symmetry

MDPI

Article

A Lightweight PCB Defect Detection Algorithm Based on Improved YOLOv8-PCB

Jianan Wang [1,2], Xin Xie [3], Guoying Liu [1,2,4,*] and Liang Wu [1,2]

[1] School of Software Engineering, Anyang Normal University, Anyang 455000, China; wangjianan@aynu.edu.cn (J.W.); 02031@aynu.edu.cn (L.W.)
[2] Engineering Research Center for Intelligent Industry of Oracle Bone Culture in Henan Province, Anyang Normal University, Anyang 455000, China
[3] School of Information and Software Engineering, East China Jiaotong University, Nanchang 330013, China; xienew@gmail.com
[4] Key Laboratory of Oracle Bone Inscriptions Information Processing, Ministry of Education of China, Anyang 455000, China
* Correspondence: guoying.liu@aynu.edu.cn

Abstract: Tackling the widespread problems of inaccuracies, slow detection speed, and poor adaptability in small object defect detection on PCB circuits, this study suggests a lightweight printed circuit board surface defect identification algorithm, building upon an improved YOLOv8-PCB. This algorithm first introduces the C2f_SHSA attention mechanism in the backbone network, which unites the merits of channel attention and spatial attention, facilitating an efficient fusion of local and global features in a lightweight manner, thereby enhancing the model's identification preciseness for small defects. Subsequently, in the neck network, the C2f_IdentityFormer structure, which combines the C2f structure with the IdentityFormer structure, supplants the initial C2f structure. This enhancement improves the model's sensitivity to subtle features and further optimizes the effect of feature fusion. Eventually, the PIoU is presented to enhance the model's adaptability to small, complex PCB defects with varying sizes and shapes, while also accelerating the mode's convergence speed. Experimental outcomes reveal that the improved YOLOv8-PCB algorithm displays remarkable performance in the PCB dataset, with a Recall rate of 94.0%, a mean Average Precision (mAP) of 96.1%, and an F1 score of 94.35%. Moreover, the model's weight size is only 5.2 MB. Compared to the YOLOv8n baseline model, the Recall rate has a 3.6% improvement, the mAP is raised by 1.8%, and the F1 score is enhanced by 1.9%, while the model's weight is reduced by 17.46%. The enhancements in performance metrics confirm that the improved algorithm not only fulfills the requirements for efficient and real-time detection in PCB surface defect identification tasks but is also better suited for deployment and operation on edge devices.

Keywords: defect detection; YOLOv8; PCB; C2f_SHSA attention mechanisms; C2f_IdentityFormer structure

check for updates

Academic Editor: Theodore E. Simos

Received: 23 January 2025
Revised: 13 February 2025
Accepted: 17 February 2025
Published: 19 February 2025

Citation: Wang, J.; Xie, X.; Liu, G.; Wu, L. A Lightweight PCB Defect Detection Algorithm Based on Improved YOLOv8-PCB. *Symmetry* **2025**, *17*, 309. https://doi.org/10.3390/sym17020309

Copyright: © 2025 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) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Printed circuit board (PCB) is classified as a high-tech and strategic emerging industry in "Made in China 2025" [1]. Serving as the support and interconnection in electronic devices, PCB is known as the "mother of electronic products". As a core component in modern electronic devices, PCBs are widely used in various electronic products due to their characteristics of high integration, lightweight, low cost, and high stability. However, during the manufacturing process of PCBs, various factors such as equipment failures,

operational errors, or design defects may lead to defects such as mouse bites, shorts, open circuits, spurs, missing holes, spurious coppers, and more [2]. If these defects are not detected and addressed in a timely manner, they will directly threaten the performance, quality, and safety of the PCB, subsequently having a severe effect on the reliability and operational lifespan of the entire electronic device.

Faced with this challenge, conventional approaches to PCB defect detection, including manual visual inspection [3] and electrical testing [4], though to some extent effective, have their own apparent limitations. Manual visual inspection is not only inefficient but also susceptible to subjective judgments, leading to persistently high false detection rates. On the other hand, electrical testing is a contact-based detection method that, while improving detection efficiency to some extent, often incurs high costs and may easily cause secondary damage to the PCB during the testing process. More crucially, with the continuous advancement of PCB manufacturing processes, their complexity and integration levels have increased significantly. Traditional detection methods have struggled to meet the urgent demands for high-efficiency and high-quality detection in modern PCB production. Therefore, the capability to swiftly and precisely detect PCB defects is crucial for ensuring the quality of electronic products and improving production efficiency.

In recent years, the accelerated progress in deep learning technology, particularly its remarkable achievements in image recognition and object detection, has provided new ideas and solutions for PCB surface defect detection. Object detection methods utilizing deep learning techniques, due to their efficiency and accuracy, have quickly gained widespread application in defect detection within the industrial sector [5–7]. Against this backdrop, numerous researchers have begun actively prospecting the utilization of deep learning technology for PCB defect identification. For instance, Ding et al. [8] employed the Faster R-CNN framework to build the TDD-Net. This methodology leverages the natural multi-scale and pyramidal hierarchies of deep convolutional networks to build feature pyramids. TDD-Net stands out from existing approaches by introducing two key innovations. Firstly, it employs k-means clustering to design appropriate anchors. Secondly, TDD-Net enhances the relationship between feature maps from different levels and takes advantage of low-level structural information, making it particularly appropriate for detecting minute defects. Ultimately, TDD-Net demonstrated impressive accuracy in detecting tiny defects on PCBs. However, its large model size poses challenges for deployment on integrated systems. Shi et al. [9] put forward the SSDT algorithm, which is built upon the SSD algorithm. Specifically, they presented a semantic ascending module that fuses features from various levels to propagate the semantic properties from deep layers to shallow layers. An attention mechanism is employed to understand the relationships among the features across channels that are to be merged, and a shuffle structure is introduced to address aliasing effects after the fusion process. Furthermore, an enhanced non-maximum suppression method is introduced to minimize the overlap effect when evaluating the entire PCB image, ultimately effectively enhancing the precision of PCB defect identification. Lan et al. [10] optimized the YOLOv3 algorithm using the K-means++ clustering algorithm to enhance anchor box positioning accuracy. By integrating Batch Normalization layers with convolutional layers, they accelerated model inference speed and shortened training time. Moreover, the adoption of the GIoU further boosted the detection performance for PCB defects. Jigang Wu et al. [11] proposed a solution to address the issues of numerous feature extraction layers and a multitude of parameters in the YOLOv4 algorithm by using the MobileNetV3 network to replace the CSPDarknet53 network. This modification led to improved mean Average Precision (mAP) and speed in detection while reducing the model's weight size to 53 MB, rendering it more fitting for deployment on lightweight handheld devices. Wu et al. [12] introduced a lightweight network model that is an im-

provement upon YOLOv5. In this model, two key innovations are implemented. Firstly, Ghost Conv and Ghost Bottleneck methods are employed to achieve a lighter algorithm structure, significantly decreasing the quantity of parameters and computational complexity (Flops) required by the model. Secondly, the dual attention mechanism, combining the SE structure and the CBAM, is integrated to enhance the algorithm's performance, effectively tackling the difficulty of detecting minute defects. Yan Zhao et al. [13] presented a real-time non-contact detection method utilizing deep learning, improving the YOLOv4 algorithm using the Darknet framework. Although this enhanced detection accuracy, the modified network required substantial training time and support from high-performance equipment. Lan et al. [14] further proposed an enhanced YOLOv8-based defect detection algorithm. In this algorithm, a Cross-scale Fusion structure (CFM) is incorporated between the backbone and neck networks. This CFM structure facilitates the fusion of features from three adjacent layers. By leveraging the top-down and bottom-up feature extraction processes inherent in the YOLO network architecture, it enhances the integration of adjacent features from the backbone network, bolstering the neck network's capability for feature fusion and interaction, thereby further improving defect detection performance. Despite these algorithms achieving significant improvements in defect detection accuracy, they still face issues such as numerous model parameters and large sizes, demanding high computational power from devices. Especially when applied on edge devices with relatively limited computational capabilities, the hardware may struggle to meet these requirements. Therefore, further improvements and optimizations are needed for PCB defect detection models to reduce the demand for computational resources and make them more suitable for edge device applications.

Within the field of industrial PCB defect identification, pursuing real-time performance and high-precision detection has always been the core objective. However, existing PCB defect detection models generally face the challenge of balancing detection accuracy and model size, which greatly limits their application on edge devices. Despite the fact that existing deep learning-based target recognition methods have shown certain potential in PCB defect identification, their performance in handling small target defects on PCBs is not ideal, and the algorithms have numerous parameters, placing high demands on computational resources. To address this problem, this paper introduces a lightweight YOLOv8-PCB algorithm for defect detection. This algorithm aims to balance detection accuracy and model size to adapt to the application scenarios of edge devices. Its main innovations include the following:

1. Improved backbone network: The C2f_SHSA is introduced to replace the C2f structure in the backbone network. The improvement optimizes the model for better lightweight performance while simultaneously bolstering its capability in capturing information about small object defects on PCBs, thereby improving the model's expressive power and robustness.
2. Improved neck network: To tackle the problems of diverse defect sizes and complex features on PCBs, the C2f_IdentityFormer structure is presented in the neck network of YOLOv8 to take the place of the original C2f structure. This structure integrates feature data from different scales, thereby enriching the feature representation while also increasing the model's sensitivity to subtle features. As a result, the improved YOLOv8 model maintains fast inference speed while demonstrating superior accuracy and stability in detecting small object defects on PCBs.
3. Optimized loss function: To further augment the precision of the PCB defect detection algorithm and accelerate model convergence, enabling the model to better locate and predict targets, we substitute the original CIoU in YOLOv8 with the PIoU. Compared to CIoU, on the one hand, the PIoU introduces a target size-adaptive penalty

factor, allowing the model to more flexibly handle defects of different sizes during the regression process. This feature enables the improved model to maintain high accuracy when dealing with diverse-sized PCB defects while giving sufficient attention to PCB surface defects of all sizes. On the other hand, the PIoU also adopts a gradient adjustment mechanism that relies on the standard of anchor boxes. This novel approach allows the model to dynamically modify the gradient size according to the anchor box quality during the regression process, guiding the anchor box to progress along a more efficient path. This ensures that the improved model maintains rapid convergence rates and stable performance even in complex scenarios.

2. YOLOv8 Algorithm

YOLOv8, the latest addition to the YOLO series launched by Ultralytics in 2023, encompasses multiple scaled versions including YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x [15], representing a further optimization and upgrade from YOLOv5. The primary differences between YOLOv8n in the YOLOv8 series and YOLOv5 include the following: In the backbone network, YOLOv8n adopts the C2f structure [16], which provides richer gradient flow, to replace the C3 structure in YOLOv5, significantly boosting the model's capability to extract features. Secondly, in the design of the head network, YOLOv8n introduces an innovative decoupled head design that separates the classification task from the detection task. Simultaneously, it utilizes a task-oriented assigner for dynamic allocation of positive samples, a strategy that greatly improves detection accuracy and robustness. Notably, YOLOv8n ensures relatively fast detection speed while providing accuracy suitable for edge devices. We can clearly understand the principles of YOLOv8n from Figure 1. Meanwhile, as can also be observed from that, the neck network and the detection head of the model exhibit advantageous symmetry.

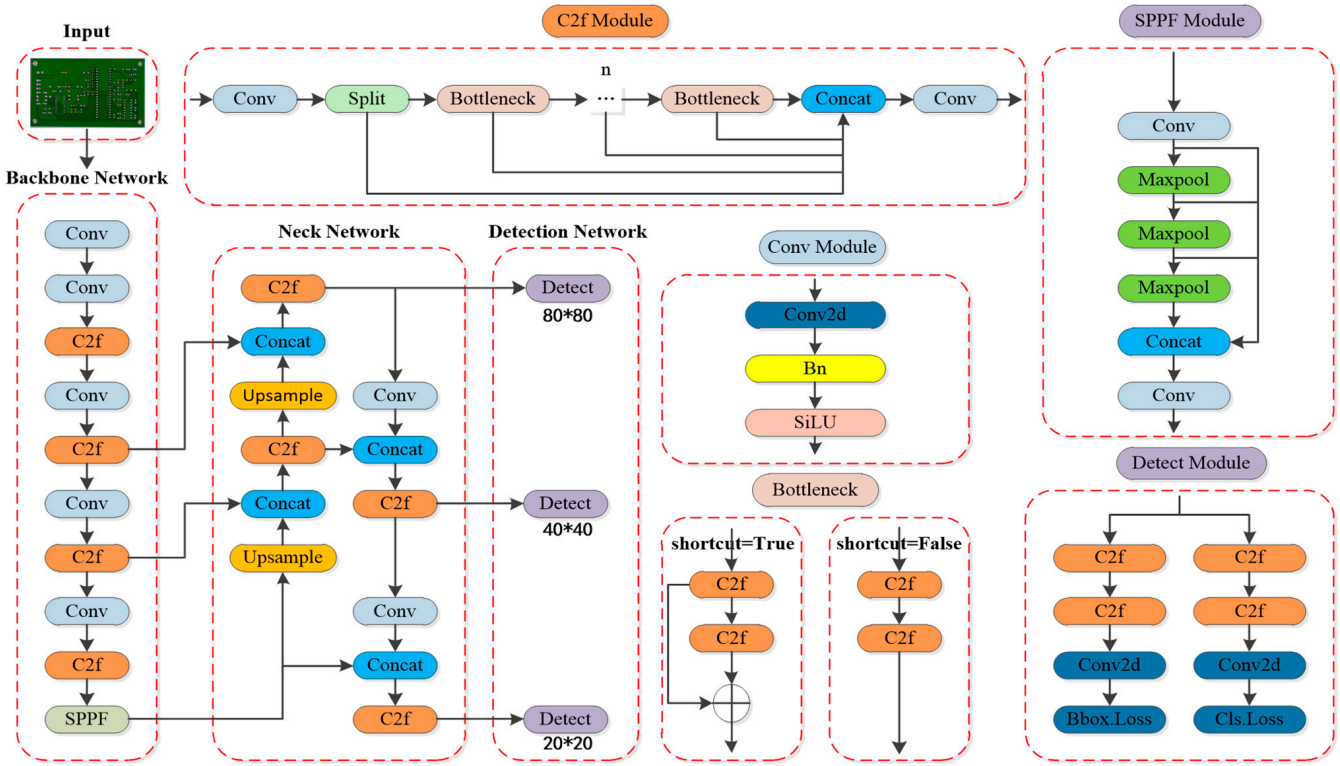


Figure 1. YOLOv8n network structure model.

3. Improved YOLOv8-PCB Algorithm

The YOLOv8n algorithm maintains high efficiency and real-time performance in defect detection tasks while exhibiting excellent detection accuracy and modular design capabilities, making it ideal for lightweight yet high-precision defect detection tasks. However, due to the minute features of PCB defects and their similarity to the background, YOLOv8n still performs poorly in detecting PCB defects, prone to missed and false detections, indicating room for improvement. Therefore, this paper chooses YOLOv8n as the baseline model for enhancement. Specifically, the enhanced YOLOv8-PCB algorithm introduced in this study ingeniously integrates the C2f_SHSA into the backbone network of the YOLOv8n network. This attention mechanism, with lightweight design as its core, can effectively promote the deep fusion of local and global features, significantly improving the mode's capability to identify minute defects. Next, at the neck network level, the innovative C2f_IdentityFormer structure is introduced to substitute the original C2f structure. This module cleverly combines the advantages of C2f and IdentityFormer, not only improving the model's sensitivity in capturing subtle features but also further optimizing the overall effect of feature fusion. Ultimately, the PIoU is presented as the loss function of the improved model to enhance its adaptability to PCB defects and accelerate the convergence speed of the model. The network structure model of the improved algorithm is depicted in Figure 2.

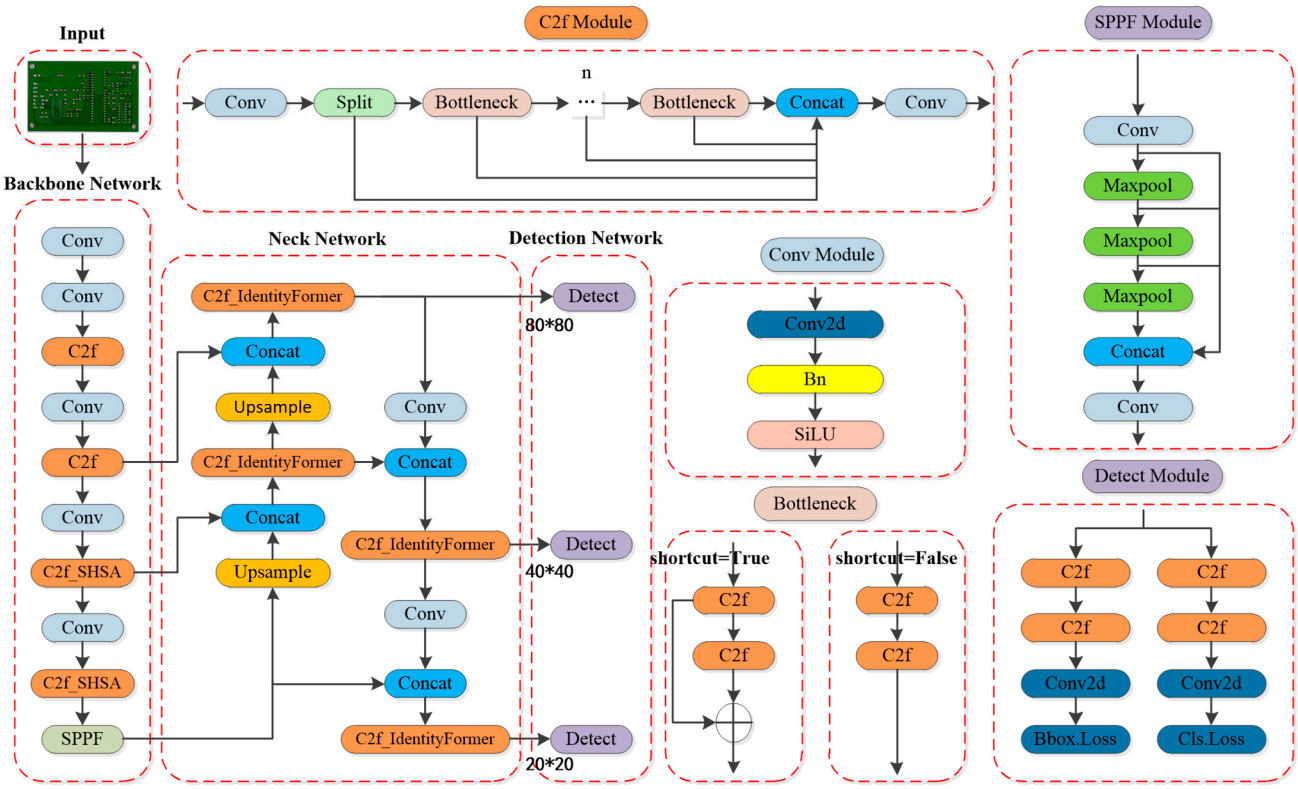


Figure 2. Improved YOLOv8-PCB network structure model.

3.1. C2f_SHSA Structure

The backbone network of YOLOv8n is critical in extracting features from the input image, providing the foundation for subsequent object detection. In the YOLOv8n network structure, the backbone network utilizes the C2f structure, which can extract more abundant feature information. However, the standard convolution in the C2f structure involves a large amount of computation and requires frequent memory access, leading to slow detection speeds and difficulties in model deployment. Therefore, the C2f structure in YOLOv8n is not suitable for PCB surface defect detection. Building on this, the current study integrates

the C2f structure with the SHSA (single-head self-attention) mechanism to enhance the C2f structure [17], forming the C2f_SHSA structure which is a combination of the C2f structure and the SHSA structure. In comparison with the conventional multi-head self-attention mechanism (MHSA) [18], the SHSA captures information using only one attention head. This simplification reduces computational redundancy and memory access costs, enabling efficient fusion of local and global features in a lightweight manner while enhancing the model's detection precision for small defects on PCB surfaces. It is particularly effective for PCB surface defect recognition in resource-constrained environments. Its network structure is illustrated in Figure 3.

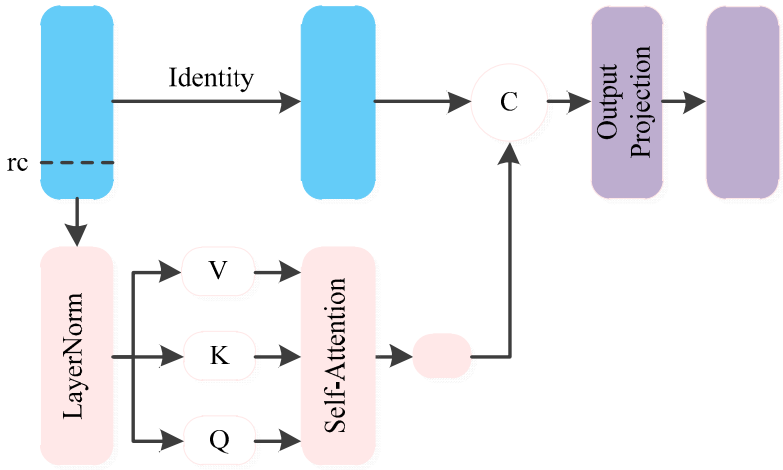


Figure 3. SHSA network structure.

Specifically, the SHSA can be summarized as follows:

$$SHSA(X) = Concat(\widetilde{X}_1, X_2)W^D \tag{1}$$

$$\widetilde{X}_1 = Attention(X_1W^A, X_1W^B, X_1W^C) \tag{2}$$

$$Attention(A, B, C) = Softmax(AB^T/\sqrt{d_{ab}})C \tag{3}$$

$$X_1, X_2 = Split(X, [E_p, E - E_p]) \tag{4}$$

Here, W^A, W^B, W^C, and W^D represent the projection weights, d_{ab} denotes the dimension of the queries and keys (default is 16), and $Concat(\cdot)$ denotes the concatenation operation. To achieve consistent memory access, the initial E_p channels are used to represent the entire feature map. Furthermore, the SHSA's ultimate projection encompasses all channels, extending beyond the initial E_p channels, thereby guaranteeing efficient dissemination of attention features to the entire set of channels.

3.2. C2f_IdentityFormer Structure

In the object detection model of YOLOv8n, the neck network plays a vital role, situated between the backbone and head networks, primarily responsible for feature fusion and enhancement. To further augment model performance, we innovatively introduce the C2f_IdentityFormer structure to substitute the original C2f structure. This structure adopts identity mapping as its core concept [19] to ensure the integrity of information during the feature extraction process. Specifically, the C2f_IdentityFormer structure not only inherits the basic structure of the C2f structure but also introduces an identity mapping mechanism on this basis to enhance the representation ability of features. In the C2f_IdentityFormer structure, which is a combination of the C2f structure and the IdentityFormer structure, the input data first undergo feature transformation through the first convolutional layer, CV_1.

The output is then divided into two branches: to preserve the original information, one branch is transmitted directly to later layers via identity mapping, whereas the complementary branch is processed through multiple Bottleneck modules to learn more sophisticated feature representations. Ultimately, the outputs from both branches are merged along the channel axis before being processed by the subsequent convolutional layer, CV_2, to generate the final result. Mathematically, the output of the C2f_IdentityFormer structure can be formulated as follows:

$$IdentityFormer(X) = CV_2(Concat(W, V)) \tag{5}$$

$$W = IdentityMapping(X_1) \tag{6}$$

$$V = Bottleneck(X_2) \tag{7}$$

where X_1 and X_2 represent two parts of the input data after passing through the CV_1 convolutional layer, respectively, $IdentityMapping(X_1)$ denotes the identity mapping applied to X_1, $Bottleneck(X_2)$ indicates the processing of X_2 through multiple Bottleneck modules, $Concat(\cdot)$ signifies concatenation along the channel dimension, and CV_2 represents the second convolutional layer. Its network structure is shown in Figure 4.

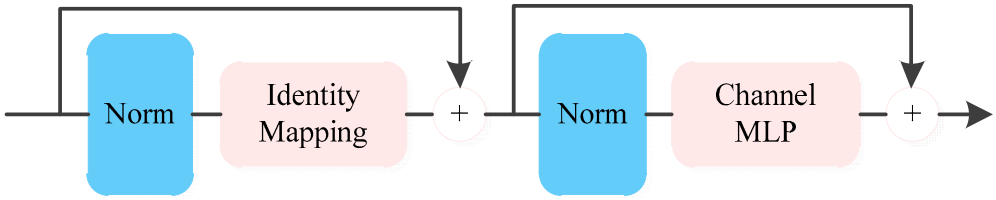


Figure 4. IdentityFormer network structure.

Despite the relatively complex structure of the C2f_IdentityFormer structure, it efficiently integrates features and employs an identity mapping mechanism, effectively enhancing performance without significantly increasing the computational burden of the model. This allows the improved model to maintain high identification correctness while still achieving a relatively fast inference speed, satisfying the real-time demands for PCB defect detection.

3.3. Optimizing the Loss Function of the YOLOv8-PCB Model

The loss function is a crucial component of target identification algorithms, directly impacting the precision of model predictions. Especially in the distinct task of identifying surface defects, the minute size and varying shapes of defects on PCB surfaces elevate the difficulty of defect identification, making the selection of an appropriate loss function even more pivotal. The original CIoU (Complete Intersection over Union) loss function in the YOLOv8n model is a bounding box loss function based on the IoU (Intersection over Union). The IoU loss function primarily focuses on the ratio of the intersection to the union between the predicted box and the ground truth box. When there is a lack of intersection or low overlap between them, the IoU loss function may fail to provide effective gradient information. In contrast, the CIoU significantly improves upon the IoU by introducing multiple geometric factors, such as the distance between the centers of the predicted box and the ground truth box, in addition to penalties for aspect ratio discrepancies [20]. However, the CIoU does not account for the impact of rotation angle, limiting its adaptability to variations in target size and shape. In the context of printed circuit board defect detection tasks, the diverse types and morphologies of PCB defects can lead to decreased model performance and compromised identification accuracy. In comparison, the PIoU, by incorporating a target size-adaptive penalty factor and an anchor

box quality-based gradient adjustment function [21], can direct anchor boxes to regress along more efficient paths, demonstrating stronger adaptability to changes in target size and shape. This contributes to ensuring that the model maintains high accuracy while paying adequate attention to PCB surface defects of various sizes. Given the characteristics of PCB surface defects, this feature of the PIoU also enables it to exhibit higher accuracy and performance in PCB surface defect detection tasks. Therefore, to further enhance PCB defect detection performance, this paper replaces the CIoU in YOLOv8n with the PIoU, aiming to further improve the recognition precision of the modified model and accelerate its convergence speed. The parameters of the L_{PIoU} are illustrated in Figure 5.

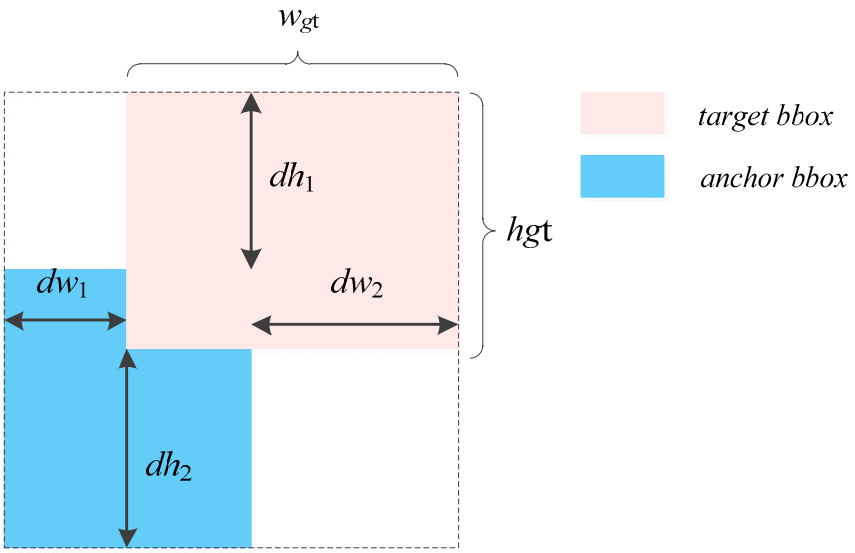


Figure 5. Parameters of the L_{PIoU} loss function.

Specifically, mathematically, the PIoU loss function is described as follows:

$$L_{PIoU} = L_{IoU} + 1 - e^{-p^2}, 0 \leq L_{PIoU} \leq 2 \tag{8}$$

$$L_{IoU} = 1 - \frac{I}{U}, 0 \leq L_{IoU} \leq 1 \tag{9}$$

$$P = (\frac{dw_1}{w_{gt}} + \frac{dw_2}{w_{gt}} + \frac{dh_1}{h_{gt}} + \frac{dh_2}{h_{gt}})/4 \tag{10}$$

Here, I denotes the intersection between the predicted box and the target box, and U signifies their union. The variables dw_1, dw_2, dh_1, and dh_2 represent the absolute distances between corresponding edges of the predicted box and the target box, while w_{gt} and h_{gt} stand for the width and height of the target box.

4. Experiments and Result Analysis

4.1. Experimental Setting

Table 1 provides a comprehensive overview of the configuration settings for our experimental environment. During the training phase, input images are meticulously resized to 640 × 640 pixels to ensure uniformity and consistency in data processing. The training process is extensive, encompassing a total of 200 epochs to allow for thorough learning and convergence. We employ the SGD optimizer, renowned for its effectiveness in such scenarios, and set the batch size to 16 to strike a balance between memory usage and convergence speed. The initial learning rate is prudently initialized at 0.01, a value chosen based on preliminary experiments and industry best practices. Regarding the dataset, it is meticulously allocated into training, validation, and test sets in a proportion of 8:1:1. This

meticulous division ensures that we have an ample amount of data for training the model, enabling it to learn patterns and relationships effectively. Simultaneously, by reserving separate sets for validating and testing the model's performance, we can objectively assess its capabilities and generalizability. This bifurcation not only bolsters the reliability of our experimental results but also provides a more robust assessment of the model's prowess in real-world applications.

Table 1. Experimental setup arrangement.

Item	Description
Operating System	Ubuntu 20.04
CPU	Intel(R) Xeon(R) Platinum 8358P
GPU	NVIDIA A100-SXM4-80GB
Programming Language	Python 3.8.10
Deep Learning Framework	PyTorch 2.0.0
Memory	120 GB
CUDA Version	CUDA 11.8

4.2. Experimental Dataset

4.2.1. Introduction to PCB_DATASET

The experiments conducted in this paper make use of the PCB_DATASET, which was generously released by the Peking University Open Lab on Human Robot Interaction (PKU-HRI) [22]. This valuable dataset comprises a total of 693 images, each meticulously captured to showcase various defects commonly found on printed circuit boards (PCBs). These images cover six distinct and prevalent categories of PCB surface defects, ensuring a comprehensive representation of the issues that can arise during the manufacturing process. To provide a clearer understanding of these defects, Figure 6 offers a detailed visual illustration, highlighting the specific characteristics and variations within each category. This figure serves as a crucial reference for researchers and practitioners alike, facilitating the identification and analysis of different defect types. Moreover, the dataset has been meticulously annotated to enhance its usability for machine learning and image processing tasks. Each defective area within the images is carefully marked using rectangular boxes, providing a precise indication of the defect's location and extent. Complementing these annotations, each image is accompanied by an xml file that contains vital metadata, including the image size, the exact coordinates of the defective areas, and the corresponding defect category.

Lastly, Table 2 presents a summary of the dataset, detailing the total number of images and the distribution of defects across the various categories. This table offers a quick overview of the dataset's composition, allowing researchers to gauge the prevalence of each defect type and plan their experiments accordingly.

Table 2. Number of defective images and defects in the PCB_DATASET.

Defect Types	Image Counts	Defect Counts
mouse_bite	115	492
short	116	491
open_circuit	116	482
spur	115	488
missing_hole	115	497
spurious_copper	116	503
Total	693	2953

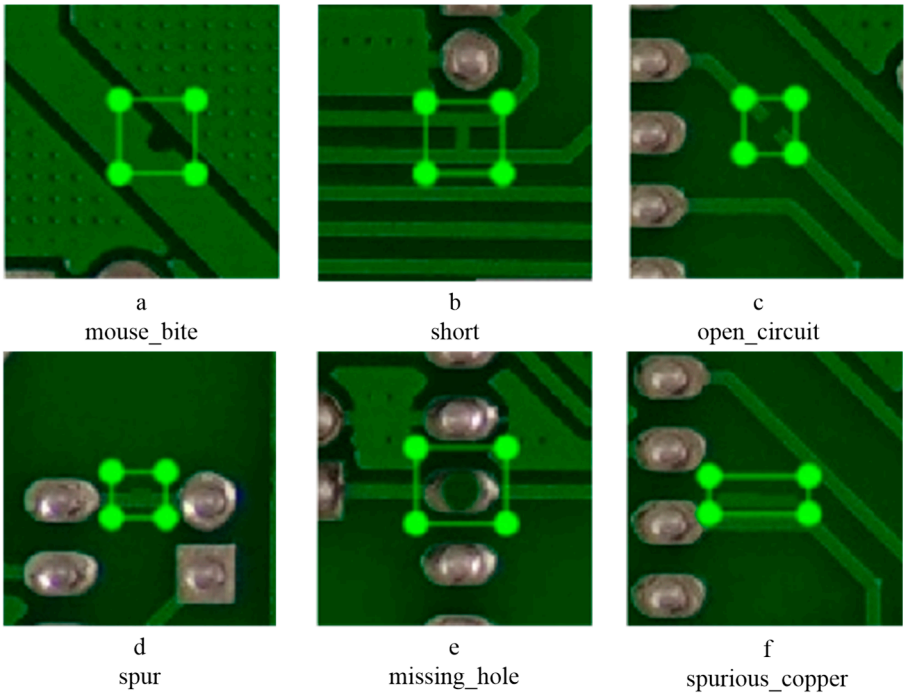


Figure 6. Six prevalent categories of PCB surface defects.

4.2.2. Data Augmentation

Considering the small number of sample images and defects available in the original dataset, direct use for training may fail to yield satisfactory results due to insufficient data volume. Furthermore, the PCB images in the dataset are all well-formed rectangles with minimal tilt angles. In practical applications, if the PCB is placed at an angle or subject to other disturbances, it may adversely affect the detection results. Additionally, factors such as lighting can also impact the detection outcomes. Therefore, to strengthen the network's generalization capabilities [23], and considering the symmetrical characteristics of the PCB circuit board layout, methods for expanding the dataset such as random rotation, shifting, and brightness adjustment were employed to expand the dataset to 2346 images (the PCB image is rotated clockwise or counterclockwise by 5 degrees, with its brightness ranging from 35% to 100% of the original image's brightness). This helps improve the mode's adaptability to unknown or slightly varied data and effectively prevents overfitting. To facilitate the division of the dataset in an 8:1:1 ratio, 690 images out of the 693 in the PCB_DATASET were randomly selected and allocated into training, validation, and test sets with a ratio of 8:1:1. Subsequently, data augmentation was utilized to the training set images to obtain the dataset for experimentation. Table 3 shows the labels of various defect types in the PCB dataset and the corresponding number of images.

Table 3. Number of images and defect labels for each category in the PCB dataset.

Defect Types	PCB_DATASET	The Experimental Dataset of This Paper
mouse_bite	115	388
short	115	397
open_circuit	115	388
spur	115	385
missing_hole	115	394
spurious_copper	115	394
Total	690	2346

In addition, the width-to-height distribution of the dataset is vividly illustrated in Figure 7, offering a comprehensive perspective on the defect dimensions within the PCB images. It becomes abundantly clear that surface defects on PCBs predominantly consist of small target anomalies, with only a negligible fraction classified as medium-sized or large target defects. This statistical insight uncovers a distinctive characteristic inherent to PCB defects, emphasizing their propensity to manifest as minute imperfections on the board's surface. Meanwhile, this observation also holds significant implications for the development and refinement of automated inspection systems. In addition, this peculiar distribution pattern not only underscores the intricacy involved in accurately detecting these tiny flaws but also underscores the necessity for algorithms tailored to recognize and differentiate such fine details amidst the complex circuitry background. Consequently, it has guided our methodological approach towards algorithm design, directing us to meticulously select and refine an advanced version of the YOLOv8 model. Our enhancements have been meticulously crafted with a sharp focus on improving small target recognition capabilities, ensuring that even the most inconspicuous defects are reliably identified.

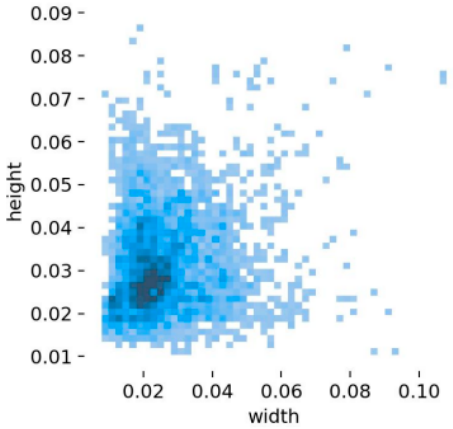


Figure 7. Aspect ratio distribution of our dataset.

4.3. Evaluation Metrics

This paper employs a series of comprehensive performance indicators to thoroughly evaluate the model's performance [24], specifically including Precision (P), Recall (R), mean Average Precision (mAP), and F1 score, as well as assessment metrics reflecting model resource consumption and deployment ease, such as the count of model parameters (Params), computational complexity (GFLOPs), and model weight size. Among them, P precisely defines the proportion of instances predicted as positive that are truly positive. R reveals the proportion of all truly positive instances that are correctly predicted by the model. As a comprehensive indicator, mAP balances and considers both Precision and Recall. The F1 score represents the harmonic mean of Precision and Recall, utilized to assess the detection capability of the model. Furthermore, in terms of resource consumption and deployment, lower values of Params and GFLOPs imply lower demands on computational resources, making it easier to deploy and run the model on edge devices with limited hardware performance. Finally, the model weight size not only intuitively demonstrates the complexity of the model but also profoundly affects the consumption of computational resources. It has a close intrinsic relationship with model performance, providing great convenience for model selection and optimization.

The calculation formulas for the aforementioned evaluation indicators are as follows:

$$P = \frac{TP}{TP + FP} \tag{11}$$

$$R = \frac{TP}{TP + FN} \tag{12}$$

$$AP = \int_0^1 P(R)d(R) \tag{13}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{14}$$

$$F1 = 2\frac{P \times R}{P + R} \tag{15}$$

In the formulas, TP represents the number of correctly detected defects by the model; FP indicates the number of falsely detected defects; and FN stands for the quantity of defects that the model failed to detect. A higher P value suggests better detection accuracy of the model, while a higher R value indicates a lower missed detection rate. AP measures the precision achievement of the model on each detection category, and mAP is the average of AP values across multiple categories, utilized to comprehensively assess the model's achievement. Finally, the $F1$ score is used to measure the recognition capability of the model.

4.4. Ablation Experiment

To ratify the efficacy of the improvements proposed in the YOLOv8-PCB algorithm in this paper, this experiment conducted an ablation study by progressively adding improved modules, using P, R, mAP@0.5, F1, GFLOPs, and weight size as assessment criteria. The outcomes of the ablation study are shown in Table 4.

Table 4. Ablation experiment results.

YOLOv8n	C2f_SHSA	C2f_IdentityFormer	PIoU	P/%	R/%	mAP @0.5/%	F1	GFLOPs	Weight Size /MB
√				94.60	90.40	94.30	92.45	8.1	6.3
√	√			93.00	90.40	93.60	91.68	7.6	5.7
√		√		96.20	91.30	95.20	93.69	7.6	5.8
√	√	√		96.50	92.10	95.60	94.25	7.1	5.2
√	√	√	√	94.70	94.00	96.10	94.35	7.1	5.2

Analysis of the experimental data in Table 4 reveals that, as contrasted with the YOLOv8n baseline model, embedding the SHSA into the backbone network after the C2f, as shown in Group 2, results in a 6.17% reduction in computation and a 9.52% decrease in model weights. This indicates that the improved algorithm facilitates lightweight design. In the neck network, by inserting the IdentityFormer after the C2f, as indicated in Group 3, there is a 6.17% decrease in computation, along with a 0.9% rise in mAP@0.5 and a 1.24% enhancement in F1 score. This suggests that the modified algorithm not only enhances algorithm accuracy and recall rate but also reduces model complexity. It verifies the algorithm's ability to enhance focus on target areas, extract features of targets of varying sizes and shapes, and accelerate computation speed. When both the C2f_SHSA and the C2f_IdentityFormer are integrated into the YOLOv8n network, as shown in Group 4, the computation decreases by 12.35%, mAP@0.5 improves by 1.3%, and F1 score increases by 1.8%. Compared to adding a single module, all evaluation metrics show further improvements. Building upon Group 4, Group 5 experiments utilize the PIoU as the loss function. Against the YOLOv8n baseline model, this configuration reduces computation by 12.35% and model weights by 17.46% while simultaneously increasing mAP@0.5 and F1 score by 1.8% and 1.9%, respectively. This demonstrates that the PIoU enhances the detection capability of the algorithm in low-quality datasets. Therefore, the

improved algorithm achieves optimal enhancement in detection performance. Additionally, with a computation of only 7.1 GFLOPs, this algorithm is easily deployable on edge devices.

4.5. Comparison of AP Metrics for Various Types of Defects

Table 5 presents a comparison of the AP values for various defects between the model before and after improvement. As shown in the table, for the defects of missing hole and spurious copper, the AP values of the improved YOLOv8-PCB algorithm are similar to those of the YOLOv8n baseline model. The improved model does not have an advantage in terms of mouse-bite defects. However, for the defects of open circuit, short circuit, and spur, there is a significant improvement in the improved model, with AP values increasing by 4.0%, 1.3%, and 7.5%, respectively. This verifies that the improved YOLOv8-PCB model has enhanced its ability to capture the features of minute defects on PCBs.

Table 5. Comparison of AP values for various PCB defects before and after improvement.

Model	AP%					
	missing_hole	**mouse_bite**	**open_circuit**	**short**	**spur**	**spurious_copper**
YOLOv8n	99.5	95.7	91.7	96.7	90.7	91.7
YOLOv8-PCB (ours)	99.5	93.5	95.7	98.0	98.2	91.6

4.6. Visualization Comparison

4.6.1. Comparison of Recall Rate and mAP@0.5

The comparison of the recall rate and mAP@0.5 curves for the models prior to and following optimization is shown in Figure 8. The blue solid line denotes the metric curve of the YOLOv8n baseline model, while the orange solid line corresponds to the metric curve of the improved YOLOv8-PCB model. As can be seen from Figure 8, both the recall rate and mAP@0.5 of the improved model have seen improvements over the previous model. Additionally, the convergence speed of the improved model has accelerated, enhancing detection precision.

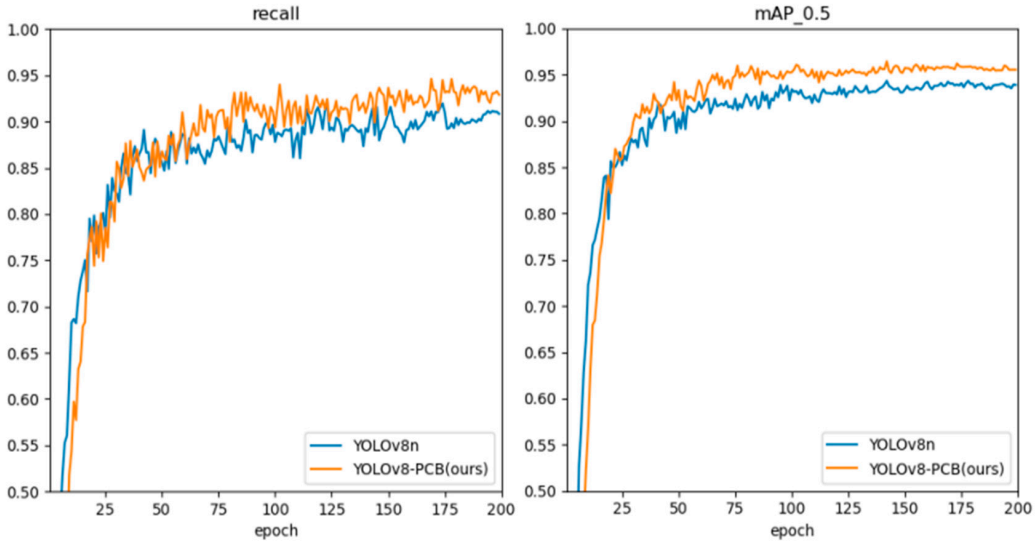
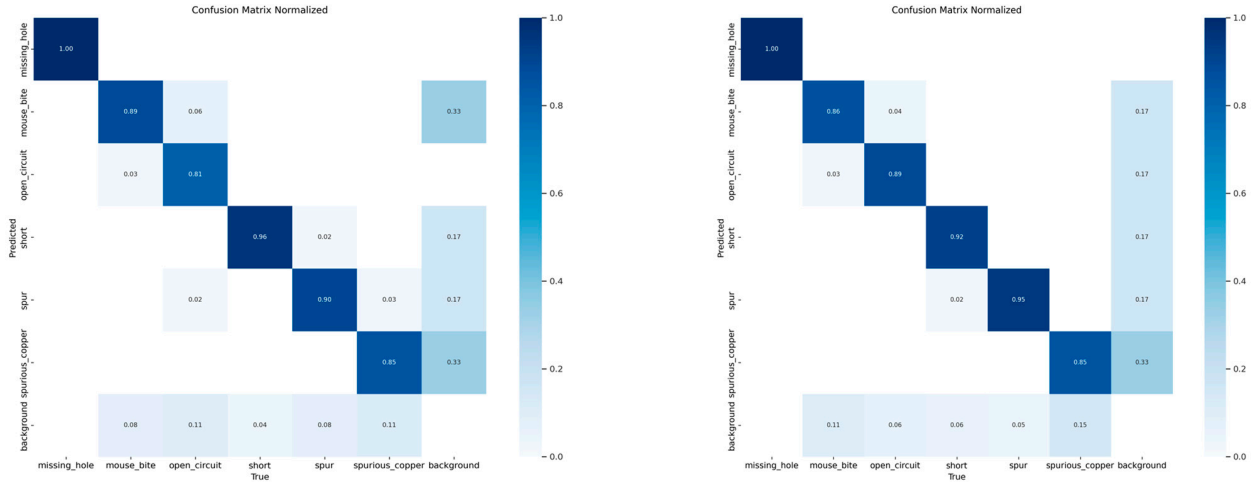


Figure 8. Comparison of the recall rate and mAP@0.5 value curves for the model before and after the improvement in PCB defect detection.

4.6.2. Comparison of Confusion Matrices

The confusion matrix exhibits good symmetry, making it convenient to observe the characteristics of the model in certain aspects. Therefore, this paper selects the confusion matrix to verify the superior performance of the improved YOLOv8-PCB model.

Figure 9 presents a comparison of the confusion matrices for the models prior to and following enhancement. The diagonal elements indicate the count of samples accurately predicted by the model, whereas the off-diagonal elements show the quantity of samples that were either incorrectly detected or missed by the model. From Figure 9a, it can be observed that the miss rates for mouse bite, open circuits, and spurious copper are relatively high, with lower accuracy in comparison with other defects. In Figure 9b, the values on the diagonals for open circuit and spur are higher than those in Figure 9a, with other defects being equal or showing insignificant differences. This indicates that the overall performance of the proposed YOLOv8-PCB model is superior to the YOLOv8n baseline model, and there is a noticeable improvement in the miss rate.



（a）Confusion matrix of YOLOv8n （b）Confusion matrix of YOLOv8-PCB

Figure 9. Comparison of confusion matrix.

4.6.3. Analysis of F1 Score for the Improved Model

As seen in Figure 10, the F1 curve has a wide confidence interval and reaches a top value of 0.94, demonstrating that the model can effectively balance precision and recall, fully confirming the model's excellence.

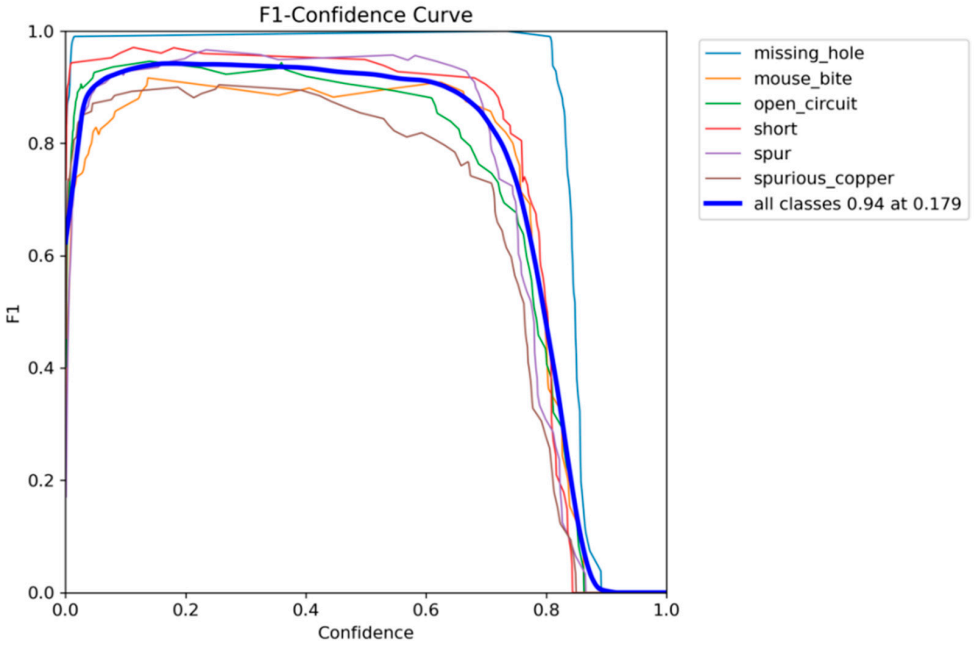


Figure 10. F1 Score Curve.

4.7. Comparative Experiments

To verify the advantages and efficacy of the YOLOv8-PCB algorithm put forward in this paper, we carried out comparative experiments using P, R, mAP@0.5, F1, Params, GFLOPs, and weight size as assessment metrics. The proposed algorithm, YOLOv8-PCB, was compared with SSD, RTDETR, Centernet, Faster RCNN, YOLOv3, YOLOv5s, YOLOv6, YOLOv7, YOLOv8n, YOLOv9s, and YOLOv10n on the same configuration, parameters, and dataset. The results of the experiments are shown in Table 6. According to Table 6, the SSD model, due to its algorithmic characteristics, exhibits weaker performance in detecting small targets, with an mAP@0.5 of only 43.75%, and it has an excessively large model size. The improved YOLOv8-PCB algorithm, when compared to SSD, Centernet, Faster RCNN, YOLOv3, YOLOv5s, YOLOv6, YOLOv8n, YOLOv9s, and YOLOv10n, achieves an increase in mAP@0.5 by 52.35%, 1.41%, 25%, 0.6%, 0.5%, 2.1%, 1.8%, 0.3%, and 1.8%, respectively. Additionally, compared to the YOLOv8n baseline model, YOLOv8-PCB improves the mean Average Precision by 1.8%, enhances the F1 score by 1.9%, cuts down the parameter count by 18.27%, decreases the computational load by 12.35%, and shrinks the model weights by 17.46%.

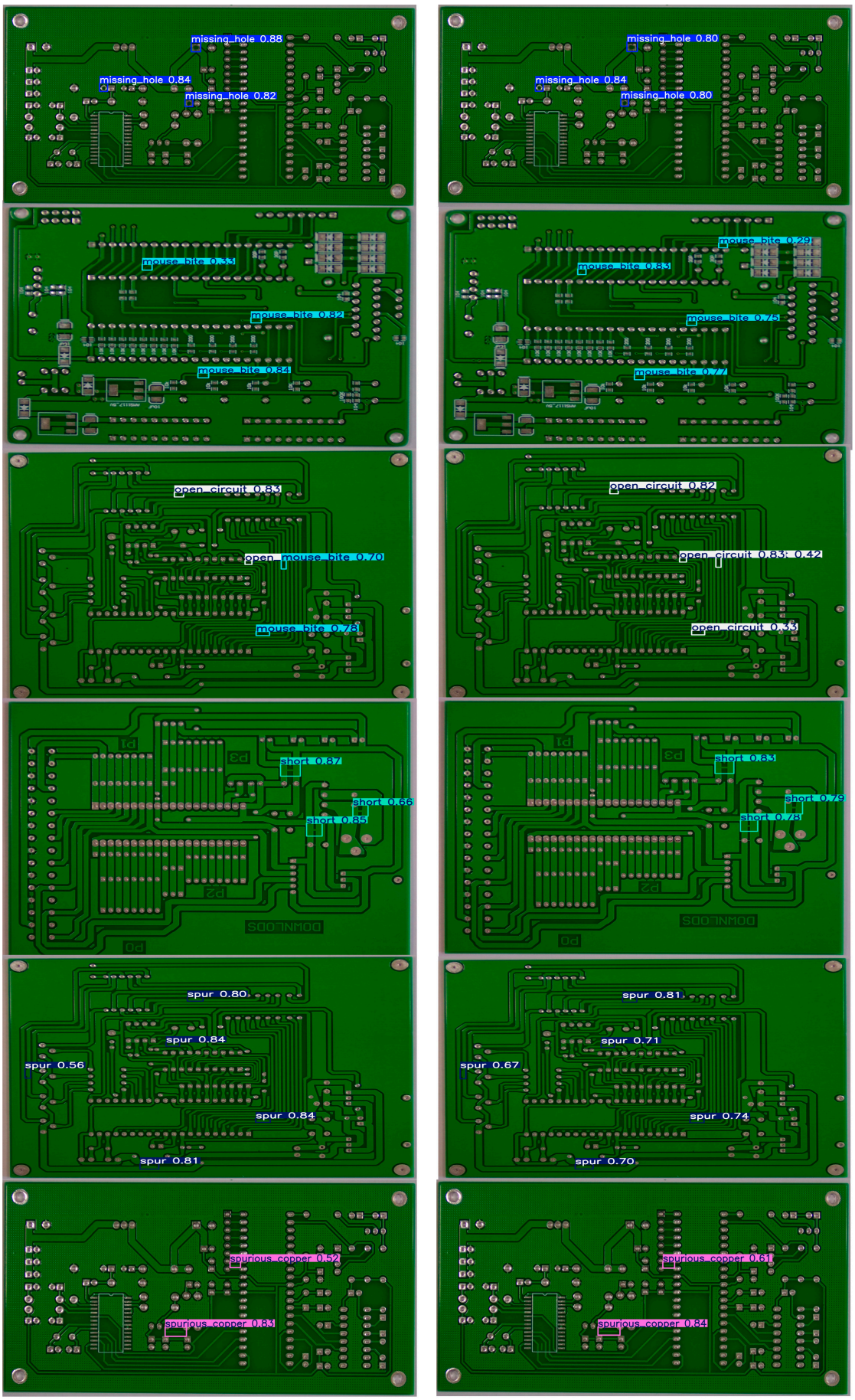
Table 6. Comparison of experimental results.

Model	P/%	R/%	Map @0.5/%	F1	Params/M	GFLOPs	Weight Size /MB
SSD	66.67	1.38	43.75	2.70	24	61.25	97.68
RTDETR	96.80	94.60	96.50	95.69	19.88	57	40.5
Centernet [25]	-	-	94.69	-	32.7	70.2	-
Faster RCNN [26]	61.40	75.60	71.10	67.80	-	273.4	-
YOLOv3	96.40	93.30	95.50	94.82	61.55	155.3	123.5
YOLOv5s	95.60	93.50	95.60	94.54	9.11	23.8	18.5
YOLOv6	93.40	89.10	94.00	91.20	4.23	11.8	8.7
YOLOv7	95.20	93.30	96.30	94.24	37.22	105.2	74.9
YOLOv8n	**94.60**	**90.40**	**94.30**	**92.45**	**3.01**	**8.1**	**6.3**
YOLOv9s	96.00	92.20	95.80	94.06	7.17	26.7	15.2
YOLOv10n	96.80	88.60	94.30	92.52	2.27	6.5	5.8
YOLOv8-PCB (ours)	**94.70**	**94.00**	**96.10**	**94.35**	**2.46**	**7.1**	**5.2**

In summary, compared to other models, the improved YOLOv8-PCB algorithm proposed in this paper demonstrates significant overall performance, achieving a balance between detection accuracy and speed, which verifies the effectiveness of the model. Further experiments indicate that the improved YOLOv8-PCB algorithm, which is the focus of this study, succeeds in reducing the model size while substantially improving detection precision and speed, thereby well meeting the practical detection requirements of industrial scenes.

4.8. Demonstration of Experimental Results

To graphically show the detection capability of the algorithm presented in this paper for various defects on PCB surfaces, a contrast of PCB defect detection is carried out between the YOLOv8n model and the YOLOv8-PCB model put forward in this research. The detection outcomes of the YOLOv8n model are illustrated in Figure 11a, and the detection results of our YOLOv8-PCB model are presented in Figure 11b. From Figure 11, it is noticeable that the YOLOv8n model failed to detect mouse-bite defects and mistakenly identified open circuit as mouse-bite defects, while our YOLOv8-PCB algorithm detected all defects correctly. The detection capabilities of the two algorithms for other defects are similar, but our YOLOv8-PCB algorithm still maintains high detection performance after being lightweighted.



(a) YOLOv8n model detection effect

(b) YOLO-PCB model detection effect

Figure 11. Comparison of outcomes for PCB defect detection.

4.9. Discussion of Experimental Results

In comparison to the YOLOv8n baseline model, this study has achieved certain advancements. On the PCB dataset, our proposed YOLOv8-PCB improved model has demonstrated significant enhancements in Recall, mAP, and F1 score, while substantially decreasing the model's parameters, computational complexity, and model weights. This fully satisfies the strict efficiency and real-time performance demands of PCB defect detection tasks, rendering it more appropriate for deployment and operation on edge equipment. However, it should be noted that this work is limited by the relatively small size of the original dataset and the idealized PCB image acquisition environment. As a result, the improved model's adaptability in complex environments may not be optimal. Therefore, future research should both broaden the dataset and endeavor to improve the model's adaptability and robustness in such environments.

5. Conclusions

This study presents an optimized algorithm for small object defect detection on printed circuit boards (PCBs), aiming to address prevalent challenges in current PCB defect detection processes, including inadequate accuracy, low detection efficiency, and limited adaptability. Specifically, we innovatively integrate the C2f_SHSA attention mechanism into the backbone network of YOLOv8n, achieving efficient and lightweight integration of local and global features, thereby improving the model's capability to recognize small-scale defects. Furthermore, we introduce the C2f_IdentityFormer structure in the neck network, which combines the advantages of the C2f structure and the IdentityFormer structure, replacing the original C2f structure. This further strengthens the model's capability to capture subtle features, optimizes the feature fusion strategy, and enhances the precision of PCB surface defect detection. Additionally, we select the PIoU as the loss function for the improved model to enhance its adaptability and convergence speed when dealing with small, complex, and morphologically diverse PCB defects. Experimental outcomes illustrate that, when compared to the YOLOv8n, the improved YOLOv8-PCB algorithm achieves a Recall rate of up to 94.0%, an mAP of 96.1%, and an F1 score of 94.35% on the PCB dataset, with a model size of only 5.2 MB, demonstrating a lightweight design. The improved algorithm achieves improvements of 3.6%, 1.8%, and 1.9% in Recall rate, mAP, and F1 score, respectively, while reducing the count of parameters by 18.27%, computational complexity by 12.35%, and model weight by 17.46%. It satisfies the strict criteria of PCB surface defect detection tasks for efficiency and real-time performance, exhibiting unique advantages in model lightweighting and deployment flexibility, and is more fitting for deployment and operation on edge devices. Consequently, the algorithm put forward in this paper not only offers an efficient, reliable, and adaptable solution for the field of PCB surface defect recognition but also offers powerful technical support for intelligent production in smart factories.

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

Funding: This research was supported by the National Natural Science Foundation of China (No. 62162026), Jiangxi Provincial Natural Science Foundation (No. 20242BAB26019), the Special Project for Cultural Research of Henan Xing Culture Engineering (No. 2022XWH115), and the Anyang City Science and Technology Research Project (No. 2023C01GX019).

Data Availability Statement: The address of the dataset is https://robotics.pkusz.edu.cn, accessed on 10 August 2024.

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

References

1. Peng, H.; Zhou, B.; Ouyang, W. Research on defect detection of lightweight PCB based on dual channel attention. *J. Optoelectron. Laser* **2024**, *35*, 506–515.
2. Tang, S.; He, F.; Huang, X. Online PCB Defect Detector on A New PCB Defect Dataset. *arXiv* **2019**, arXiv:1902.06197.
3. Chaudhary, V.; Dave, I.; Upla, K. Automatic visual inspection of printed circuit board for defect detection and classification. In Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 22–24 March 2017; IEEE: Piscataway, NJ, USA; pp. 732–737.
4. Li, J. Research on PCB defect detection and classification based on machine learning. *Print. Circuit Inf.* **2024**, *32*, 57–59.
5. Cheng, S.; Yang, H.; Xu, X. Improved LightweightX-Ray Aluminum Alloy Weld Defects Detection Algorithm Based on YOLOv5. *Chin. J. Lasers* **2022**, *49*, 136–144.
6. Liu, L.; Ye, Y.; Guo, T. Detection Method for OLED Pixel Defects Based on Extended Feature Pyramid. *Acta Opt. Sin.* **2023**, *43*, 115–123.
7. Hao, B.; Fan, Y.; Song, Z. Deep Transfer Learning-Based Pulsed Eddy Current Thermography for Crack Defect Detection. *Acta Opt. Sin.* **2023**, *43*, 146–154.
8. Ding, R.; Dai, L.; Li, G. TDD-net: A tiny defect detection network for printed circuit boards. *CAAI Trans. Intell. Technol.* **2019**, *4*, 110–116. [CrossRef]
9. Shi, W.; Lu, Z.; Wu, W. Single-shot detector with enriched semantics for PCB tiny defect detection. *J. Eng.* **2020**, *2020*, 366–372. [CrossRef]
10. Lan, Z.; Hong, Y.; Li, Y. An improved YOLOv3 method for PCB surface defect detection. In Proceedings of the 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), Shenyang, China, 22–24 January 2021; IEEE: Piscataway, NJ, USA; pp. 1009–1015.
11. Wu, J.; Cheng, Y.; Shao, J. A defect detection method for PCB based on the improved YOLOv4. *Chin. J. Sci. Instrum.* **2021**, *42*, 171–178.
12. Wu, L.; Zhang, L.; Zhou, Q. Printed Circuit Board Quality Detection Method Integrating Lightweight Network and Dual Attention Mechanism. *IEEE Access* **2022**, *10*, 87617–87629. [CrossRef]
13. Zhao, Y.; Kong, X.; Ma, C. Real-time PCB Fault Detection Algorithm Based on Darknet Network and YOLO4 Algorithm. *Comput. Meas. Control* **2023**, *31*, 101–108.
14. Lan, H.; Zhu, H.; Luo, R.; Ren, Q.; Chen, C. PCB Defect Detection Algorithm of Improved YOLOv8. In Proceedings of the 2023 8th International Conference on Image, Vision and Computing (ICIVC), Dalian, China, 27–29 July 2023; pp. 178–183.
15. Pan, W.; Wei, C.; Qian, C. Improved YOLOv8s Model for Small Object Detection from Perspective of Drones. *Comput. Eng. Appl.* **2024**, *60*, 142–150.
16. Shen, L.; Lang, B.; Song, Z. DS-YOLOv8-Based Object Detection Method for Remote Sensing Images. *IEEE Access* **2023**, *11*, 125122–125137. [CrossRef]
17. Yun, S.; Ro, Y. Shvit: Single-head vision transformer with memory efficient macro design. In Proceedings of the IEEE CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 5756–5767.
18. Tan, H.; Liu, X.; Yin, B. MHSA-Net: Multihead self-attention network for occluded person re-identification. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *34*, 8210–8224. [CrossRef] [PubMed]
19. Weihao, Y.; Chenyang, S.; Pan, Z. MetaFormer Baselines for Vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 896–912.
20. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
21. Liu, C.; Wang, K.; Li, Q. Powerful-IoU: More straightforward and faster bounding box regression loss with a nonmonotonic focusing mechanism. *Neural Netw.* **2024**, *170*, 276–284. [CrossRef]
22. Xu, S.; Li, Y.; Liang, Q. Bare PCB Defect Detection Based on Improved YOLOv5 Algorithm. *Packag. Eng.* **2022**, *43*, 33–41.
23. Li, D.; Xu, A.; Wang, S. Printed circuit board defect detection based on improved YOLOv5. *Electron. Meas. Technol.* **2023**, *46*, 112–119.
24. Zhou, X.; Zhu, J.; Wang, Y. PCB Bare Board Defect Detection Based on Improved YOLOv7 Algorithm. *Radio. Eng.* **2023**, *53*, 2791–2797.

25. Wang, Y.; Luo, J.; Tao, J. Defect detection of printed circuit boards based on YOLOv8-PCB. *Laser Optoelectron. Prog.* **2025**, *62*, 0412002.
26. Li, J.; Gong, H. DCW-YOLOv8 Detection Algorithm for PCB Electronic Components. *J. Chin. Comput. Syst.* **2024**, *45*, 2479–2484.

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.