribute to future developments in digital agriculture. Insufficient power infrastructure is one obstacle in adapting novel technologies in agricultural fields. Researchers successfully developed an IoT-based solar-energy-powered smart farm irrigation system in the United Arab Emirates that harvested renewable energy for smart farm irrigation [72]. This study outcome paved the way to developing three operation modes that are available for farmers' use.

VRT is a major constituent in precision agriculture that deploys field maps, GPSs, and GNSSs to establish the precision of input applications. A study of a data fusion method for yield and soil sensor maps [21] evaluated fusion results on fields, highlighting their usefulness in decision support for drainage, irrigation, and variable yield goals. It uncovered hidden areas of lost yield potential using soil sensing, EC, pH, organic matter, and topography data fusion. Researchers in Beijing, China, developed a new method using image segmentation and pixel-level visual features to accurately classify field and road areas in GNSS recordings of agricultural machinery, surpassing existing methods and demonstrating a superior performance for high-frequency GNSS trajectories [46]. A multisensor data fusion approach was used by Whattoff et al. for creating variable depth tillage zones [27]. Variable depth tillage (VDT) reduced costs, labor, and fuel consumption. A multisensor data fusion approach was developed to map soil properties for VDT implementation, showing the depth of tillage needed in different areas. This approach proved useful in guiding VDT operations for efficient soil management.

One study in Germany integrated computer-aided farming, an IoT-based pH sensor, and VRT for effective VR liming, and the lime requirement was successfully determined in situ by establishing a buffer curve [11]. A field evaluation of a VR aerial application was conducted in the study of Martin and Yang [23] utilizing prescription maps for aerial glyphosate applications with variable rate nozzles. Accurate spray deposition within 20 feet

of the target was confirmed using multispectral imagery, boosting confidence in variable rate application and encouraging adoption. Italian authors Corbari et al. [73] explored the integration of a satellite-driven soil–water balance model and meteorological forecasts to enable precision smart irrigation. It discussed model performance and emphasized the importance of using consistent data for the calibration and validation of soil hydrological parameters [73]. The short communication of Jang et al. [22], "Spatial Dependence Analysis as a Tool to Detect the Hidden Heterogeneity in a Kenaf Field", presented high-throughput phenotyping as having potential in precision agriculture. This study demonstrated its application for revealing field heterogeneity and suggested its use for better analysis and management in plant breeding and precision agriculture. In [57], Kim et al. emphasized the importance of evaluating drought effects during the vegetative stages of soybean, indicating the potential of using phenotypic traits as selection indicators for breeding drought-resistant soybean cultivars, especially considering the escalating crop damage caused by drought and global warming.

Another study asserted precision agricultural applications in agroforestry. Tree species identification and classification is important when combatting climate change, as well as monitoring ecosystem health 17. Researchers used images from SENTINEL-2 to propose methods to determine tree type distribution in a wide forest area using UAV images [14,17]. They effectively distinguished evergreen, deciduous trees, and grassland areas, aiding in forest planning and preparing for climate change impacts. Ma et al. [14] used a random forest classifier with satellite images to improve texture feature separation among tree species. The overall classification achieved 86.49% accuracy and a 0.83 Kappa coefficient, although altitude, slope, and aspect influenced tree distribution. These outcomes were important in species classification and biodiversity monitoring, as well as in informing inventory estimation [14].

An evaluation of soybean wildfire prediction via hyperspectral transmission imaging was performed with Python, which detected bacterial wildfire in soybean leaves where different varieties exhibited distinct spectral signatures. This allowed the precise detection and differentiation of healthy and diseased plants effectively with high accuracy (97.19% and 95.69%) in early disease detection, confirming its usefulness in soybean plant monitoring [32].

Aasim et al. [44] focused on establishing the efficient and reproducible in vitro regeneration of common beans through a combined approach of in vitro regeneration and machine-learning algorithms. ML models, particularly ANN algorithms, were used for prediction and optimization. The ML and ANN models demonstrated superior performances, proving their efficacy in analyzing and optimizing complex conditions in plant tissue culture protocols for breeding purposes.

A computer vision and deep-learning-enabled weed detection model for precision agriculture was proposed in [25] integrating computer vision, DL, the IoT and a smartphone. The proposed CVDL-WDC technique combined multiscale object detection and ELM-based weed classification. The results showed improved outcomes over recent approaches, and future extensions included integration with IoT and smartphones.

At the same time, a novel procedure involving machine learning and UAV-based imagery was developed to accurately identify crops and weeds, offering potential integration into autonomous weed management systems and contributing to improved precision agriculture practices with reduced resource consumption [45].

At Sairam Institute of Technology in India, a flood detection system based on the IoT, big data, and a convolutional deep neural network (CDNN) was developed [30]. The CDNN algorithm demonstrated superior accuracy, achieving an impressive accuracy of 93.23%, a sensitivity of 91.43%, a specificity of 91.56%, a precision of 92.23%, a recall of 90.36%, and an F-score of 91.28% with a data set of 500. The flood detection system outperformed existing methods and holds potential for further enhancement through the integration of IoT devices and advanced algorithms, ensuring improved flood detection capabilities.

In order to alleviate the strain on agri-food production, the introduction of alternative nutrient sources can be explored, particularly through the utilization of cultured meat and 3D-printed meat as substitutes for traditional animal meats, thus reducing the demand on animal husbandry. In China, the production of lab-grown meat using muscle stem cells necessitated edible 3D scaffolds created through electrohydrodynamic (EHD) printing, showcasing the significant potential of prolamin scaffolds for cultivating cultured meat [74]. Similarly, the construction of 3D-printed meat analogs from plant-based proteins has been conducted, improving the printing performance of soy protein- and gluten-based pastes facilitated by rice protein. This study examined the rheological properties and printing performances of edible inks made from soy protein isolate (SPI), wheat gluten (WG), and rice protein (RP). Increasing the proportion of rice protein improved the 3D-printing performance, holding potential for the 3D printing of plant-based foods and constructing meat analogs simulating real meat properties [75].

Several studies have shown why the adaption rate of these studies is slow, and one case study conducted in Chumphon Province, Thailand, by Kasetsart University examined the adoption of smart farming technology among durian farmers, highlighting that factors such as age, occupation, access to extension services, and farm size influenced technology adoption, with younger farmers having larger farms being more inclined to adopt technology, resulting in decreased labor and fertilizer expenses, which emphasized the importance of providing continuous training and promoting extension services for sustainable adoption [76].

Table 2. Studies of successful precision agriculture proposals and implementations.

Exploration	Location	Technology Used	References
Usage of Smart Contracts with FCG for Dynamic Robot Coalition Formation in Precision Farming	St. Petersburg, Russia	IoT, agricultural robotics, blockchain technology with hyperledger fabric platform	[7]
A mobile lab-on-a-chip device for on-site soil nutrient analysis	Vienna University of Technology, Vienna, Austria	Micro-chip capillary electrophoresis sensor device	[71]
Development and test of an electric precision seeder for small-sized vegetable seeds	Henan University of Science and Technology, Luoyang, China	Optical fiber detection technology	[70]
Smart irrigation forecast using satellite LANDSAT data and meteo-hydrological modeling	Politecnico di Milano, Milan, Italy	IoT sensors	[73]
IoT solar-energy-powered smart farm irrigation system	American University of Sharjah, Sharjah, United Arab Emirates	Chip controller with built-in WiFi connectivity, IoT	[77]
Autonomous fertilizer mixer through the Internet of Things (IoT)	University Tenaga Nasional, Selangor Darul Ehsan, Malaysia	IoT	[68]
Design and development of a robot for spraying fertilizers and pesticides for agriculture	University Tenaga Nasional, Selangor Darul Ehsan, Malaysia	Agricultural robots	[67]
25 years of Precision Agriculture in Germany—A retrospective	Federal Research Institute for Cultivated Plants, Bundesallee, Braunschweig	Computer-aided farming, IoT-based pH sensor, VRT	[11]
Field Evaluation of a Variable Rate Aerial Application System	United States Department of Agriculture, Texas, USA	UAVs, VRT, high-resolution camera	[23]
A harvesting robot system for cherry tomatoes in greenhouses	Beijing Research Center of Intelligent Equipment for Agriculture, Beijing, China	Agricultural robots	[69]
Characterization of Tree Composition using Images from SENTINEL-2: A Case Study with Semiyang oreum	Republic of Korea	SENTINEL-2 satellite, image analysis, remote sensing,	[17]

Table 2. *Cont.*

Exploration	Location	Technology Used	References
Innovation in the Breeding of Common Beans Through a Combined Approach of in vitro Regeneration and Machine-Learning Algorithm Citation	Sivas, Turkey	ML and ANN models	[44]
3D-Printed Prolamin Scaffolds for Cell-Based Meat Cultures	Suzhou, Jiansu, China	3D-printing technology, high-precision microstructures for biomedical applications	[74]
Construction of 3D-printed meat analogs from plant-based proteins: Improving the printing performance of soy protein- and gluten-based pastes facilitated by rice protein	Nanchang, China	3D-printing technology	[75]
Tree Species Classification Based on Sentinel-2 Imagery and Random Forest Classifier in the Eastern Regions of the Qilian Mountains	Qilian Mountains, China	SENTINEL-2 images	[14]
Detection of flood disaster system based on IoT, big data, and convolutional deep neural network	Sairam Institute of Technology, India	CDNN classifier, ANN, DL, deep-learning neural network (DNN)	[30]
A multisensor data fusion approach for creating variable depth tillage zones	Newbury, UK	VRT	[27]
A Data Fusion Method for Yield and Soil Sensor Maps	Veris Technologies Inc., Kansas, USA	IoT, GPS, soil data maps, yield data maps	[21]
Computer Vision and Deep-learning-enabled Weed Detection Model for Precision Agriculture		Computer vision, DL, IoT, smartphone	[25]
Short Communication: Spatial Dependence Analysis as a Tool to Detect the Hidden Heterogeneity in a Kenaf Field	Jeju National University kenaf-breeding field, Jeju, Republic of Korea	LISA analysis	[22]
Evaluation of Soybean Wildfire Prediction via Hyperspectral Imaging	Kyungpook National University, Daegu, Republic of Korea	Hyperspectral transmission imagery, multispectral camera, Python	[32]
Field road classification for GNSS recordings of agricultural machinery using pixel-level visual features	Beijing, China	GNSS	[46]
A New Procedure for Combining UAV-Based Imagery and Machine Learning in Precision Agriculture	Alma Mater Studiorum University of Bologna, Bologna, Italy	UAV, GIS, ML	[45]
Cooperative Heterogeneous Robots for Autonomous Insects Trap Monitoring System in a Precision Agriculture Scenario	Campus de Santa Apolónia, Bragança, Portugal	UAV	[49]
Drought Stress Restoration Frequencies of Phenotypic Indicators in Early Vegetative Stages of Soybean (Glycine max L.)	Rural Development Administration, LemnaTec, Germany	RGB images, Python	[57]
Durian Farmer Adoption of Smart-Farming Technology: A Case Study of Chumphon Province	Kasetsart University, Bangkok, Thailand	IoT, UAV	[76]

5. Barriers to Adapting New Technologies in Precision Agriculture

High-tech technologies from the fourth industrial revolution have the potential to revolutionize the agriculture industry, enabling more efficient and sustainable practices

while improving productivity and reducing resource wastage. The adaptation of these intelligent, advanced technologies in precision agriculture is still in its early stages, and as such, there exist several barriers (Table 3) that must be addressed to facilitate the transformation of precision agriculture. However, it is essential to carefully consider the specific requirements, challenges, and implementation considerations for each technology in the context of the agricultural operation at hand.

A lack of interdisciplinary skills is one of the major roadblocks, as big data engineers, data analysts, and data scientists do not have an agricultural background. On the other hand, farmers with long experience and practical knowledge are not educated enough to handle high technology like artificial intelligence [8]. The production and development costs of high-tech applications and the capital for establishing them in real-world agriculture are also high [78]. This high cost of the production and implementation of advanced technologies may render them inaccessible to small-scale farmers, who may lack the financial resources to invest in such technologies [79].

Furthermore, the unavailability of affordable technologies for small-scale farmers may create a digital divide, where only large-scale, educated farmers may be able to benefit from such technologies [20,80]. In the unequal distribution of resources in the world, it is difficult for certain groups to reach for such new technological inventions. The implementation of precision agriculture trends in many developing agricultural countries has become a difficult task due to lack of necessary funds, lack of confidence in the technologies, lack of proper infrastructure, lack of necessary resources, etc. [8,76,78,81]. Additionally, the lack of sufficient energy in rural areas hinders the use of new technologies, even as science strives to develop wireless power transfer methods and ambient or on-site energy-generating methods [8]. Furthermore, low digital literacy and unequal accessibility to digital technologies in rural areas, coupled with connectivity issues, pose significant challenges in establishing sustainable intelligent technologies in agricultural processes [20,79,82].

Limited computer power, storage capacity, and processing speed and high energy consumption by batteries are some technical obstacles in precision agricultural adaptations [43], especially when dealing with big data. In addition, collecting and analyzing data from agricultural operations may raise concerns about data privacy and security [4]. These data are heterogenous and, when transferring and storing vast amounts, software platforms from private companies are needed. This reveals some ownership controversies of data [78]. Blockchain interoperability, privacy problems, data leakage, cyber terrorism, and some nonrepudiation issues associated with big data are still difficulties in precision agriculture [5,8,78,83], thereby causing farmers to be reluctant to share their data with third-party service providers [4,16]. In many areas where agriculture is practiced, reliable internet connectivity, which is essential for collecting, transmitting, and analyzing data, may not be readily available [7] and, thus, may affect the absorption capacities of novel technologies [80].

The implementation of trending technologies requires technical expertise that may be unavailable in some regions, leading to job displacement and unemployment as new technologies increase the demand for highly skilled laborers while decreasing opportunities for nonskilled workers. This has implications for both small-scale and family commercial farmers [8,9,20,79]. To effectively use these technologies, farmers and service providers may need training. However, different technologies may not be compatible with each other or with existing agricultural machinery and equipment, which could limit the adoption of advanced technologies in precision agriculture [2,8,84]. Furthermore, the presence of bias and discrimination intertwined with information technology, education, risk-taking attitudes, and western power structures constitute formidable obstacles, hindering the equitable dissemination and advancement of smart-farming technologies, particularly within developing nations [80,85]. This highlights the need for policies on data sharing that cater to both the public and farming industries and are sufficient to ensure data security [20].

Table 3. Advantages, limitations, and main applications of advanced technologies in precision agriculture.

	Advantages	Limitations	Main Applications
Big Data	Data-driven insights Resource optimization Enhanced decision making [1,78]	Robust data management infrastructure Data privacy and security considerations Challenges in integrating heterogeneous data sources [8,78]	Crop yield forecasting Disease and pest management Precision agriculture Predictive analytics Farm management systems [1,6,8]
Machine Vision Technologies	Automated image capture and analysis Enhanced efficiency Reduction of reliance on manual labor Precise monitoring of plant health	Dependence on high-quality images Challenges in image interpretation under varying lighting and environmental conditions	Crop monitoring Disease detection Quality assessment Plant phenotyping Weed detection Yield estimation [8]
IoTs (Internet of Things)	Real-time monitoring Facilitation of data-driven decision making Optimization of resource usage Early detection of issues [78]	Requires reliable network infrastructure Data management and integration challenges Maintenance of hardware [8,16]	Precision agriculture Smart irrigation systems Livestock monitoring Environmental sensing Fishery management Remote farm management [8,13,16,78]
Artificial Intelligence (AI)	Automation and predictive analytics of decision support systems Enhancment of crop management, disease detection, and yield optimization [16,85]	Requires large data sets Computational resources Challenges in explainability and interpretability of AI models	Crop yield prediction Disease detection Pest management Image recognition Mobile expert systems Anomaly detection [8,85]
Machine Learning (ML)	Enables pattern recognition Predictive modeling Data analysis Assists in crop disease diagnosis, yield prediction, and recommendation systems [6,14]	Requires labeled training data, model training, and optimization Potential bias in algorithmic decision making [6,28]	Crop disease diagnosis Yield prediction Soil analysis Yield optimization Breeding optimization Farm management systems [6,28,44]
Deep Learning	Complex pattern recognition Analysis of large data sets Suitable for image and signal processing tasks, disease detection, and plant phenotyping [25,31]	Requires substantial computational resources Large labeled data sets Potential overfitting with limited data [31,86]	Plant disease detection Plant classification Object recognition Plant phenotyping Image-based analysis [25,31,86]
Guidance Systems	Precise navigation and operation of agricultural machinery Reduces overlaps and optimizes resource usage [47]	Requires accurate positioning systems Potential dependency on external signals Challenges in complex terrains [78]	Precision agriculture Automated field operations Autonomous machinery Variable rate application [34,47]
Blockchain Technologies	Provides transparency, traceability, and secure data sharing in the agricultural supply chain Enables trust, verification, and fair transactions	Scalability challenges Energy consumption Integration complexity	Supply chain management Food traceability Quality assurance Fair trade [8,16]
Robotics and Autonomous Systems	Enables automation, precision tasks, and labor reduction Assists in autonomous field operations, weeding, harvesting, and data collection [63,78]	Cost of implementation Limited adaptability to changing field conditions Detection accuracy and technical challenges in complex environments [8,63]	Automated harvesting Weeding Field monitoring Planting Labor-intensive operations [8,34,63]
UAVs (Unmanned Aerial Vehicles)	Remote sensing Aerial imaging Monitoring of large agricultural areas Provides timely data collection Improved field management Cost-effective crop assessment [34,78,86]	Restricted flight regulations Limited payload capacity Challenges in data analysis and interpretation Expensive and break easily [14,34]	Crop monitoring Mapping Aerial imaging Precision agriculture Disease detection [8,34,76]
Unmanned Ground Vehicles	Ground-level monitoring Data collection Field operations in various terrains Assists in precision spraying, mapping, and soil sampling	Limited mobility in challenging environments Dependence on stable terrain conditions	Precision spraying Soil sampling Field mapping Data collection [49,78,86]
High-Throughput Phenotyping	Facilitates rapid and non-destructive measurement of plant traits and characteristics Enhances breeding programs, genetic analysis, and crop improvement [56]	Cost of high-throughput phenotyping platforms Challenges in data interpretation Standardization of measurement protocols	Plant breeding Crop improvement Stress tolerance assessment Genetic analysis Trait selection [56,71]
Telematics	Enables real-time monitoring, tracking, and data collection from vehicles Enhances fleet management, route optimization, and driver safety	Requires reliable connectivity Potential data security concerns Challenges in integrating with existing vehicle systems	Fleet tracking Logistics management Fuel efficiency analysis Predictive maintenance Driver behavior monitoring [2]

6. Future Developments Required

In recent years, the agricultural sector has recognized the potential benefits of adopting new digital technologies. However, the slow rate of adaptation can be attributed to several roadblocks and uncertainties associated with these advancements. Despite the challenges, there is a growing demand for organic foods [78], leading to a shift from sustainable agriculture to smart organic farming. To capitalize on this emerging opportunity, certain steps need to be taken.

One crucial aspect is bridging the gap between expertise personnel and farmers. Providing better education, along with vocational training on novel technological applications, can empower farmers to make effective use of new technologies [20,78]. Governments can play a significant role by creating physical, economic, legal, and social infrastructure that supports the establishment of precision agriculture. Investments in energy infrastructure and communication infrastructure, internet connectivity, service markets, consultancy services, and credit markets can instill trust and willingness among farmers to embrace these technologies [2,20,81].

To further enhance precision agriculture, addressing the lack of professional agricultural sensors is paramount. The design of high-quality, high-resolution, and reliable sensors powered by the IoT that are specifically tailored for the agricultural production environment and the monitoring of plant and animal physiological signs is essential [8]. Moreover, integrating wireless power transfer options can eliminate the need for frequent battery replacements. However, special attention should be given to enabling underground or underwater transmission capabilities [8]. At the same time, on-site energy generation with renewable solar power or biogas energy can be considered comparatively to long distance energy transfer [77]. Although capital investment is high for establishment, it is more profitable than grid power.

Cross-technology communication is another crucial aspect that needs to be addressed. Machine vision for animal monitoring, the development of smart phone applications for the real-time tracking of spatial and temporal variations, and the utilization of 6G mobile networks are promising avenues for generating valuable data and informed decisions [34,83]. Additionally, the emergence of new agricultural systems such as smart hydroponics with the IoT and advancements in breeding technologies with DL and ML technologies contribute to the overall progress of precision agriculture.

Future advancements in precision agricultural technologies hold great promise for the agricultural sector. Overcoming the existing roadblocks and uncertainties is essential to unlocking the full potential of these technologies. By focusing on education, infrastructure development, sensor technology, communication systems, and novel agricultural approaches, we can pave the way for a more efficient, sustainable, and productive future in agriculture.

7. Conclusions

Precision agriculture, now part of Agriculture 4.0, harnesses the power of digitalization for improved farming management. The integration of Industry 4.0 technologies has led to notable trends, such as drones, GPS technology, data analytics, and artificial intelligence, enabling informed decision making in farming practices. Despite these advancements, achieving a fully integrated agricultural management system that comprehensively addresses the complexities of the field requires further studies and innovations. Crucially, the development of adaptive and predictive information systems that effectively integrate diverse data sources is essential for ensuring sustainable and intelligent precision agriculture. While precision agriculture offers numerous benefits, it also poses challenges for its widespread adoption. The initial investment in technology, concerns related to data privacy, and compatibility issues with existing farming systems can be significant barriers for small-scale farmers. Moreover, the scalability and adaptability of these technologies to different farming conditions may limit their applicability in certain regions. Overcoming

these challenges necessitates the implementation of education and training programs to equip farmers with the necessary skills to leverage these technologies effectively.

This review paper serves as a valuable resource for farmers and companies seeking to adopt Industry 4.0 technologies in agriculture. By providing insights into IoT devices, automation systems, data analytics, and precision-farming techniques, this paper fosters awareness and understanding of the opportunities and challenges in smart farming. Armed with this knowledge, companies can make informed decisions regarding technology investments and strategic planning while promoting sustainable farming practices and collaboration within the industry. By embracing Industry 4.0 technologies, farmers and companies can enhance their agricultural operations, optimize resource utilization, and contribute to the collective progress toward smart farming's promising future.

Author Contributions: E.M.B.M.K., A.T.L., S.H., Y.S.C. and S.M. made equal contributions. All authors have read and agreed to the published version of the manuscript.

Funding: The Basic Science Research Program supported this research through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (2019R1A6A1A11052070).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: No additional data is associated with this study.

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

References

1. Alfred, R.; Obit, J.H.; Chin, C.P.-Y.; Haviluddin, H.; Lim, Y. Towards Paddy Rice Smart Farming: A Review on Big Data, Machine Learning, and Rice Production Tasks. *IEEE Access* **2021**, *9*, 50358–50380. [CrossRef]
2. McFadden, J.; Njuki, E.; Griffin, T. Precision Agriculture in the Digital Era: Recent Adoption on U.S. Farms. US Department of Agriculture, Economic Research Service 248. 2023. Available online: https://www.ers.usda.gov (accessed on 2 March 2023).
3. Monteiro, A.; Santos, S.; Gonçalves, P. Precision Agriculture for Crop and Livestock Farming—Brief Review. *Animals* **2021**, *11*, 2345. [CrossRef] [PubMed]
4. Bhat, S.A.; Huang, N.-F. Big Data and AI Revolution in Precision Agriculture: Survey and Challenges. *IEEE Access* **2021**, *9*, 110209–110222. [CrossRef]
5. Yazdinejad, A.; Zolfaghari, B.; Azmoodeh, A.; Dehghantanha, A.; Karimipour, H.; Fraser, E.; Green, A.G.; Russell, C.; Duncan, E. A Review on Security of Smart Farming and Precision Agriculture: Security Aspects, Attacks, Threats and Countermeasures. *Appl. Sci.* **2021**, *11*, 7518. [CrossRef]
6. Cravero, A.; Sepúlveda, S. Use and Adaptations of Machine Learning in Big Data—Applications in Real Cases in Agriculture. *Electronics* **2021**, *10*, 552. [CrossRef]
7. Filipe, J.; Śmiałek, M.; Brodsky, A.; Hammoudi, S. (Eds.) Enterprise Information Systems: 21st International Conference, ICEIS 2019, Heraklion, Crete, Greece, 3–5 May 2019, Revised Selected Papers. In *Lecture Notes in Business Information Processing*; Springer International Publishing: Cham, Switzerland, 2020; Volume 378. [CrossRef]
8. Liu, Y.; Ma, X.; Shu, L.; Hancke, G.P.; Abu-Mahfouz, A.M. From Industry 4.0 to Agriculture 4.0: Current Status, Enabling Technologies, and Research Challenges. *IEEE Trans. Ind. Inform.* **2021**, *17*, 4322–4334. [CrossRef]
9. Trivelli, L.; Apicella, A.; Chiarello, F.; Rana, R.; Fantoni, G.; Tarabella, A. From precision agriculture to Industry 4.0: Unveiling technological connections in the agrifood sector. *Br. Food J.* **2019**, *121*, 1730–1743. [CrossRef]
10. Shin, J.; Mahmud, S.; Rehman, T.U.; Ravichandran, P.; Heung, B.; Chang, Y.K. Trends and Prospect of Machine Vision Technology for Stresses and Diseases Detection in Precision Agriculture. *Agriengineering* **2022**, *5*, 20–39. [CrossRef]
11. Haneklaus, S.; Lilienthal, H.; Schnug, E. 25 years Precision Agriculture in Germany—A retrospective. In Proceedings of the 13th International Conference on Precision Agriculture, St. Louis, MO, USA, 31 July 31–4 August 2016.
12. Hedley, C. The role of precision agriculture for improved nutrient management on farms: Precision agriculture managing farm nutrients. *J. Sci. Food Agric.* **2015**, *95*, 12–19. [CrossRef]
13. Hundal, G.S.; Laux, C.M.; Buckmaster, D.; Sutton, M.J.; Langemeier, M. Exploring Barriers to the Adoption of Internet of Things-Based Precision Agriculture Practices. *Agriculture* **2023**, *13*, 163. [CrossRef]
14. Ma, M.; Liu, J.; Liu, M.; Zeng, J.; Li, Y. Tree Species Classification Based on Sentinel-2 Imagery and Random Forest Classifier in the Eastern Regions of the Qilian Mountains. *Forests* **2021**, *12*, 1736. [CrossRef]
15. Sendros, A.; Drosatos, G.; Efraimidis, P.S.; Tsirliganis, N.C. Blockchain Applications in Agriculture: A Scoping Review. *Appl. Sci.* **2022**, *12*, 8061. [CrossRef]
16. Javaid, M.; Haleem, A.; Singh, R.P.; Suman, R. Enhancing smart farming through the applications of Agriculture 4.0 technologies. *Int. J. Intell. Netw.* **2022**, *3*, 150–164. [CrossRef]

17. Chung, Y.S.; Yoon, S.U.; Heo, S.; Kim, Y.S.; Ahn, J.; Han, G.D. Characterization of Tree Composition using Images from SENTINEL-2: A Case Study with Semiyang Oreum. *J. Environ. Sci. Int.* **2022**, *31*, 735–741. [CrossRef]
18. Marr, B. *Tech Trends in Practice: The 25 Technologies That Are Driving the 4th Industrial Revolution*; John Wiley & Sons: Chichester, UK, 2020.
19. Hegde, P. Precision Agriculture: How Is It Different from Smart Farming? Croping. 24 September 2021. Available online: https://www.cropin.com/blogs/smart-farming-vs-precision-farming-systems (accessed on 10 June 2023).
20. FAO. *In Brief to the State of Food and Agriculture 2022. Leveraging Automation in Agriculture for Transforming Agrifood Systems*; FAO: Rome, Italy, 2022. [CrossRef]
21. Lund, E.D.; Maxton, C.R.; Lund, T.J. *A Data Fusion Method for Yield and Soil Sensor Maps*; International Society of Precision Agriculture: Monticello, IL, USA, 2016.
22. Jang, G.; Kim, D.-W.; Kim, H.-J.; Chung, Y.S. Short Communication: Spatial Dependence Analysis as a Tool to Detect the Hidden Heterogeneity in a Kenaf Field. *Agronomy* **2023**, *13*, 428. [CrossRef]
23. Martin, D.E.; Yang, C. *Field Evaluation of a Variable-Rate Aerial Application System*; International Society of Precision Agriculture: Monticello, IL, USA, 2016.
24. Walter, A.; Finger, R.; Huber, R.; Buchmann, N. Opinion: Smart farming is key to developing sustainable agriculture. *Proc. Natl. Acad. Sci. USA* **2017**, *114*, 6148–6150. [CrossRef] [PubMed]
25. Punithavathi, R.; Rani, A.D.C.; Sughashini, K.R.; Kurangi, C.; Nirmala, M.; Ahmed, H.F.T.; Balamurugan, S.P. Computer Vision and Deep Learning-enabled Weed Detection Model for Precision Agriculture. *Comput. Syst. Sci. Eng.* **2023**, *44*, 2759–2774. [CrossRef]
26. Vellidis, G.; Liakos, V.; Porter, W.; Tucker, M.; Liang, X. *A Dynamic Variable Rate Irrigation Control System*; International Society of Precision Agriculture: Monticello, IL, USA, 2016.
27. Whattoff, D.; Mouazen, A.; Waine, T. A multi sensor data fusion approach for creating variable depth tillage zones. *Adv. Anim. Biosci.* **2017**, *8*, 461–465. [CrossRef]
28. Shaikh, T.A.; Rasool, T.; Lone, F.R. Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming. *Comput. Electron. Agric.* **2022**, *198*, 107119. [CrossRef]
29. Tanikawa, T. Mechanization of Agriculture Considering Its Business Model. In *Smart Plant Factory*; Kozai, T., Ed.; Springer: Singapore, 2018; pp. 241–244.
30. Anbarasan, M.; Muthu, B.; Sivaparthipan, C.B.; Sundarasekar, R.; Kadry, S.; Krishnamoorthy, S.; Dasel, A.A. Detection of flood disaster system based on IoT, big data and convolutional deep neural network. *Comput. Commun.* **2020**, *150*, 150–157. [CrossRef]
31. Saranya, T.; Deisy, C.; Sridevi, S.; Anbananthen, K.S.M. A comparative study of deep learning and Internet of Things for precision agriculture. *Eng. Appl. Artif. Intell.* **2023**, *122*, 106034. [CrossRef]
32. Lay, L.; Lee, H.S.; Tayade, R.; Ghimire, A.; Chung, Y.S.; Yoon, Y.; Kim, Y. Evaluation of Soybean Wildfire Prediction via Hyperspectral Imaging. *Plants* **2023**, *12*, 901. [CrossRef] [PubMed]
33. Chung, S.; Breshears, L.E.; Yoon, J.-Y. Smartphone near infrared monitoring of plant stress. *Comput. Electron. Agric.* **2018**, *154*, 93–98. [CrossRef]
34. Tomaszewski, L.; Kołakowski, R. Mobile Services for Smart Agriculture and Forestry, Biodiversity Monitoring, and Water Management: Challenges for 5G/6G Networks. *Telecom* **2023**, *4*, 67–99. [CrossRef]
35. Khalifeh, A.; Aldahdouh, K.A.; Darabkh, K.A.; Al-Sit, W. A Survey of 5G Emerging Wireless Technologies Featuring LoRaWAN, Sigfox, NB-IoT and LTE-M. In Proceedings of the 2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), Chennai, India, 21–23 March 2019; pp. 561–566. [CrossRef]
36. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. Overview of Cellular LPWAN Technologies for IoT Deployment: Sigfox, LoRaWAN, and NB-IoT. In Proceedings of the 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Athens, Greece, 19–23 March 2018; pp. 197–202. [CrossRef]
37. Zhang, L.; Liang, Y.-C.; Xiao, M. Spectrum Sharing for Internet of Things: A Survey. *IEEE Wirel. Commun.* **2019**, *26*, 132–139. [CrossRef]
38. Walter, T.B.; Lörsch, K.; Stroh, M.-F.; Stich, V. *Specification of 5G Networks for Agricultural Use Cases Using the Example of Harvesters Operated by Swarm Robotics*; PNAS: Zurich, Switzerland, 2023. [CrossRef]
39. Kautish, E. Edge Computing and Intelligent Blockchain in the Construction of Agricultural Supply Chain System. *Acad. J. Agric. Sci.* **2023**, *4*, 81–96. [CrossRef]
40. Gebresenbet, G.; Bosona, T.; Patterson, D.; Persson, H.; Fischer, B.; Mandaluniz, N.; Chirici, G.; Zacepins, A.; Komasilovs, V.; Pitulac, T.; et al. A concept for application of integrated digital technologies to enhance future smart agricultural systems. *Smart Agric. Technol.* **2023**, *5*, 100255. [CrossRef]
41. Koubaa, A.; Ammar, A.; Abdelkader, M.; Alhabashi, Y.; Ghouti, L. AERO: AI-Enabled Remote Sensing Observation with Onboard Edge Computing in UAVs. *Remote Sens.* **2023**, *15*, 1873. [CrossRef]
42. Bourechak, A.; Zedadra, O.; Kouahla, M.N.; Guerrieri, A.; Seridi, H.; Fortino, G. At the Confluence of Artificial Intelligence and Edge Computing in IoT-Based Applications: A Review and New Perspectives. *Sensors* **2023**, *23*, 1639. [CrossRef]
43. Wang, X.; Wu, Z.; Jia, M.; Xu, T.; Pan, C.; Qi, X.; Zhao, M. Lightweight SM-YOLOv5 Tomato Fruit Detection Algorithm for Plant Factory. *Sensors* **2023**, *23*, 3336. [CrossRef]

44. Aasim, M.; Katirci, R.; Baloch, F.S.; Mustafa, Z.; Bakhsh, A.; Nadeem, M.A.; Ali, S.A.; Hatipoğlu, R.; Çiftçi, V.; Habyarimana, E.; et al. Innovation in the Breeding of Common Bean Through a Combined Approach of in vitro Regeneration and Machine Learning Algorithms. *Front. Genet.* **2022**, *13*, 897696. [CrossRef] [PubMed]
45. Fragassa, C.; Vitali, G.; Emmi, L.; Arru, M. A New Procedure for Combining UAV-Based Imagery and Machine Learning in Precision Agriculture. *Sustainability* **2023**, *15*, 998. [CrossRef]
46. Chen, Y.; Quan, L.; Zhang, X.; Zhou, K.; Wu, C. Field-road classification for GNSS recordings of agricultural machinery using pixel-level visual features. *Comput. Electron. Agric.* **2023**, *210*, 107937. [CrossRef]
47. Du, Y.; Saha, S.S.; Sandha, S.S.; Lovekin, A.; Wu, J.; Siddharth, S.; Chowdhary, M.; Jawed, M.K.; Srivastava, M. Neural-Kalman GNSS/INS Navigation for Precision Agriculture. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2023), London, UK, 29 May–2 June 2023. Available online: https://www.researchgate.net/publication/370634153 (accessed on 15 July 2023).
48. Patil, A.S.; Tama, B.A.; Park, Y.; Rhee, K.-H. A Framework for Blockchain Based Secure Smart Green House Farming. In *Advances in Computer Science and Ubiquitous Computing*; Park, J.J., Loia, V., Yi, G., Sung, Y., Eds.; in Lecture Notes in Electrical Engineering; Springer: Singapore, 2018; Volume 474, pp. 1162–1167.
49. Berger, G.S.; Teixeira, M.; Cantieri, A.; Lima, J.; Pereira, A.I.; Valente, A.; de Castro, G.G.R.; Pinto, M.F. Cooperative Heterogeneous Robots for Autonomous Insects Trap Monitoring System in a Precision Agriculture Scenario. *Agriculture* **2023**, *13*, 239. [CrossRef]
50. Kim, D.-W.; Kim, Y.; Kim, K.-H.; Kim, H.-J.; Chung, Y.S. Case Study: Cost-effective Weed Patch Detection by Multi-Spectral Camera Mounted on Unmanned Aerial Vehicle in the Buckwheat Field. *Korean J. Crop Sci.* **2019**, *64*, 159–164. [CrossRef]
51. Thakur, S.; Sharma, S.; Barela, A.; Nagre, S.P. Plant phenomics through proximal remote sensing: A review for improved crop yield. *Pharma Innov. J.* **2023**, *12*, 2432–2442.
52. Xie, P.; Du, R.; Ma, Z.; Cen, H. Generating 3D Multispectral Point Clouds of Plants with Fusion of Snapshot Spectral and RGB-D Images. *Plant Phenomics* **2023**, *5*, 0040. [CrossRef] [PubMed]
53. Arunachalam, A.; Andreasson, H. Real-time plant phenomics under robotic farming setup: A vision-based platform for complex plant phenotyping tasks. *Comput. Electr. Eng.* **2021**, *92*, 107098. [CrossRef]
54. Ngongoma, M.S.P.; Kabeya, M.; Moloi, K. Maximizing a Farm Yield Through Precision Agriculture utilizing Fourth Industrial Revolution (4IR) Tools and Space Technology. *Engineering* **2023**, *preprint*. [CrossRef]
55. Li, K.; Zhu, X.; Qiao, C.; Zhang, L.; Gao, W.; Wang, Y. The Gray Mold Spore Detection of Cucumber Based on Microscopic Image and Deep Learning. *Plant Phenomics* **2023**, *5*, 0011. [CrossRef]
56. Wong, C.Y.; E Gilbert, M.; A Pierce, M.; A Parker, T.; Palkovic, A.; Gepts, P.; Magney, T.S.; Buckley, T.N. Hyperspectral Remote Sensing for Phenotyping the Physiological Drought Response of Common and Tepary Bean. *Plant Phenomics* **2023**, *5*, 0021. [CrossRef]
57. Kim, J.; Lee, C.; Park, J.-E.; Mansoor, S.; Chung, Y.S.; Kim, K. Drought Stress Restoration Frequencies of Phenotypic Indicators in Early Vegetative Stages of Soybean (*Glycine max* L.). *Sustainability* **2023**, *15*, 4852. [CrossRef]
58. Araus, J.L.; Cairns, J.E. Field high-throughput phenotyping: The new crop breeding frontier. *Trends Plant Sci.* **2014**, *19*, 52–61. [CrossRef]
59. Zhang, X.; Huang, C.; Wu, D.; Qiao, F.; Li, W.; Duan, L.; Wang, K.; Xiao, Y.; Chen, G.; Liu, Q.; et al. High-Throughput Phenotyping and QTL Mapping Reveals the Genetic Architecture of Maize Plant Growth. *Plant Physiol.* **2017**, *173*, 1554–1564. [CrossRef] [PubMed]
60. Guo, W.; Carroll, M.E.; Singh, A.; Swetnam, T.L.; Merchant, N.; Sarkar, S.; Singh, A.K.; Ganapathysubramanian, B. UAS-Based Plant Phenotyping for Research and Breeding Applications. *Plant Phenomics* **2021**, *2021*, 9840192. [CrossRef] [PubMed]
61. Mark, T.B.; Whitacre, B.; Griffin, T.W. Assessing the Value of Broadband Connectivity for Big Data and Telematics: Technical Efficiency. In Proceedings of the Southern Agricultural Economics Association's 2015 Annual Meeting, Atlanta, Georgia, 31 January–3 February 2015.
62. Nootjaroen, M.P. Adoption of Tractor Technology by Thai Rice Farmers: The Case of Kubota Tractors with Telematics Systems. Master's Thesis, Thammasat University, Bangkok, Thailand, 2020.
63. Hua, Y.; Nagasaka, K. Recent patents on intelligent automated fruit harvesting robots for sweet pepper and apple. *J. Appl. Hortic.* **2023**, *25*, 65–68. [CrossRef]
64. Davidson, J.R.; Hohimer, C.J.; Mo, C.; Karkee, M. Dual Robot Coordination for Apple Harvesting. In Proceedings of the 2017 American Society of Agricultural and Biological Engineers Annual International Meeting, Spokane, WA, USA, 16–19 July 2017. [CrossRef]
65. Lehnert, C.; English, A.; McCool, C.; Tow, A.W.; Perez, T. Autonomous Sweet Pepper Harvesting for Protected Cropping Systems. *IEEE Robot. Autom. Lett.* **2017**, *2*, 872–879. [CrossRef]
66. AFP. AI-Powered Robots Lend a Hand with Fruit Harvesting. NEWS 18, Israel. 13 July 2023. Available online: https://www.news18.com/viral/ai-powered-robots-lend-a-hand-with-fruit-harvesting-8323279.html (accessed on 16 July 2023).
67. Ghafar, A.S.A.; Hajjaj, S.S.H.; Gsangaya, K.R.; Sultan, M.T.H.; Mail, M.F.; Hua, L.S. Design and development of a robot for spraying fertilizers and pesticides for agriculture. *Mater. Today Proc.* **2023**, *81*, 242–248. [CrossRef]
68. Ishak, A.H.; Hajjaj, S.S.H.; Gsangaya, K.R.; Sultan, M.T.H.; Mail, M.F.; Hua, L.S. Autonomous fertilizer mixer through the Internet of Things (IoT). *Mater. Today Proc.* **2023**, *81*, 295–301. [CrossRef]

69. Qingchun, F.; Xiu, W.; Xiaonan, W.; Guohua, W. A harvesting robot system for cherry tomato in greenhouse. In Proceedings of the 13th International Conference on Precision Agriculture, St. Louis, MO, USA, 31 July–3 August 2016.
70. Jin, X.; Li, Q.; Zhao, K.; Zhao, B.; He, Z.; Qiu, Z. Development and test of an electric precision seeder for small-size vegetable seeds. *Int. J. Agric. Biol. Eng.* **2019**, *12*, 75–81. [CrossRef]
71. Smolka, M.; Puchberger-Enengl, D.; Bipoun, M.; Klasa, A.; Kiczkajlo, M.; Śmiechowski, W.; Sowiński, P.; Krutzler, C.; Keplinger, F.; Vellekoop, M.J. A mobile lab-on-a-chip device for on-site soil nutrient analysis. *Precis. Agric.* **2017**, *18*, 152–168. [CrossRef]
72. Senapaty, M.K.; Ray, A.; Padhy, N. IoT-Enabled Soil Nutrient Analysis and Crop Recommendation Model for Precision Agriculture. *Computers* **2023**, *12*, 61. [CrossRef]
73. Corbari, C.; Salerno, R.; Ceppi, A.; Telesca, V.; Mancini, M. Smart irrigation forecast using satellite LANDSAT data and meteo-hydrological modeling. *Agric. Water Manag.* **2019**, *212*, 283–294. [CrossRef]
74. Su, L.; Jing, L.; Zeng, X.; Chen, T.; Liu, H.; Kong, Y.; Wang, X.; Yang, X.; Fu, C.; Sun, J.; et al. 3D-Printed Prolamin Scaffolds for Cell-Based Meat Culture. *Adv. Mater.* **2022**, *35*, e2207397. [CrossRef] [PubMed]
75. Qiu, Y.; McClements, D.J.; Chen, J.; Li, C.; Liu, C.; Dai, T. Construction of 3D printed meat analogs from plant-based proteins: Improving the printing performance of soy protein- and gluten-based pastes facilitated by rice protein. *Food Res. Int.* **2023**, *167*, 112635. [CrossRef] [PubMed]
76. Wetchasit, P.; Lilavanichakul, A. Durian Farmer Adoption of Smart Farming Technology: A Case Study of Chumphon Province. *J. Food Sci. Agric. Technol.* **2023**, *7*, 8–13.
77. Al-Ali, A.; Al Nabulsi, A.; Mukhopadhyay, S.; Awal, M.S.; Fernandes, S.; Ailabouni, K. IoT-solar energy powered smart farm irrigation system. *J. Electron. Sci. Technol.* **2019**, *17*, 100017. [CrossRef]
78. Saiz-Rubio, V.; Rovira-Más, F. From Smart Farming towards Agriculture 5.0: A Review on Crop Data Management. *Agronomy* **2020**, *10*, 207. [CrossRef]
79. Mizik, T. How can precision farming work on a small scale? A systematic literature review. *Precis. Agric.* **2023**, *24*, 384–406. [CrossRef]
80. Eastwood, C.; Klerkx, L.; Nettle, R. Dynamics and distribution of public and private research and extension roles for technological innovation and diffusion: Case studies of the implementation and adaptation of precision farming technologies. *J. Rural Stud.* **2017**, *49*, 1–12. [CrossRef]
81. Khanna, A.; Kaur, S. An empirical analysis on adoption of precision agricultural techniques among farmers of Punjab for efficient land administration. *Land Use Policy* **2023**, *126*, 106533. [CrossRef]
82. Li, X.; Zhu, L.; Chu, X.; Fu, H. Edge Computing-Enabled Wireless Sensor Networks for Multiple Data Collection Tasks in Smart Agriculture. *J. Sensors* **2020**, *2020*, 4398061. [CrossRef]
83. Karanisa, T.; Achour, Y.; Ouammi, A.; Sayadi, S. Smart greenhouses as the path towards precision agriculture in the food-energy and water nexus: Case study of Qatar. *Environ. Syst. Decis.* **2022**, *42*, 521–546. [CrossRef]
84. Maffezzoli, F.; Ardolino, M.; Bacchetti, A.; Perona, M.; Renga, F. Agriculture 4.0: A systematic literature review on the paradigm, technologies and benefits. *Futures* **2022**, *142*, 102998. [CrossRef]
85. Foster, L.; Szilagyi, K.; Wairegi, A.; Oguamanam, C.; de Beer, J. Smart farming and artificial intelligence in East Africa: Addressing indigeneity, plants, and gender. *Smart Agric. Technol.* **2023**, *3*, 100132. [CrossRef]
86. Bacco, M.; Berton, A.; Ferro, E.; Gennaro, C.; Gotta, A.; Matteoli, S.; Paonessa, F.; Ruggeri, M.; Virone, G.; Zanella, A. Smart farming: Opportunities, challenges and technology enablers. In Proceedings of the 2018 IoT Vertical and Topical Summit on Agriculture—Tuscany (IOT Tuscany), Tuscany, Italy, 8–9 May 2018; pp. 1–6. [CrossRef]

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
www.mdpi.com

Agriculture Editorial Office
E-mail: agriculture@mdpi.com
www.mdpi.com/journal/agriculture

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-03928-598-3