

Article

Photovoltaic Module Surface Defect Detection Model Based on Improved YOLOv8s

Chengjie Zhong ¹, Hao Zhu ¹, Jinyi Huang ¹, Yanchong Feng ¹, Longyang Liu ¹ and Weijia Zheng ^{2,*} 

¹ Zhaoqing Power Supply Bureau, Guangdong Power Grid Co., Ltd., Zhaoqing 526060, China; zcj611822@163.com (C.Z.); 15156409440@163.com (H.Z.); 13425288165@163.com (J.H.); 13527013030@163.com (Y.F.); 13929868156@163.com (L.L.)

² School of Mechanical and Electrical Engineering and Automation, Foshan University, Foshan 528000, China

* Correspondence: wjzheng@fosu.edu.cn

Abstract

Overcoming the limited accuracy and poor multi-scale feature capture of conventional vision methods, we present a lightweight, highly precise YOLOv8s detector tailored for PV surface defects. First, the C2f module in the backbone network is structurally reconstructed, integrating the SENetV2 enhanced channel attention mechanism to improve multi-scale feature integration capability and reduce the omission of multi-scale and irregular defect features. Second, an efficient multi-scale attention mechanism (EMA) with cross-space interaction is introduced, leveraging a lightweight design to significantly enhance the discriminative representation capability of contextual features. Finally, a composite bounding box loss function (Inner-ShapeIOU) combining InnerIOU and ShapeIOU is proposed to optimize localization accuracy and accelerate model convergence. Experimental results show that the improved model significantly enhances detection accuracy while maintaining model lightweightness, achieving an average precision (mAP) of 92.1% at 116 FPS, which is a 2.7% improvement over the baseline YOLOv8s model.

Keywords: surface defects; fault detection; object detection; photovoltaic modules; YOLOV8

1. Introduction

Although automated PV fault diagnosis has progressed considerably [1–4], climate-induced thermal stress remains a severe threat to long-term array reliability [5]. Rooftop installations in high-temperature regions routinely exceed their designed thermal limits. Beyond causing immediate power drops, this chronic heat exposure accelerates the chemical breakdown of encapsulants and multiplies the occurrence of surface defects. Accurate defect detection is therefore essential to avert catastrophic thermal failures and curb maintenance costs in harsh operating environments.

Traditional inspection of PV array surface defects mainly relies on manual visual checks. This method has inherent drawbacks, such as high labor costs and long inspection cycles. More importantly, the quality of inspections is easily influenced by subjective factors like the experience and condition of inspectors, leading to high rates of misjudgment and missed detection, making it difficult to meet the requirements of efficient operation and maintenance for modern PV power plants. In recent years, computer vision technologies, represented by deep learning, have shown enormous potential in multiple industrial inspection fields due to their outstanding capabilities in feature extraction and pattern recognition. Consequently, many researchers have introduced



Academic Editor: Andrea Bonfiglio

Received: 10 June 2026

Revised: 9 July 2026

Accepted: 18 July 2026

Published: 22 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

deep learning techniques into the automatic identification and diagnosis of PV array surface defects. Zhang Bide et al. [6] proposed a lightweight improved YOLOv8 model for PV array surface defect detection. By introducing the CSPHet module, PSA attention mechanism, CCFM neck structure, and ACmix global perception module, the model improves detection accuracy while reducing the number of parameters and computational load. Wang Haiqun et al. [7] proposed an improved YOLOv10n-based PV cell defect detection algorithm, which enhances feature fusion capability by removing the original C2f Bottleneck structure and designing the PMSFA_CSP module; it also introduces the FPSC_SENetV2 module to reduce local information loss and improve detail capture. Sun Jianbo et al. [8] built a feature-enhanced PV module fault detection network based on the YOLOv5 framework: by replacing the original loss function with the EIOU loss function and embedding the InRe feature enhancement module at the front end of the detection head, the detection accuracy and efficiency of UAV PV inspections were significantly improved. Peng Ziran et al. [9] addressed challenges such as suboptimal detection precision and frequent misjudgments or missed detections in PV cell defect detection by integrating a spatial-depth attention mechanism into the YOLOv8 architecture and optimizing the loss function to MPDIU, thereby improving model accuracy.

While these methods have advanced detection accuracy and reduced computational overhead, an optimal balance between model lightweighting and detection performance remains elusive: some solutions improve detection performance but have excessive algorithmic complexity, limiting their deployability on resource-constrained lightweight devices; other lightweight improvements reduce computational burden but fail to significantly enhance detection accuracy. Therefore, existing object detection algorithms still have room for breakthroughs in balancing accuracy and efficiency. Recent Transformer-based detectors like RT-DETR have shown high accuracy in general object detection. However, their self-attention mechanisms often require significant computational and memory resources, which limits their direct deployment on the resource-constrained edge devices typically used for remote PV plant inspections.

Current object detection frameworks struggle to achieve an optimal balance between computational efficiency and accuracy. Heavy models provide strong detection performance but are too complex for edge deployment, whereas aggressively lightweight networks typically compromise on precision.

Unlike previous YOLO-based PV inspection models that struggle with complex background noise and irregular defect morphologies, this study develops a modified YOLOv8s network tailored specifically to the needs of PV surface defect detection, successfully resolving this efficiency–accuracy conflict. We introduce three core improvements:

- By embedding the SENetV2 attention mechanism into the standard C2f module, the network retains fine-grained spatial details more effectively and improves multi-scale feature integration.
- We adopt an efficient multi-scale attention mechanism. It aggregates spatial and channel data to boost contextual learning while keeping the computational overhead to a minimum.
- The conventional CIoU is replaced with Inner-ShapeIoU. This custom loss function combines the strengths of InnerIoU and ShapeIoU to accurately evaluate the match between predicted boxes and actual defect contours, thereby refining localization and speeding up convergence.

2. Methodology

YOLOv8s [10] is a highly efficient and widely-adopted single-stage object detection algorithm launched by Ultralytics in January 2023, offering five different model configurations

with varying parameter sizes: n , s , m , l , and x . To achieve better integrated deployment performance, this paper selects the highly efficient YOLOv8s model as the baseline for optimization. The overall structure consists of four core modules in sequence: the Input layer, Backbone layer, Neck layer, and Head layer. The Input layer integrates data augmentation techniques such as Mosaic and Mixup [11], aiming to improve the model's generalization ability and robustness by increasing the diversity of training samples. The Backbone network mainly consists of three parts: Conv, C2f, and SPPF. The Conv layer is responsible for performing down-sampling operations, the C2f module is used for efficient feature extraction, and the SPPF module focuses on expanding the network's receptive field. In the Neck layer, a combination of a Feature Pyramid Network (FPN) and a Path Aggregation Network (PAN) is used for multi-scale feature fusion; the FPN path transmits deep semantic information from top to bottom, while the PAN path transmits shallow positional details from bottom to top. The two are fused through cross-layer connections, ultimately outputting feature maps rich in both semantic and precise positional information. The Head layer receives these fused features and generates prediction information for each pixel, including object class, confidence, and bounding box coordinates. Finally, a confidence threshold is set for preliminary filtering, and Non-Maximum Suppression (NMS) is applied to remove redundant boxes, thereby outputting the final detected object boxes and their related information in the image.

YOLOv8s serves as the baseline for our lightweight design, offering an optimal trade-off between detection precision and inference speed for resource-constrained PV systems. The structure of the improved model is shown in Figure 1, mainly including three optimized components:

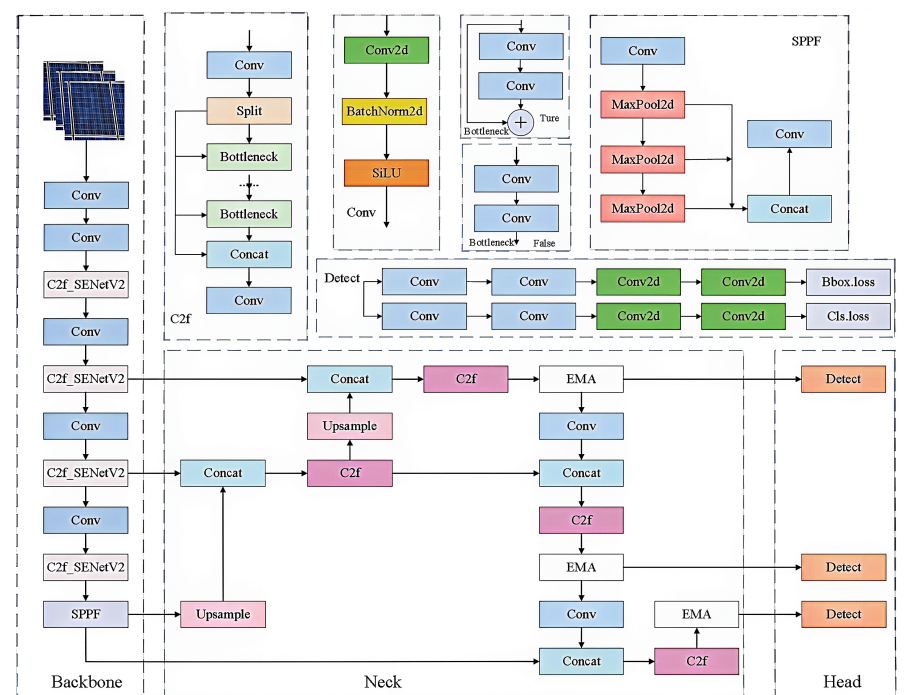


Figure 1. Structure of the Improved YOLOv8s Model.

- (1) In the original C2f (Cross-Stage Partial Fusion) module, the SENetV2 (Squeeze-and-Excitation Network Version 2) channel attention mechanism is integrated. This mechanism adaptively recalibrates the weights of each channel, enhancing the response to key features while suppressing non-salient information, thereby improving the feature extraction accuracy and robustness of the backbone network under complex background interference.
- (2) In the Neck part, an Efficient Multi-scale Attention (EMA) mechanism based on cross-space learning is introduced. This module, with its efficient multi-scale feature

extraction and channel transformation capabilities, collaboratively captures local detailed features and global contextual information, significantly enhancing the network's ability to recognize and focus on key defect features.

- (3) To address the limitations of traditional IOU-based loss functions in handling rotated targets and targets with abnormal aspect ratios, this paper introduces the Inner-ShapeIOU loss function. This loss not only measures the overlap between bounding boxes but also optimizes the evaluation of internal shape matching of targets, effectively mitigating the gradient disappearance issue of CIOU when detecting targets with similar aspect ratios but significantly different sizes, improving localization accuracy.

2.1. C2f-SENetV2 Module

To improve the network's target discrimination performance under complex background interference of photovoltaic modules, this paper integrates SENetV2 (Squeeze-and-Excitation Network Version 2) mechanisms into the C2f (Cross-Stage Partial Fusion) module of the YOLOv8s backbone network to construct a C2f-SENetV2 structure. Surface defects in photovoltaic modules generally exhibit microscopic characteristics, low contrast, and variable morphology, making them easily affected by complex background textures, imaging noise, and defect-like artifacts. In environments with highly similar target features and significant background noise, the standard C2f architecture has limitations in channel feature selectivity and discrimination. C2f-SENetV2 dynamically enhances the weights of key feature channels by modeling dependencies between channels, suppressing non-critical or interference information, thereby improving the model's focusing ability and discrimination accuracy for defect targets in complex contexts.

The SENetV2 module selectively enhances the response of key feature channels through adaptive learning mechanisms while suppressing secondary information, thereby significantly improving the model's ability to distinguish between target and background interferants. Additionally, the improved C2f-SENetV2 module structure is shown in Figure 2, which can more effectively model dependencies between image channels and capture global contextual features, thereby enhancing feature representation quality and achieving more accurate target identification. This module adopts an enhanced SENet architecture, with its innovation being the introduction of SaE units to enhance network representation. The SaE unit integrates SENetV1's squeeze and excitation operations [12], while enhancing the learning capability of global features by integrating multi-branch fully connected layers. As shown in Figure 3, the working principle of the SaE module is to extend the SENet infrastructure: after feature compression, instead of using a single fully connected layer for excitation, it models compressed features through multiple parallel, fully connected branches, aiming to capture complex cross-channel dependencies more granularly. Subsequently, each branch output is excitedly and fused to complete feature recalibration of the channel dimension. Essentially, SENetV2 integrates the design concepts of SENet and ResNeXt, using Squeeze and Excitation operations to dynamically adjust channel weights and enhance the model's channel awareness. A multi-branch FC layer structure is used to learn global features. By concatenating the outputs of these branches, SENetV2 can generate richer and more discriminative global feature representations than SENetV1. Figure 4 visually compares the structural differences between ResNeXt modules, SENet(V1) modules, and SENetV2 modules. The core formula is as follows:

$$Y = x \cdot \sigma \left(\sum_{i=1}^k FC_i(GAP(x)) \right) \quad (1)$$

In this formula, x represents the input feature map. $GAP(\cdot)$ performs Global Average Pooling to squeeze global spatial information, while $FC_i(\cdot)$ denotes the i -th branch of

the multi-branch fully connected structure, which captures cross-channel dependencies. σ represents the Sigmoid activation function that recalibrates channel weights, and $\cdot$ signifies channel-wise multiplication.

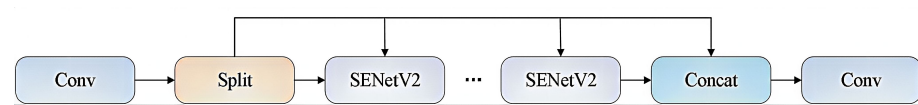


Figure 2. Structure of the C2f-SENetV2 Module.

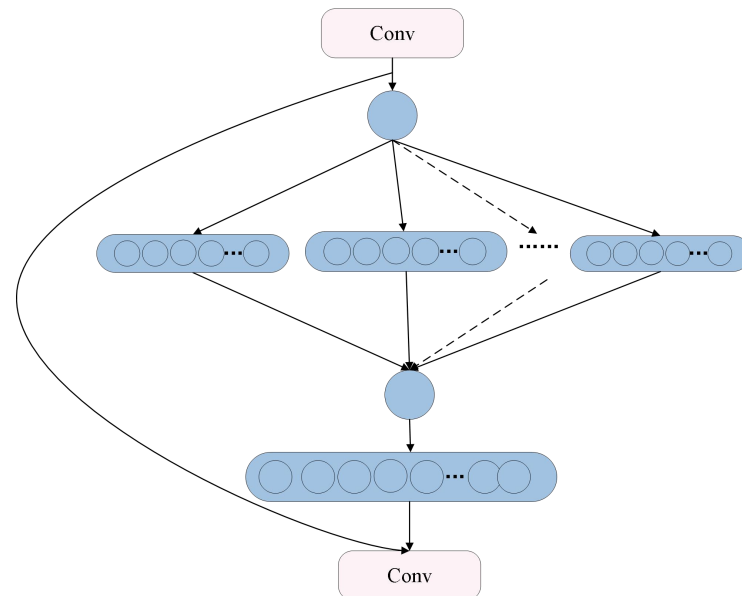


Figure 3. SaE Structure.

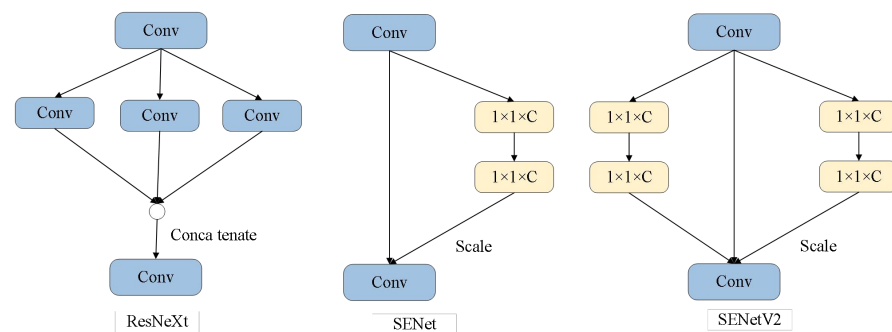


Figure 4. Comparative Architectures of ResNeXt, SENet and SENetV2 Models.

2.2. EMA Attention Mechanism

To enhance the feature extraction capability of the YOLOv8s model in the complex and variable scenarios of photovoltaic modules, this study introduces the Efficient Multi-scale Attention (EMA) module into the YOLOv8s model. The surface defects of photovoltaic modules vary significantly in scale, ranging from microscopic cracks (hidden cracks) and point-like busbar breaks to macroscopic fragments, contamination, and bubbles. At the same time, the imaging background contains dense textures of cell busbars, EVA film textures, dust spots, and reflective interference, forming a highly complex and multi-scale visual environment. Given that the target defect scales are highly variable and highly coupled with dense background interference, standard feature extraction architectures or single-scale attention mechanisms struggle to collaboratively optimize the accurate detection of defects at different scales and the effective filtering of complex background interference. The EMA module extracts and fuses multi-scale contextual information in

parallel and adaptively enhances the representation of key-scale features, significantly improving the model's robustness in detecting multi-scale defects. Its structure is shown in Figure 5 (in the figure, H , W , and C represent the height, width, and channel dimensions of the feature map, respectively; Softmax and Sigmoid are two different activation functions; the symbol " $\oplus$ " represents element-wise addition between feature maps; X AvgPool and Y AvgPool represent average pooling along the X -axis and Y -axis, respectively; $\text{Conv}(3 \times 3)$ indicates a convolution operation with a 3×3 kernel size; Concat represents a concatenation operation; Re-weight represents a re-weighting operation (adjusting the weights of features); GroupNorm represents group normalization; AvgPool represents global average pooling; Matmul represents matrix multiplication).

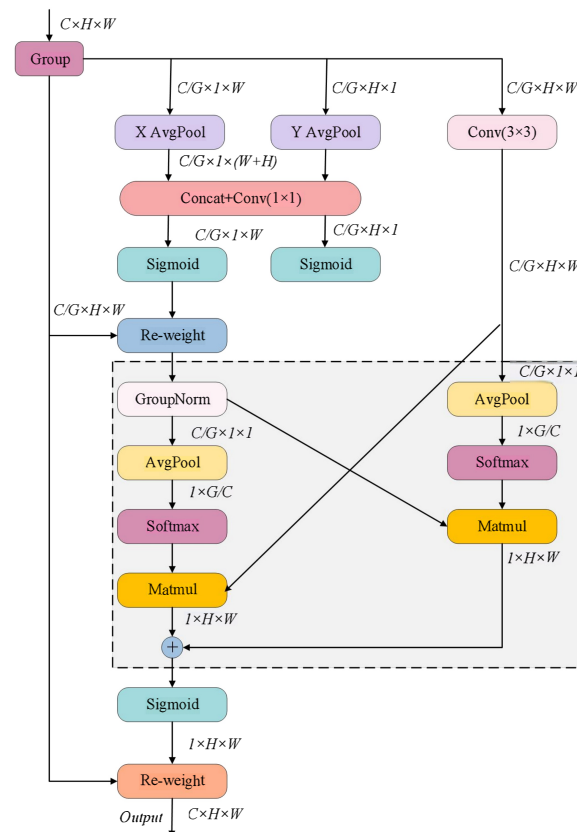


Figure 5. Structure of the EMA Enhanced Module.

The EMA module draws on the multi-branch design concept of the Inception architecture and its multi-scale convolution strategy. After completing feature grouping, the module sets up three parallel branches: two of them are 1×1 sub-networks that encode channels along the horizontal and vertical spatial dimensions, respectively, each generating corresponding feature maps to capture long-range dependencies in different directions. That is:

$$Z_c^H(H) = \frac{1}{W} \sum_{i=0}^W X_c(H, i) \quad (2)$$

$$Z_c^W(W) = \frac{1}{H} \sum_{j=0}^H X_c(j, W) \quad (3)$$

In the formula: X_c —the input feature of the C -th channel, whose feature position is jointly defined by the horizontal direction i and the vertical direction j . After concatenating the feature maps generated by the encoding, they are processed through a shared 1×1

convolutional layer, which deepens feature extraction while maintaining the dimensions unchanged, ultimately obtaining the feature map $D \in \mathbb{R}^{1 \times HW \times (C/G)}$, that is:

$$D = \text{Conv}(\text{Concat}[Z_c^H(H), Z_c^W(W)]) \quad (4)$$

In the formula: Concat—feature concatenation; Conv 1×1 —convolution operation with a 1×1 kernel. The decomposed feature map D needs to be non-linearly activated using the Sigmoid function. In addition, this branch uses a 3×3 convolution to achieve cross-channel information interaction and feature space expansion.

After the two 1×1 branch subnets complete weight adjustment and dimensional transformation, global spatial information is aggregated using two-dimensional average pooling, and the final output is denoted as the feature map Z_c . The process can be represented as:

$$Z_c = \frac{\sum_m^H \sum_n^W X_c(m, n)}{HW} \quad (5)$$

In the formula: (m, n) —the specific spatial coordinates of the feature. First, apply the Softmax function to Z_c , and then perform a matrix dot product of its result with the output from the 3×3 parallel branch, thereby generating a spatial attention map. In the 3×3 branch, a two-dimensional global average pooling is also first performed, followed by processing through the Softmax function to obtain another attention map that retains precise spatial location information. Finally, the outputs of each feature map within the group are integrated to obtain two sets of spatial attention weights, and the Sigmoid function is used to complete the final output of the features.

2.3. Improvement of the Loss Function

Existing IOU-based loss function improvements, such as CIOU, DIOU, and SIOU, are still fundamentally limited to the traditional paradigm of “optimizing bounding box regression.” These methods aim to refine the geometric relationship between two bounding boxes, such as center point distance, aspect ratio, or orientation angle; essentially, they allow a predicted rectangle to geometrically better match another ground truth rectangle. However, in scenarios such as photovoltaic module defect detection, the actual targets (e.g., cracks, corrosion spots) often have irregular, non-rectangular shapes. To address this, Inner-ShapeIOU introduces a new dimension of “in-box shape fitting”. Its optimization goal is, while ensuring alignment outside the bounding box, to further encourage the internal region defined by the predicted box to more closely conform to the irregular pixel distribution of the target. This mechanism enhances the geometrical sensitivity of the loss function to the target shape representation. Among them, Shape-IOU measures based on shape similarity, considering both bounding box overlap and shape similarity [13]; Inner-IOU focuses on the degree of matching within the box region [14]. Inner-ShapeIOU combines the advantages of both, making it more effective in improving localization accuracy and reducing missed and false detections for complex surface morphologies, variable scales, or partially occluded defect targets on photovoltaic modules. The structure of the ShapeIOU loss function is shown in Figure 6.

The calculation process of the Shape-IOU loss function is:

$$L = \frac{|b \cap b^{gt}|}{|b \cup b^{gt}|} \quad (6)$$

$$ww = \frac{2 \times (w^{gt})^S}{(w^{gt})^S + (h^{gt})^S} \quad (7)$$

$$hh = \frac{2 \times (h^{gt})^S}{(w^{gt})^S + (h^{gt})^S} \quad (8)$$

$$d_{Shape} = hh \times \frac{(x_c - x_c^{gt})^2}{c^2} + ww \times \frac{(y_c - y_c^{gt})^2}{c^2} \quad (9)$$

$$\Omega^{Shape} = \left(1 - e^{-\frac{|w - w^{gt}|}{\max(w, w^{gt})}}\right)^\theta + \left(1 - e^{-\frac{|h - h^{gt}|}{\max(h, h^{gt})}}\right)^\theta \quad (10)$$

$$L_{Shape} = 1 - L + d_{Shape} + 0.5 \times \Omega^{Shape} \quad (11)$$

where: S —scaling factor, related to the size of the target in the dataset, usually ranging from 0 to 1.5; e —natural constant; c —distance between the centers of the ground truth box and the predicted box; ww —horizontal weight coefficient; hh —vertical weight coefficient, whose value depends on the shape of the GT. d_{Shape} —distance loss, Ω^{Shape} —shape loss, θ —factor normalizing the shape loss (Ω^{Shape}), usually set to 4.

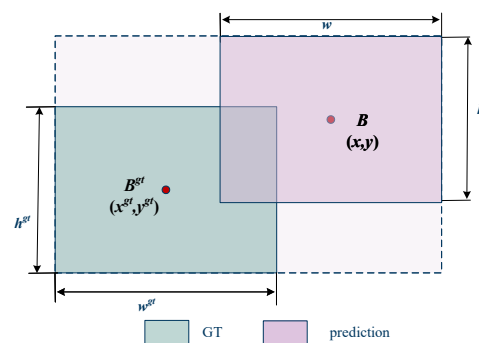


Figure 6. Framework of the Shape-IoU Loss Function.

As shown in the structure in Figure 7, the Inner-IoU loss function optimizes the loss calculation process by incorporating auxiliary bounding boxes. This method not only considers the overall overlap between the predicted box and the ground truth box, but also utilizes internal auxiliary boxes to more precisely adjust the correspondences within their internal regions. This helps the model learn the positional information of small targets more accurately, achieving finer localization. With this mechanism, the loss function significantly improves bounding box localization accuracy for small target workpieces, thereby enhancing overall detection performance.

In the figure, the center coordinates of the ground truth box and its auxiliary ground truth box are represented as (x_c^{gt}, y_c^{gt}) , while the center coordinates of the predicted box and its auxiliary predicted box are represented as (x_c, y_c) . The width and height of the auxiliary ground truth box are defined as w_{Inner}^{gt} and h_{Inner}^{gt} , respectively, whereas the width and height of the auxiliary predicted box are w_{Inner} and h_{Inner} . The calculation process of Inner-IoU is as follows:

$$b_l^{gt} = x_c^{gt} - \frac{w^{gt} \times R}{2}, \quad b_r^{gt} = x_c^{gt} + \frac{w^{gt} \times R}{2} \quad (12)$$

$$b_t^{gt} = y_c^{gt} - \frac{h^{gt} \times R}{2}, \quad b_b^{gt} = y_c^{gt} + \frac{h^{gt} \times R}{2} \quad (13)$$

$$b_l = x_c - \frac{w \times R}{2}, \quad b_r = x_c + \frac{w \times R}{2} \quad (14)$$

$$b_t = y_c - \frac{h \times R}{2}, \quad b_b = y_c + \frac{h \times R}{2} \quad (15)$$

$$inter = \left(\min(b_r^{gt}, b_r) - \max(b_l^{gt}, b_l) \right) \times \left(\min(b_b^{gt}, b_b) - \max(b_t^{gt}, b_t) \right) \quad (16)$$

$$union = (w^{gt} \times h^{gt}) \times (R)^2 + (w \times h) \times (R)^2 - inter \quad (17)$$

$$IOU^{inner} = \frac{I}{U} \quad (18)$$

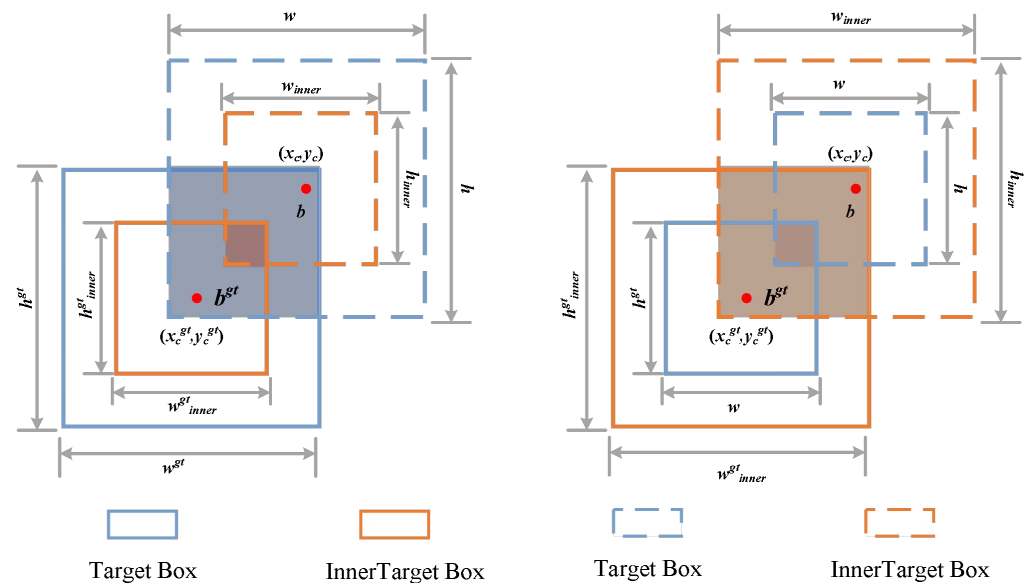


Figure 7. Framework of the Inner-IOU Loss Function.

In the formula: b_l —left boundary of the predicted bounding box; b_r —right boundary of the predicted bounding box; b_t —top boundary of the predicted bounding box; b_b —bottom boundary of the predicted bounding box; b_l^{gt} —left boundary of the ground truth bounding box; b_r^{gt} —right boundary of the ground truth bounding box; b_t^{gt} —top boundary of the ground truth bounding box; b_b^{gt} —bottom boundary of the ground truth bounding box; R —scale factor, usually in the range of 0.5–1.5; I —overlapping area between the predicted box and the ground truth box; U —non-overlapping area between the predicted box and the ground truth box.

Inner-ShapeIOU is a bounding box regression loss function that further refines ShapeIOU. Its core idea is to consider global shape alignment between the predicted box and the ground truth box, while additionally introducing a strong constraint on the coincidence of their centers, thereby guiding the model to produce predictions that are highly consistent with the target object both in position and in shape. Its expression formula is as follows:

$$L_{Inner-ShapeIOU} = L_{Shape} + L - L_{Inner} \quad (19)$$

where L_{Shape} denotes shape IOU loss, which inherits and maintains the optimization goal of global shape alignment. L denotes basic IOU term, serving as a reward term, continuously optimizing the overall overlap area to prevent the model from neglecting the fundamental coverage objective during optimization. L_{Inner} denotes internal IOU term, acting as a fine-tuner, by constructing a highly center-point alignment sensitive penalty mechanism through subtracting the internal IOU from the loss: when the center point is misaligned,

the value of L_{Inner} is small, significantly increasing the total loss; when the center point is well aligned, the value of L_{Inner} is large, reducing its contribution to the total loss, thereby forming an effective reward. By substituting the general expansion of L_{Shape} , formula (19) can be specified as:

$$L_{Inner-Shape} = 1 + d_{Shape} + 0.5 \times \Omega^{Shape} - \frac{L}{U} \quad (20)$$

3. Experimental Results and Analysis

3.1. Experimental Setup

The experiment was conducted on a computer running Windows 11, with hardware configurations including an RTX 4060 graphics card and 8 GB of memory. The versions of Python, torch, and CUDA used were 3.8.5, 1.9.0, and 11.6, respectively. In the training settings, the batch size was 16, and the total number of epochs was 200. The optimizer chosen was SGD, with parameters set as follows: initial learning rate 0.01, momentum 0.937, and weight decay 0.0005. Detailed hyperparameters of the experiment can be found in Table 1.

Table 1. Hyperparameter Settings.

Hyperparameter Name	Parameter Value
image-size	640
epochs	200
batch	16
workers	0
device	0
optimizer	SGD
amp	True
lr0	0.01
momentum	0.937
weight-decay	0.0005

3.2. Dataset

To address the scarcity of public datasets for fine-grained PV surface impairments, we constructed a custom dataset of 14,478 images spanning six common categories: bird-drop, clean, dusty, electrical-damage, physical-damage, and snow-covered.

Our data preparation followed four distinct steps. First, we aggregated raw images from industrial PV plants and public web sources. Second, we manually curated the dataset to remove duplicates and near-identical frames. Third, we annotated bounding boxes using LabelImg in the standard YOLO format. Finally, to prevent data leakage, we implemented a ‘partitioning before augmentation’ protocol. The 14,478 raw images were randomly split into training, validation, and test sets at an 8:1:1 ratio. Data augmentation—including Mosaic, affine transformations, and pixel-level perturbations—was then applied exclusively to the training set, while the validation and test sets remained unaugmented to ensure unbiased performance evaluation.

The instance counts and bounding box distributions are illustrated in Figure 8. The natural class imbalance shown in the data—where, for instance, dusty panels appear far more frequently than those with physical damage—accurately mirrors real-world defect frequencies in operational plants.

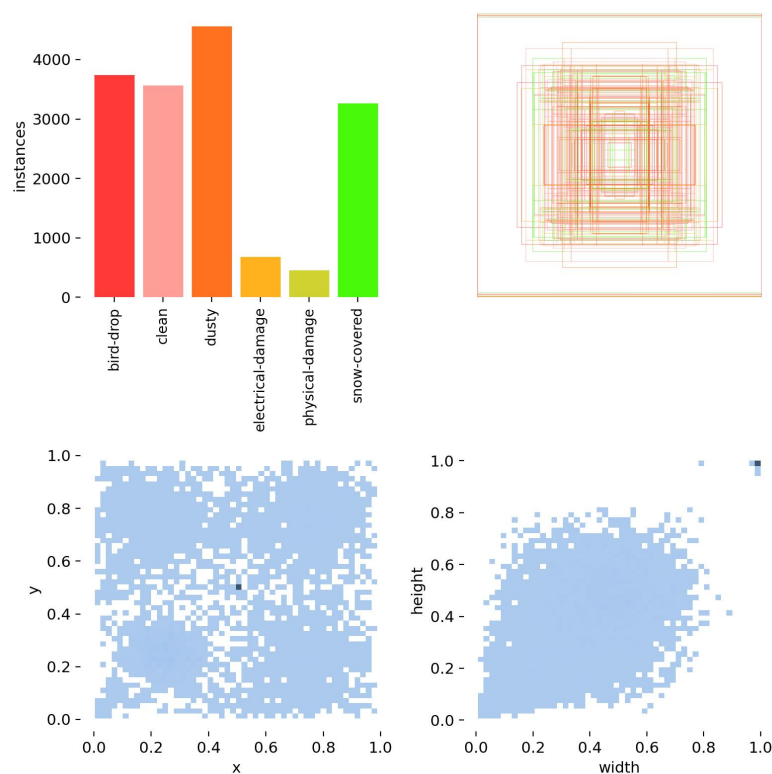


Figure 8. Sample distribution of surface defects in photovoltaic panels.

To improve model generalization while actively preventing data leakage, we applied online data augmentation strictly to the training set after the dataset partition. These techniques included Mosaic (randomly combining four images), affine transformations (translation, scaling, rotation), and pixel-level perturbations (grayscale conversion, Gaussian noise, salt-and-pepper noise). The validation and testing sets were subjected only to standard image resizing without any augmentation, ensuring a completely unbiased performance evaluation.

3.3. Evaluation Metrics

To evaluate model performance, the experiment selected precision (P), recall (R), average precision (AP), and frames per second (FPS) as the core evaluation metrics. The related formulas are as follows:

$$P = \frac{T_P}{T_P + F_P} \quad (21)$$

$$R = \frac{T_P}{T_P + F_N} \quad (22)$$

$$AP = \int_0^1 PR dr \quad (23)$$

$$mAP = \sum_{i=1}^n \frac{A_i}{n} \quad (24)$$

Here, T_P , F_P , and F_N denote the true positive, false positive, and false negative counts, respectively. A_i denotes the average precision (AP) for the i -th class, and $mAP@0.5$ is the mean AP calculated at an IoU threshold of 0.5. FPS indicates the detection speed, defined as the number of images processed per second. This metric serves as the primary indicator for real-time performance, governed by both the algorithm's computational complexity

and the hardware configuration. For practical PV module defect detection, detection speed and localization reliability at a standard IoU threshold are critical. While $mAP@0.5:0.95$ offers a more rigorous evaluation across various thresholds, $mAP@0.5$ is sufficient for determining whether a defect is detected and localized for maintenance purposes. We therefore prioritize $mAP@0.5$ as the primary metric to effectively balance detection accuracy with real-time inference requirements.

3.4. C2f Module Comparison Experiment

To verify the impact of the C2f-SENetV2 module on model performance, different C2f modules were introduced for comparison experiments based on the original improved model. All experiments in this paper were conducted under completely consistent experimental configurations and hyperparameter settings. The experimental results are shown in Table 2.

Table 2. Comparison Experiment of the Improved C2f Module.

C2f Module	P/%	R/%	mAP0.5/%	FPS
C2f-AKConv [15]	82.9	79.1	87.1	200
C2f-FasterBlock [16]	80.4	83.0	87.2	141
C2f-DFF [17]	86.6	80.1	89.3	135
C2f-SENetV2 [18]	81.2	88.1	90.2	135
C2f-iEMA	83.7	85.6	89.5	123
C2f-WTConv [19]	83.4	82.8	89.4	118

As shown in Table 2, in the comparative experiment of the improved C2f module, there are significant differences in the overall performance of different modules in the object detection task. Specifically, the C2f-DFF module achieves the best precision at 86.6%, but its recall is only 80.1%. The C2f-SENetV2 module performs best in both recall and mAP, reaching 88.1% and 90.2%, respectively, demonstrating strong detection accuracy. Additionally, the C2f-AKConv module excels in FPS, reaching 200, indicating a clear advantage in real-time performance. However, this module is relatively weaker in other performance metrics. Overall, the C2f-SENetV2 module performs well across all metrics, proving its suitability for photovoltaic module defect detection tasks. The AP0.5 curves of the various improved C2f modules over epochs are shown in Figure 9.

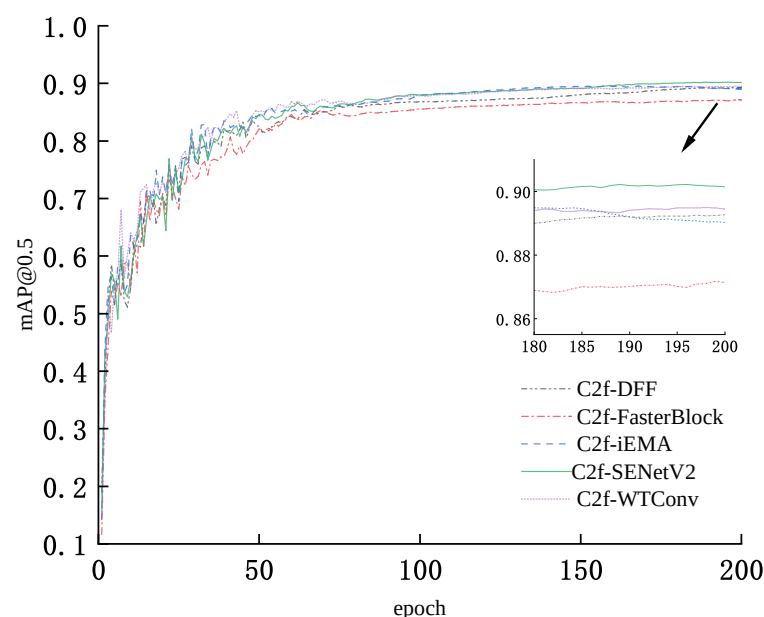


Figure 9. mAP@0.5 Variation of Improved C2f Modules over Training Epochs.

3.5. Comparison Experiments of Loss Functions

To verify the impact of different loss functions on model performance, CIOU [20], WIOU [21], EIOU [22], ShapeIOU, InnerIOU, and Inner-ShapeIOU were introduced on the basis of the original improved model for comparison experiments. The experimental results are shown in Table 3.

Table 3. Comparison Experiment of Loss Functions.

Loss Function	P/%	R/%	mAP0.5/%	FPS
CIOU	82.6	83.6	89.4	137
WIOU	82.2	85.1	89.0	135
EIOU	83.1	80.6	90.2	136
ShapeIOU	86.7	84.1	89.5	135
InnerIOU	85.5	80.5	89.1	137
Inner-ShapeIOU	87.2	83.6	90.5	137

As can be seen from Table 3, the Inner-ShapeIOU loss function cited in this paper has the best precision and mAP compared with other loss functions, which are 87.2% and 90.5%, respectively, and the frame rate FPS reaches 137. This proves that this loss function is more efficient in terms of accuracy for the task of photovoltaic module defect detection. The curves of mAP0.5 for each loss function with respect to epochs are shown in Figure 10.

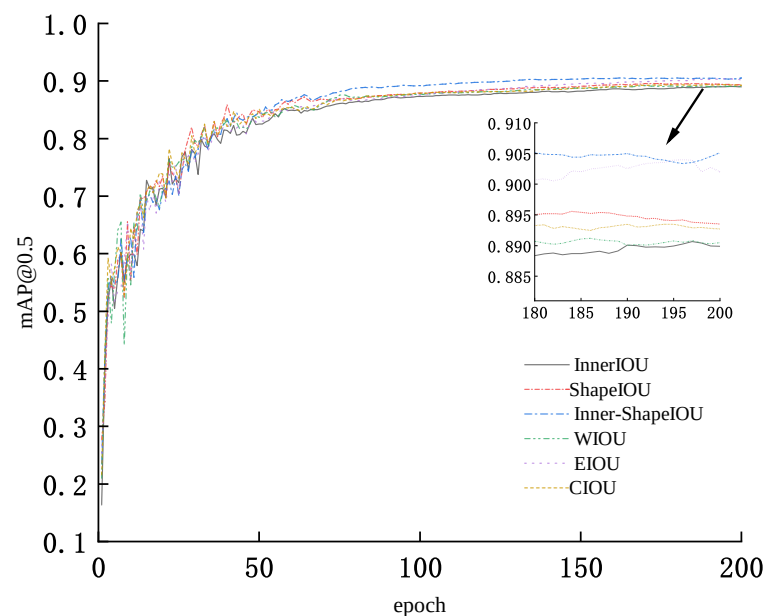


Figure 10. mAP@0.5 Evolution of Various Loss Functions over Training Epochs.

3.6. Ablation Experiment

To further test the performance of the algorithm after improvements, ablation experiments were conducted on a self-made photovoltaic module surface defect dataset using YOLOv8s as the baseline model by combining multiple modules. The C2f-SENetV2 module, EMA attention module, and Inner-ShapeIOU loss function were respectively added to the YOLOv8s model. The same training strategy was used for each set of experiments. In the table, “√” indicates that the module was used, and “—” indicates that the module was not used. The experimental results are shown in Table 4.

As indicated in Table 4, the proposed enhancements do not scale linearly. Integrating C2f-SENetV2, EMA, or Inner-ShapeIOU in isolation yields mAP@0.5 increases of 0.8%, 0.7%, and 1.1%, respectively. However, intermediate setups—such as combining C2f-SENetV2

and Inner-ShapeIOU—cause a slight regression to 90.1% mAP@0.5. This drop points to a representation misalignment between the channel-attention and bounding-box regression tasks during training. Introducing the EMA module corrects this imbalance. Its multi-scale feature alignment smooths out the conflicting features, allowing the network to converge effectively. Ultimately, the complete model achieves 88.3% precision and 92.1% mAP@0.5 with a low variance ($\pm 0.38\%$), confirming that all three components are required for optimal performance.

Table 4. Ablation Experiment.

YOLOv8s	C2f-SENetV2	Inner-ShapeIOU	EMA	P/%	R/%	mAP0.5/%	FPS
✓	—	—	—	82.6	83.6	89.4	137
✓	✓	—	—	81.2	88.1	90.2	135
✓	—	✓	—	87.2	83.6	90.5	137
✓	—	—	✓	90.1	80.4	90.1	119
✓	✓	✓	—	84.0	85.3	90.1	135
✓	—	✓	✓	86.5	80.4	89.8	119
✓	✓	—	✓	80.5	86.3	90.8	116
✓	✓	✓	✓	88.3	81.7	92.1 $\pm$ 0.38	116

The performance variations observed across intermediate configurations stem from the complex interplay of architectural components within the feature extraction pipeline. Our results demonstrate that full integration is required to achieve synergistic optimization, as the final model yields superior and stable convergence compared to partial configurations.

3.7. Comparison Experiments of the YOLO Series Algorithms

Benchmarking against mainstream detectors confirms that our modified YOLOv8s achieves superior performance in PV surface defect recognition. We selected several mainstream object detection algorithms as baselines, including YOLOv5s, YOLOv5-v7, YOLOv6s [23], YOLOv7-tiny [24], YOLOv8s, YOLOv9s [25], YOLOv11s [26], YOLOv8s-ghost [27], as well as some other improved algorithms. All experiments were conducted under consistent experimental conditions, and the experimental results are shown in Table 5.

Table 5. Model Comparison Experiments.

Model	P/%	R/%	mAP0.5/%
YOLOv5-v7	75.1	60.0	66.0
YOLOv7-tiny	76.1	73.7	79.0
YOLOv5s	85.5	77.9	86.6
YOLOv6s	77.8	84.5	86.5
YOLOv9s	84.8	74.4	85.1
YOLOv11s	83.7	80.4	88.5
YOLOv8s	82.6	83.6	89.4
YOLOv8s-EfficientViT2	85.5	80.6	88.4
YOLOv8s-ghost	83.0	81.1	87.5
YOLOv8s-SE-SPDEMA	87.4	79.8	89.4
YOLOv8s-RFACnv [28]	81.6	85.7	89.2
YOLOv8s-ODConv [29]	86.1	81.3	89.2
YOLOv8s-SCDown [30]	86.0	82.0	89.3
YOLOv8s-bifpn-SE	76.7	86.9	88.1
Proposed model	88.3	81.7	92.1

Table 5 validates our selection of the YOLOv8s baseline, which naturally exceeds the mAP@0.5 of other mainstream detectors (YOLOv5s, YOLOv6s, YOLOv7-tiny, YOLOv9s,

and YOLOv11s). Our modified network pushes this performance further, reaching 88.3% precision, 81.7% recall, and 92.1% mAP@0.5. Compared to alternative enhancements like YOLOv8s-EfficientViT2 and YOLOv8s-ODConv, the proposed design yields noticeably more accurate target localization and a lower misclassification rate in PV defect detection.

4. Conclusions

To address the challenges in surface defect detection of photovoltaic modules, including complex background interference, difficulty in extracting multi-scale and irregular defect features, and inaccurate localization of irregular defects, this paper proposes a high-precision detection model based on the improved YOLOv8s. Our architecture incorporates the C2f-SENetV2 module to merge shallow positional and deep semantic features through cross-stage bidirectional connections, isolating small-scale defects from complex backgrounds. We further integrate an EMA attention module to actively filter out noise and highlight critical defect signatures. To address geometric deviations in irregularly shaped targets, bounding box regression is driven by the Inner-ShapeIOU loss function, ensuring highly precise localization. Experimental results show that the proposed algorithm achieves significant improvements in detection accuracy compared to the original YOLOv8s model, with precision increased by 5.7% to 88.3% and mAP0.5 increased by 2.7% to 92.1%. The improved YOLOv8s model proposed in this paper effectively enhances the detection accuracy of multi-scale and irregular defects under complex backgrounds, providing an efficient and reliable solution for intelligent and automated surface defect detection of photovoltaic modules.

While the proposed method achieves strong detection performance, it presents certain limitations that warrant further investigation. First, the model relies on the assumption of relatively stable illumination. In real-world PV plants, dynamic variations in lighting angles and weather conditions can compromise detection stability. Second, practical deployment is currently constrained by hardware requirements; smooth inference necessitates a GPU with at least 8GB of VRAM, which hinders direct integration into resource-limited edge devices like the NVIDIA Jetson. Although we evaluated real-time performance using inference speed (FPS), this study did not provide a detailed breakdown of hardware-agnostic metrics such as parameter counts, FLOPs, and memory consumption. Finally, complex background noise—such as solar cell grid lines and strong reflections—introduces classification uncertainty, particularly when distinct defects (e.g., severe dirt and physical damage) share highly similar visual characteristics. Although the model demonstrates high real-time efficiency on desktop GPUs, its empirical validation on resource-constrained edge platforms (e.g., NVIDIA Jetson) remains a limitation. Recognizing that physical deployment is essential for actual industrial utility, we have prioritized edge hardware testing as a primary objective for our upcoming work.

Moving forward, our research will address these constraints on multiple fronts. To improve the model's robustness against extreme lighting and weather, we intend to incorporate adversarial training and expanded data augmentation. Concurrently, we will prepare the network for edge deployment via pruning and quantization, which will also allow us to rigorously benchmark hardware-agnostic metrics like FLOPs and parameter counts. We further plan to tackle the inherent class imbalance of industrial PV data through category-specific tuning. In the longer term, exploring multi-modal inputs—such as infrared thermography—and adapting efficient Transformer architectures like RT-DETR will serve as our next major milestones for maximizing the accuracy–efficiency trade-off.

Author Contributions: Conceptualization, C.Z.; methodology, C.Z.; software, H.Z. and Y.F.; validation, J.H. and L.L.; formal analysis, H.Z.; investigation, Y.F. and L.L.; resources, L.L. and W.Z.; data curation, H.Z. and Y.F.; writing—original draft preparation, C.Z.; writing—review and

editing, J.H. and W.Z.; visualization, J.H.; supervision, W.Z.; project administration, W.Z.; funding acquisition, C.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Guangdong Power Grid Co., Ltd. under the project “Auxiliary Decision-Making for Distribution Network Faults Based on Computer Vision and Knowledge Graph Technologies” (Project No. 031200KC23120020).

Data Availability Statement: The datasets presented in this article are not readily available because they are part of an ongoing research project. Requests to access the dataset should be directed to the corresponding author.

Conflicts of Interest: Authors Chengjie Zhong, Hao Zhu, Jinyi Huang, Yanchong Feng and Longyang Liu were employed by the Guangdong Power Grid Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The authors declare that this study received funding from Guangdong Power Grid Co., Ltd. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article or the decision to submit it for publication.

References

1. Jia, L.Y.; Yun, S.N.; Zhao, Z.N.; Li, H.L.; Zhao, S.Y.; Yang, L. Recent progress of short-term forecasting of photovoltaic generation based on artificial neural networks. *Acta Energetica Solaris Sin.* **2022**, *43*, 88–97. (In Chinese)
2. Lyu, Y.; Zheng, Q.; Qi, X.Y.; Fang, F.; Liu, J.Z. A study on fault recognition of photovoltaic module with infrared images based on improved efficientnet. *Chin. J. Sci. Instrum.* **2024**, *45*, 175–184. (In Chinese)
3. Tao, Z.Y.; Yi, T.J.; Lin, S.; Du, F. Lightweight M-CNN solar cell surface defect identification. *Acta Energetica Solaris Sin.* **2024**, *45*, 341–348. (In Chinese)
4. Jiang, L.; Su, J.H.; Li, X.; Wang, H.N. Hot spot detection of photovoltaic array based on fusion of visible and infrared thermal images. *Acta Energetica Solaris Sin.* **2022**, *43*, 393–397. (In Chinese)
5. Wu, H.; Kong, Q.; Huber, M.; Sun, M.; Craig, M.T. Climate change will increase high-temperature risks, degradation, and costs of rooftop photovoltaics globally. *Joule* **2026**, *10*, 102218. [[CrossRef](#)]
6. Zhang, B.; Wang, Z.; Zeng, J.; Tang, Q.; Yan, T.; Wang, R. Surface defect detection algorithm of photovoltaic array based on spatial position and channel perception. *Comput. Eng. Appl.* **2024**, 1–17. (In Chinese)
7. Wang, H.Q.; Wu, Z.K.; Yu, H.F. Improved defect detection algorithm for YOLOv10n photovoltaic cells. *Electron. Meas. Technol.* **2024**, *48*, 52–62. (In Chinese)
8. Sun, J.B.; Wang, L.J.; Ma, J.H.; Gao, W. Photovoltaic module fault detection based on improved YOLOv5s algorithm. *Infrared Technol.* **2023**, *45*, 202–208. (In Chinese)
9. Peng, Z.R.; Wang, S.Y.; Xiao, S.P. A solar cell defect detection model optimized and improved based on YOLOv8. *Opto-Electron. Eng.* **2024**, *51*, 89–103. (In Chinese)
10. Varghese, R.; Sambath, M. Yolov8: A novel object detection algorithm with enhanced performance and robustness. In Proceedings of the 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), Chennai, India, 18–19 April 2024; pp. 1–6.
11. Li, Y.J.; Zhou, J.L.; Mai, Y.H.; Wu, S.; Gao, Y.; Li, Y. YOLOv5 is used in optimization of surface defect detection of solar cells. *Acta Energetica Solaris Sin.* **2024**, *45*, 162–169. (In Chinese)
12. Hu, J.; Li, S.H.; Gang, S. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
13. Zhang, H.; Zhang, S. Shape-IoU: More accurate metric considering bounding box shape and scale. *arXiv* **2023**, arXiv:2311.02877.
14. Zhang, H.; Xu, C.; Zhang, S. Inner-iou: More effective intersection over union loss with auxiliary bounding box. *arXiv* **2023**, arXiv:2311.11587.
15. Zhang, X.; Song, Y.; Song, T.; Yang, D.; Ye, Y.; Zhou, J.; Zhang, L. AKConv: Convolutional kernel with arbitrary sampled shapes and arbitrary number of parameters. *arXiv* **2023**, arXiv:2311.11587.
16. Chen, J.; Kao, S.; He, H.; Zhuo, W.; Wen, S.; Lee, Ch.; Chan, S.-H.G. Run, don’t walk: Chasing higher FLOPS for faster neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 12021–12031.
17. Hu, Y.; Chen, Y.; Li, X.; Feng, J. Dynamic feature fusion for semantic edge detection. *arXiv* **2019**, arXiv:1902.09104.
18. Narayanan, M. SENetV2: Aggregated dense layer for channelwise and global representations. *arXiv* **2023**, arXiv:2311.10807.

19. Finder, S.E.; Amoyal, R.; Treister, E.; Freifeld, O. Wavelet convolutions for large receptive fields. In *Proceedings of the European Conference on Computer Vision, Milan, Italy, 29 September–4 October 2024*; Springer Nature Switzerland: Cham, Switzerland, 2024; pp. 363–380.
20. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [[CrossRef](#)]
21. Tong, Z.J.; Chen, Y.H.; Xu, Z.W.; Yu, R. Wise-IoU: Bounding box regression loss with dynamic focusing mechanism. *arXiv* **2023**, arXiv:2301.10051.
22. Zhang, Y.F.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* **2022**, *506*, 146–157. [[CrossRef](#)]
23. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv* **2022**, arXiv:2209.02976.
24. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2023; pp. 7464–7475.
25. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. Yolov9: Learning what you want to learn using programmable gradient information. In *Proceedings of the European Conference on Computer Vision, Milan, Italy, 29 September–4 October 2024*; Springer Nature Switzerland: Cham, Switzerland, 2024; pp. 1–21.
26. Khanam, R.; Hussain, M. Yolov11: An overview of the key architectural enhancements. *arXiv* **2024**, arXiv:2410.17725.
27. Xiong, E.J.; Zhang, R.F.; Liu, Y.H.; Peng, J.X. Ghost-YOLOv8 detection algorithm for traffic signs. *Comput. Eng. Appl.* **2023**, *59*, 200–207. (In Chinese)
28. Zhang, X.; Liu, C.; Yang, D.; Song, T.; Ye, Y.; Li, K.; Song, Y. RFACConv: Innovating spatial attention and standard convolutional operation. *arXiv* **2023**, arXiv:2304.03198.
29. Li, C.; Zhou, A.; Yao, A. Omni-dimensional dynamic convolution. *arXiv* **2022**, arXiv:2209.07947.
30. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. In *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS 2024)*, Vancouver, BC, Canada, 9–15 December 2024.

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