Journal of Computer and Communications, 2025, 13(8), 17-39
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

BCN-YOLO: A Deep Learning Network for PCB Defect Detection

Junjie Liu[1], Shuxin Yao[1], Boxiong Li[1*], Jianqing Liu[2]

[1]College of Mechanical and Electronic Engineering, Dalian Minzu University, Dalian, China
[2]R&D Department, Dalian Rijia Electronics Co., Ltd., Dalian, China
Email: *920945947@qq.com

How to cite this paper: Liu, J.J., Yao, S.X., Li, B.X. and Liu, J.Q. (2025) BCN-YOLO: A Deep Learning Network for PCB Defect Detection. *Journal of Computer and Communications*, **13**, 17-39.
https://doi.org/10.4236/jcc.2025.138002

Received: July 10, 2025
Accepted: August 4, 2025
Published: August 7, 2025

Abstract

To address the issues of missed detection and false detection during the defect inspection process of the PCB, an improved YOLOv7-based algorithm for PCB defect detection is proposed. Firstly, the Bi-Former attention mechanism and CARAFE upsampling operator are introduced into the original YOLOv7 backbone network to achieve more flexible computation allocation and content awareness, enabling the network to dynamically perceive sparsity in queries. Secondly, a powerful feature pyramid network, CMPANet, is proposed to extract more shallow features, effectively improving the model's detection performance on small targets. Finally, the NWD loss function is introduced to optimize the regression loss function in combination with IoU, reducing sensitivity to position deviations of small targets. Experimental results demonstrate that the modified YOLOv7 achieves a mAP@0.5 value of 95.25%. Compared to the original model, the mAP@0.5 and mAP@0.5:0.9 values are improved by 3.32% and 2.86%, respectively, while the F1 score is enhanced by 3.91%. The detection speed is 44.84 FPS. These improvements effectively enhance the accuracy of detecting small target defects on PCBs. Additionally, the performance on AI-TOD, Tiny-Person, and Wider-Person small target datasets shows improvements over the original network.

Keywords

PCB Bare Board, YOLOv7, Small Object Detection, Loss Function, Attention Mechanism

1. Introduction

Owing to breakthroughs and innovations in semiconductor technology, a solid foundation has been provided for modern electronic products and the field of information technology. The densification and miniaturization of bare Printed Cir-

cuit Boards (PCBs) represent key aspects of semiconductor technology development, with their quality directly impacting the overall quality and performance of electronic products [1]. However, during the manufacturing process of bare PCBs, defects such as open circuits, short circuits, copper deficiency, and poor via pad placements often occur due to limitations in manufacturing techniques and materials. These defects can severely affect the quality and stability of the bare PCBs and may lead to substantial losses amounting to tens of thousands. Therefore, defect inspection before the dispatch of finished printed circuit boards is an indispensable quality control task.

In recent years, Convolutional Neural Networks (CNNs) have achieved substantial breakthroughs in the field of computer vision. Detection algorithms based on CNNs have successfully performed in conventional object detection tasks, with the mainstream neural network object detection algorithms divided into one-stage and two-stage detections. The one-stage algorithms are particularly advantageous in terms of detection speed, which is critical for industrial applications [2] such as PCB bare board detection, making them highly suitable for such tasks. The YOLO series [3]-[6] predominantly represents these one-stage algorithms, and researchers have continually refined these models over the years. For instance, Li [7] *et al.* enhanced the YOLOX algorithm for PCB defect detection by integrating the ECANet attention mechanism, improving the feature extraction rate for PCB defects and achieving a 1.21% increase in the mean mAP compared to the original network, though at the cost of reduced detection speed. Su [8] *et al.* modified the YOLOv4 model by improving the PANet network structure and employing the H-swish activation function alongside an attention mechanism, which increased the FPS by 2.24%. Wang [9] *et al.* proposed a lightweight YOLOv5 model, replacing YOLOv5's backbone with EfficientNetV2 to reduce computational parameters, enhancing the FPS by 5.30%. Tuo [10] *et al.* developed a PCB defect detection algorithm based on YOLOX-WSC, adding the parameter-free attention mechanism SimAM and replacing the CSPLayer structure with CSPHB modules, resulting in a 2.88% improvement over YOLOX, although further enhancements in detection speed are needed.

Although the aforementioned algorithms have improved performance to some extent, they still face several challenges, such as low accuracy in detecting small target defects on PCBs, high parameter counts, and slow detection speeds. PCB bare board defects are typically characterized by small size, low contrast, and high density, imposing greater demands on object detection algorithms and necessitating improvements in small target detection capability. To address these issues, this paper proposes four improvements to defect detection of small targets on PCB bare boards based on the YOLOv7 algorithm:

1. Integration of Bi-Former, a sparse-sampling dual-channel attention mechanism, onto the feature extraction network to enhance the efficiency of model training and inference, thereby obtaining more crucial information.

2. The introduction of the lightweight upsampling operator CARAFE enables

the enhanced network model to aggregate contextual information within larger receptive fields, thereby improving the detection performance of the model on small target defects.

3. A loss function named NWD, combined with Intersection over Union (IoU), has been introduced. This integration does not increase the additional parameters of the model, but reduces the sensitivity of small targets to the prediction bounding boxes. It enhances the precision of small object detection and accelerates the model's convergence speed, thereby improving overall detection performance.

4. A powerful feature pyramid network, named CMPANet, based on the PANet architecture, has been proposed to extract more shallow features and enhance the feature fusion capability of the model.

Based on the aforementioned improvements, the modified YOLO model in this study achieved a mAP@0.5 value of 95.25%. Compared to the original model, the mAP@0.5 value and mAP@0.5:0.9 value increased by 3.32% and 2.86% respectively. Additionally, the F1 score improved by 3.91%, while maintaining a detection rate of 44.84 FPS. These results demonstrate the superiority of the improved algorithm in detecting small target defects on PCB boards.

In Section 1, we reviewed related work. The details and improvements of TD-YOLO are discussed in Sections 2 and 3. Experimental results and discussions are presented in Section 4. Finally, Section 5 provides a summary of this paper.

2. BCN-YOLO Network Structure

The YOLOv7 network model primarily comprises three components: the input layer (Input), the backbone network (Backbone), and the detection head layer (Neck & Head). Initially, the model extracts features from image data; subsequently, these features undergo fusion in the Neck module to produce features of three different scales: large, medium, and small. Finally, these fused features are fed into the detection head, which processes and outputs the detection results.

Compared to previous YOLO architectures, the backbone network of YOLOv7 introduces three new modules: E-ELAN (Extended-ELAN), MPConv, and SPPCSPC. The E-ELAN module enhances the network's learning capabilities on the basis of the original ELAN module without disrupting the existing gradient paths, enabling more effective learning and convergence. The MPConv module expands the receptive field of the feature layer through MaxPool operations and then fuses the expanded feature information with the normally convoluted feature information, improving the network's generalization capabilities. The SPPCSPC module incorporates multiple MaxPool operations in parallel within a series of convolutions to avoid image distortion and other issues related to image processing operations. This module also addresses the issue of redundant feature extraction in convolutional neural networks. This paper presents a PCB bare board defect detection model based on YOLOv7, named BCN-YOLO, with improvements highlighted in red. The structure of this model is shown in **Figure 1**.

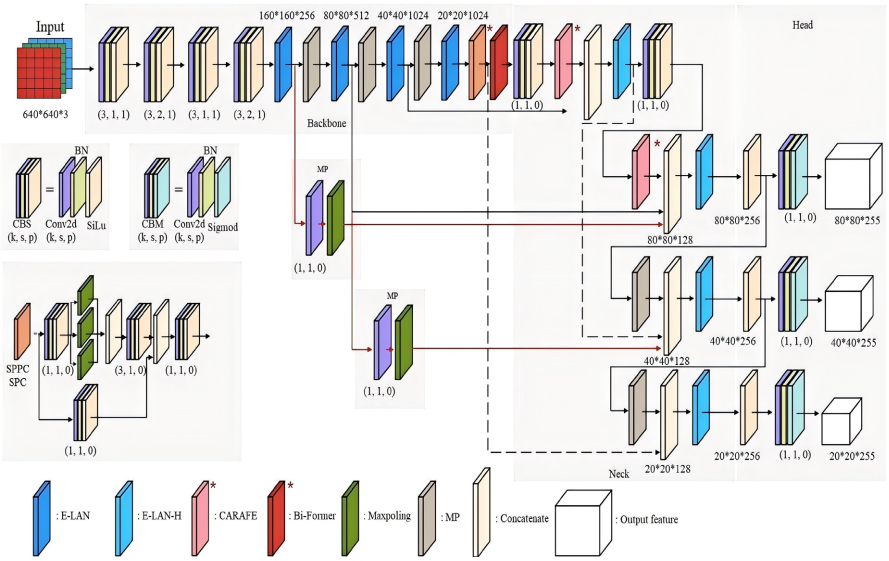


Figure 1. BCN-YOLO network architecture.

The enhancements include three key parts: Firstly, the feature extraction network integrates a dual-channel attention mechanism based on sparse sampling, named Bi-Former, which improves both training and inference efficiency. Secondly, the upsampling module is replaced with CARAFE, which enables better prediction of PCB defects without introducing an excessive number of parameters. Thirdly, to enhance the feature extraction capability for small targets, an enhanced PANet named CMPANet is designed on top of the original three-layer detection head to boost the extraction of shallow features of small targets. In addition, the commonly used IoU metric is replaced with NWD to reduce the sensitivity of small targets to the prediction boxes without increasing the model's additional parameters. Detailed descriptions of these improvements are provided in subsequent sections.

3. Methodology

3.1. Introducing Attention Modules

The Transformer [11] model leverages the self-attention mechanism to enhance its capability to capture long-range dependencies, making it widely applicable in the field of object detection. However, this structure inevitably presents two major challenges: substantial memory usage and high computational costs. To address these issues, researchers have introduced a variety of manually designed sparse attention patterns to reduce the model's complexity. Although these methods alleviate computational pressure to some extent, they still have limitations in capturing comprehensive long-range relationships. To overcome this, a dynamic sparse attention method called Bi-level Rounding Attention (BRA) has been introduced. The architectural structure of the BRA model is depicted in **Figure 2**. This mechanism aims to balance computational efficiency with the ability to effectively capture long-distance interactions

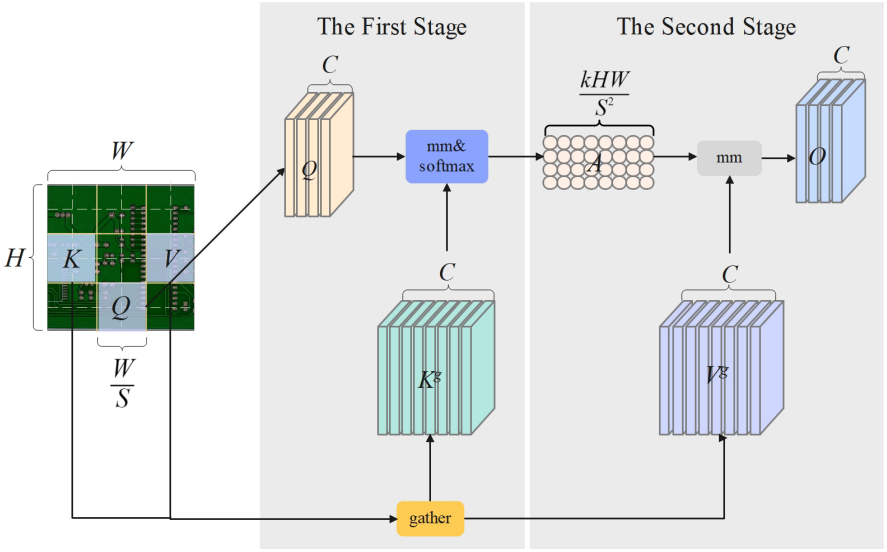


Figure 2. BRA operation diagram.

The core design principle of the BRA module lies in its utilization of coarse-grained regional-level preselection to filter out irrelevant key-value pairs, followed by the application of a refined token-to-token attention mechanism in the remaining candidate regions (*i.e.*, routing areas). This strategy not only endows the model with adaptability but also significantly enhances computational efficiency and substantially reduces memory usage. In **Figure 2**, the given intermediate feature map $X \in \mathbb{R}^{H\times W\times C}$ serves as input and is divided into $S\times S$ non-overlapping regions, each containing $\frac{HW}{S^2}$ feature vectors. The BRA module undergoes two sequential stages of attention inference: in the first stage, the feature map of the PCB for testing is divided into multiple coarse-grained blocks, and self-attention operations are conducted on these blocks. Q,K,V matrices are obtained through linear mapping. Then, based on the key-value pairs $Q,K\in\mathbb{R}^{S^2\times\frac{HW}{S^2}\times C}$, the relevance between each pair of coarse-grained blocks is calculated, resulting in a coarse-grained sparse matrix A^r, which helps estimate the semantic association between the two regions. Subsequently, only the top k most relevant adjacency matrices are retained, yielding the routing index matrix I^r:

$$I^r = topkIndex\left(A^r\right) \tag{1.1}$$

In the second stage, based on the routing index matrix $I_r\in\mathbb{N}^{S^2\times k}$ obtained from the first stage, further fine-grained self-attention is performed. Utilizing sparsity operations, computations of the least relevant areas are directly skipped to save on the number of parameters and computational resources, thereby enhancing the model's detection speed for the input PCB feature map. Ultimately, to effectively extract the contour features of the detection targets and capture the main content of these targets, the attention is focused on the aggregated key-value pairs:

$$O = Attention\left(Q, K^g, V^g\right) + LCE\left(V\right) \tag{1.2}$$

In Eqation.2, $K^g, V^g \in R^{S^2 \times \frac{KHW}{S^2} \times C}$ are clustered key-value pairs, and we introduce a local context enhancement function, parameterized with deep convolutions, to apply fine-grained T2T (token to token) attention.

To address the issue of excessive model parameters, a Bi-Former attention mechanism centered around BRA was proposed [12]. The detailed structure is shown in **Figure 3**. The core module of each stage is Bi-Former block, and its standardized structure is 3 × 3 deep convolution (position encoding) → BRA module (dynamic routing attention) → double-layer MLP (feature enhancement), which effectively reduces the number of model parameters and significantly improves the lightweight performance.

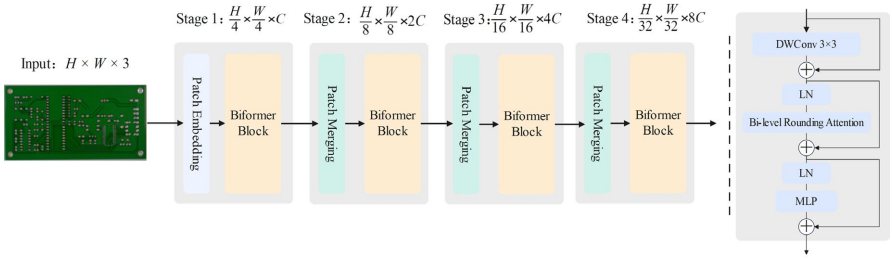


Figure 3. Bi-Former structure diagram.

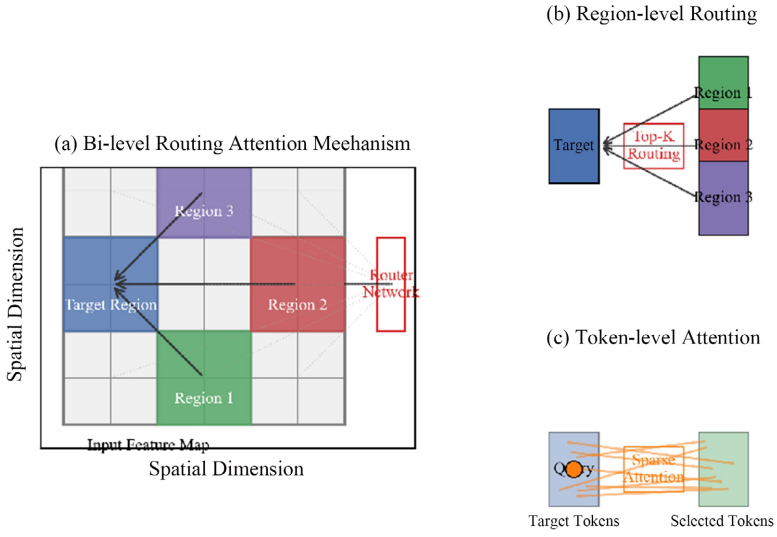


Fig. 1: Bi-Foemer attention mechanism: (a) Feature map is partitioned into non-overlapping regions: A routing network selects Top-K rel

Figure 4. Bi-level routing attention mechanism.

Figure 4 below shows the Bi-level Routing Attention Mechanism, The Bi-Former attention mechanism achieves efficient spatial sparse attention computation through two-level routing. Firstly, the feature map is divided into non-overlap-

ping regions (panel a), and the Top-K relevant source regions (green/red/purple) of each target region (blue) are dynamically screened through a lightweight routing network. Subsequently, a fine-grained token-level sparse attention computation (Panel c) is performed only within the range selected by the region-level routing (panel b), where query tokens (orange) within the target region only establish connections with tokens from the selected source region. This hierarchical sparse strategy reduces the computational complexity from O (N^2) to O (N √ N) of traditional attention, and significantly improves the processing efficiency of high-resolution images while maintaining the modeling ability.

3.2. Introducing CARAFE Up-Sampling Operator

Upsampling is a commonly used technique in image processing and computer vision that increases the resolution of an image or feature map, thereby enhancing the detail or feature representation capability of the image. The most widely used upsampling methods are bilinear interpolation and nearest neighbor interpolation. However, these methods simply perform a weighted average of adjacent pixels or use the values of nearby pixels, without utilizing the semantic information of the feature map. Another method involves the use of transposed convolution, which expands the feature map by inserting zero elements into the convolutional kernel and applying convolution operations. However, larger convolution kernels can result in pixel values being influenced by input pixels far from the target location, and the excessive number of parameters can waste computational resources. Therefore, under the premise of ensuring lightweight operation, the upsampling operator CARAFE [13] has been introduced. It possesses a larger receptive field and is able to effectively extract semantic information from the feature map. The network structure of the CARAFE upsampling operator is illustrated in **Figure 5**.

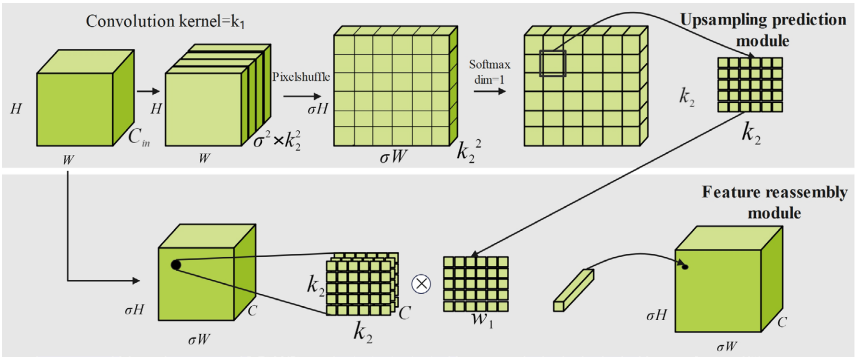


Figure 5. CARAFE operation diagram.

It consists of a prediction module and a feature reassembly module. For a feature map of size, the prediction module estimates the upsampling kernel, and the feature reassembly module completes the upsampling process to generate a feature map of size. In the prediction module, channel compression is achieved using 1x1 convolutions to reduce computational costs. Subsequently, a convolutional

layer with a kernel size of is used to predict the upsampling kernel, and the upsampling kernel size is set to. Softmax operation is then applied to normalize the weights of the convolutional kernel to ensure a sum of 1. For each position in the output feature map, the feature reassembly module maps it back to the input feature map, extracts an region centered at that position, and performs dot product operations with the predicted upsampling kernel obtained at that position, yielding the final output feature value.

By replacing the original nearest neighbor interpolation upsampling in YOLOv7 with the CARAFE upsampling operator, the network is able to better predict PCB defect targets while introducing only a small number of additional parameters.

3.3. CMPANet for Feature Fusion

Deep feature maps provide more semantic information, while shallow feature maps provide more spatial information, which is crucial for detecting small objects. Building upon the multi-scale detection algorithm of YOLOv7, the Neck layer network is further deepened to enhance the feature extraction capability for small objects. Reference [14] suggests that investing more computation into the feature fusion network helps address scale variance issues. However, previous works have only involved intra-level fusion [15] or fusion with the preceding level. CMPANet proposed in reference [16] is a novel cross-scale fusion method that integrates features from the same layer and adjacent layers. The optimized feature pyramid network structure is illustrated in **Figure 6**.

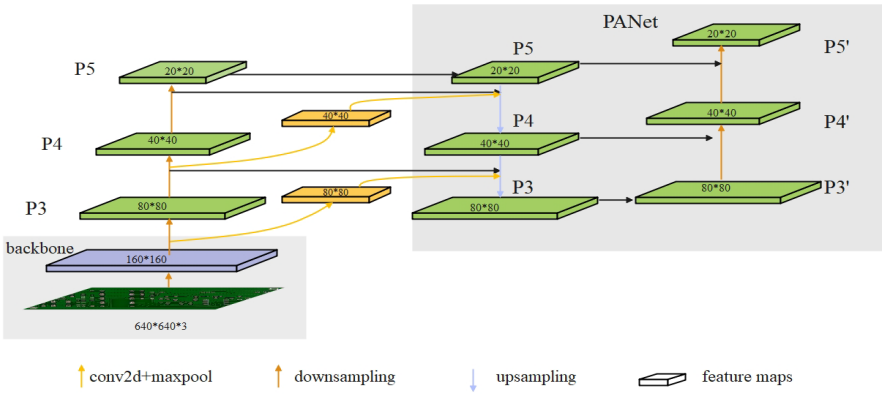


Figure 6. The architecture of proposed CMPANet.

Considering the complexity and scale of the model, this paper simplified the fusion strategy proposed in reference [15]. Building upon the original three-layer detection head network, only the additional features of P4 are concatenated with P3 responsible for small object output, and the additional features of P5 are concatenated with P4. Thus, the extra fusion for outputs P3' and P4' provides richer spatial information, aiding in better localization of small defects in the dataset and expanding the network's field of view for detecting targets. Additionally, the introduction of convolution and max-pooling operations has almost negligible im-

pact on computational cost growth. Through the new feature fusion network, better detection accuracy can be achieved without increasing computational cost and inference time, particularly for spur defects, including the smallest defects in the dataset.

3.4. Introducing NWD Loss Function

For small target defects on PCB bare boards, matching high-quality prior boxes to ground truth is particularly challenging. A simple approach is to lower the IoU threshold, which increases the number of positive samples for small targets but generally reduces overall quality. Current research aims to make label assignment more adaptive, such as Anchor-based and Anchor-free Detection via Adaptive Training Sample Selection (ATSS) [17], which automatically calculates thresholds, and approaches like Probabilistic Anchor Assignment (PAA) [18] and OTA [19]. However, these methods are still based on improvements to IoU, which are not well-suited for small target defects on PCB bare boards. Consequently, the Normalized Wasserstein Distance (NWD) loss function is introduced to mitigate the issue of small targets being overly sensitive to IoU. Figure 7 illustrates a comparison between IoU deviation curves and NWD deviation curves for objects of different sizes.

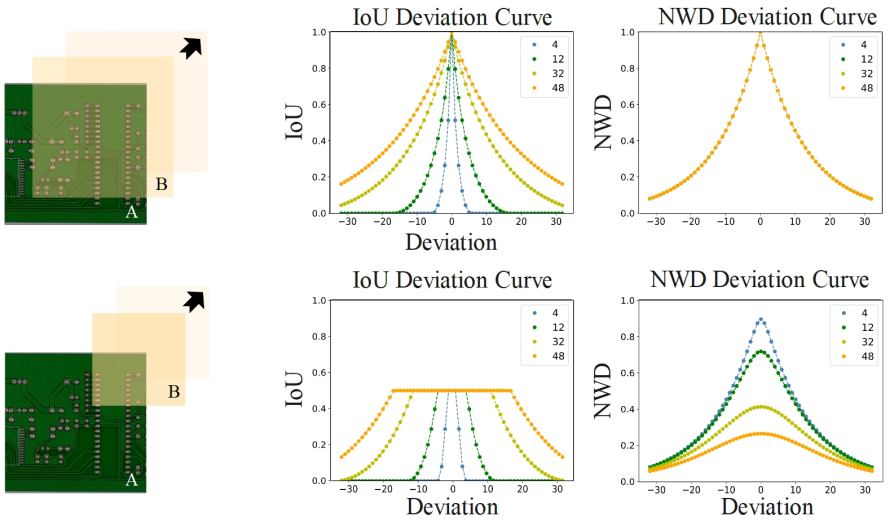


Figure 7. Comparison between IoU-Deviation Curve and NWD-Deviation Curve.

A represents the ground truth box, and B represents the predicted box. The horizontal axis represents the pixel deviation between the center points of A and B, while the vertical axis represents the corresponding metric values. Without loss of generality, we discuss the changes in the metric values under the following two scenarios. In the first row of Figure 7, keeping the predicted box A and the predicted box B at the same scale and moving B along the diagonal of A, it can be observed that the four NWD curves completely overlap, indicating that NWD is insensitive to changes in the scale of the predicted box. Additionally, it can be

observed that setting the side length of B to half that of A results in a much smoother NWD curve compared to the IoU curve. Even when $|A \cap B| = A$ or $|A \cap B| = 0$, the NWD curve consistently reflects the correlation between A and B.

The NWD introduces a new metric to calculate the similarity between the predicted box and the ground truth box. Specifically, it models the predicted box as a Gaussian distribution and then uses the Wasserstein distance [16] to measure the similarity between these two distributions, replacing the IoU. The advantage of this approach is that it can measure similarity even when there is little to no overlap between the predicted box and the ground truth box. Additionally, NWD is insensitive to the scale of the targets, making it more stable for small targets. In Figure 7, for two 2D Gaussian distributions, their second-order Wasserstein distance, after normalization, can be simplified as:

$$NWD(N_a, N_b) = \exp\left(-\frac{\sqrt{W^2(N_a, N_b)}}{C}\right) \tag{1.3}$$

In Equation 3, N_a, N_b represent the Gaussian distributions generated from the predicted box and the ground truth box, respectively. W is the metric function of $W(N_a, N_b) = \inf_{\pi \in \Gamma(N_a, N_b)} \int_{\mathbb{R}\times\mathbb{R}} |x-y| d\pi(x,y)$, and C is a constant closely related to the dataset.

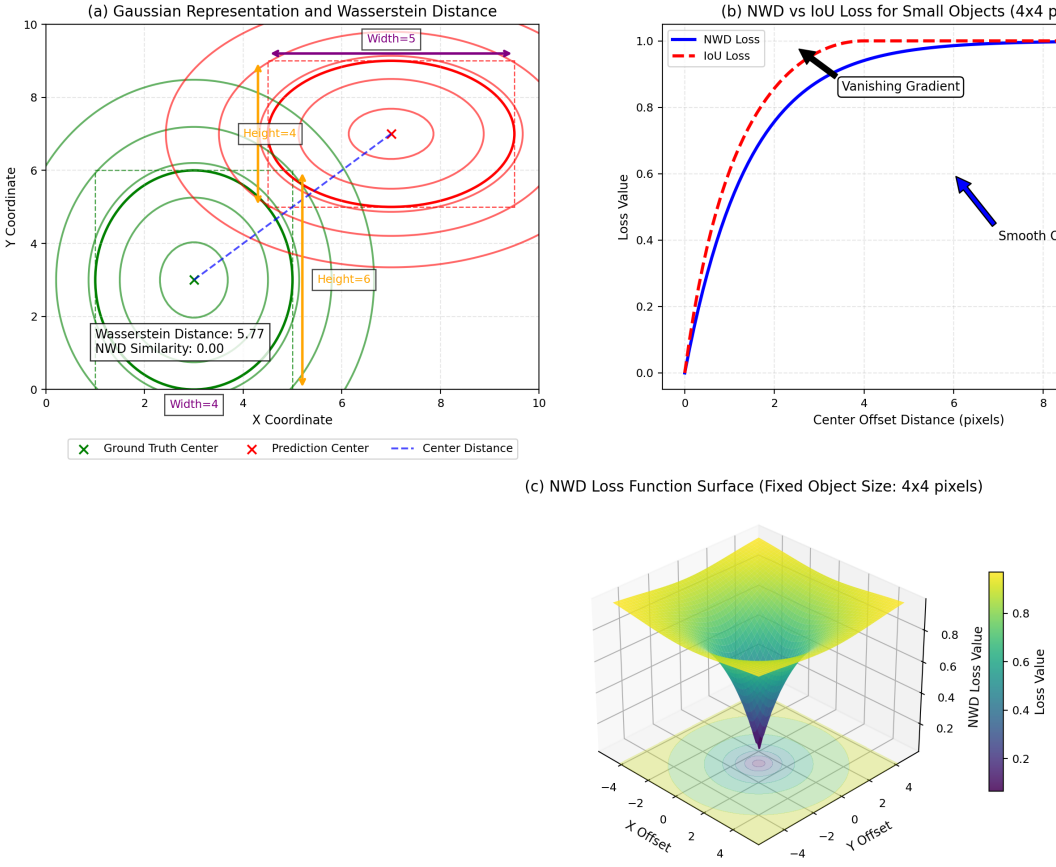


Figure 8. The mechanism and advantages of the NWD loss function.

Figure 8 presents a comprehensive schematic diagram illustrating the mechanism and advantages of the normalized Wasserstein distance (NWD) loss function: Subfigure (a) demonstrates Gaussian distribution modeling of bounding boxes, quantifying positional and dimensional differences through elliptical contour quantization (Euclidean distance between centers: 5.77, NWD similarity: 0.36). Subfigure (b) reveals that the NWD loss (blue curve) maintains non-zero gradients and smooth response characteristics even for small targets (4×4 pixels), significantly outperforming the IoU loss (red curve) in addressing gradient disappearance issues in non-overlapping regions. The 3D surface in subfigure (c) visually demonstrates the convex optimization properties of the NWD loss function, with its symmetrical and smooth topological structure providing stable convergence paths for gradient descent algorithms. This visualization comprehensively validates the core value of NWD loss in solving gradient disappearance and enhancing localization accuracy for small object detection, from theoretical modeling to functional behavior.

However, if the IoU loss function is entirely replaced with the NWD loss function, it would impact the convergence speed of the $Loss$ curve and reduce the performance of detecting medium and large targets. Therefore, a combined approach utilizing both NWD and IoU is adopted to optimize the loss function:

$$Loss = ratio * IoU + (1 - ratio) * NWD \tag{1.4}$$

In Equation 4, $ratio$ represents the weight ratio of the IoU loss function, and $Loss$ represents the loss function.

4. Experiment and Analysis

4.1. Introducing NWD Loss Function

The experimental environment consists of the Windows 11 operating system, Torch architecture, Python development language, RTX3070 graphics card for training, and CUDA and CUDNN environments for GPU acceleration. Detailed training parameters are shown in Table 1 below.

Table 1. Hardware environment and model parameters.

Parameter	Value	Parameter	Value
Epochs	200	IoU_t	0.2
Batch	4	Hsv_h	0.015
Pixel size	640	Hsv_s	0.4
Learning Rate	0.01	Hsv_v	0.2
Momentum	0.937	Optimizer	Adam
Weight decay	0.0005		

4.2. Dataset and Image Augmentation

The experiment employs a publicly available synthetic PCB dataset released by

Peking University, containing 1,386 images with a resolution of 640 × 640 pixels. The dataset simulates six types of defects (missing hole, mouse bite, open circuit, short circuit, spur, spurious copper) using a rendering procedure based on Poisson distribution layouts and procedural defect generation, including random placement and morphological operations to mimic real-world imperfections. Data augmentation techniques, such as horizontal and vertical flipping (probability 0.5), skewing (angle range ±15˚), darkening (brightness reduction 0.1 - 0.3), adding Gaussian noise (standard deviation 0.01 - 0.05), and partial occlusion (random patches covering 5% - 10% of the image), are applied to expand the dataset to 2,000 images. The dataset is split into training, validation, and test sets in an 8:1:1 ratio (1,600 training, 200 validation, 200 test) using a fixed random seed of 42 for reproducibility. Examples from the dataset are shown in Figure 9.

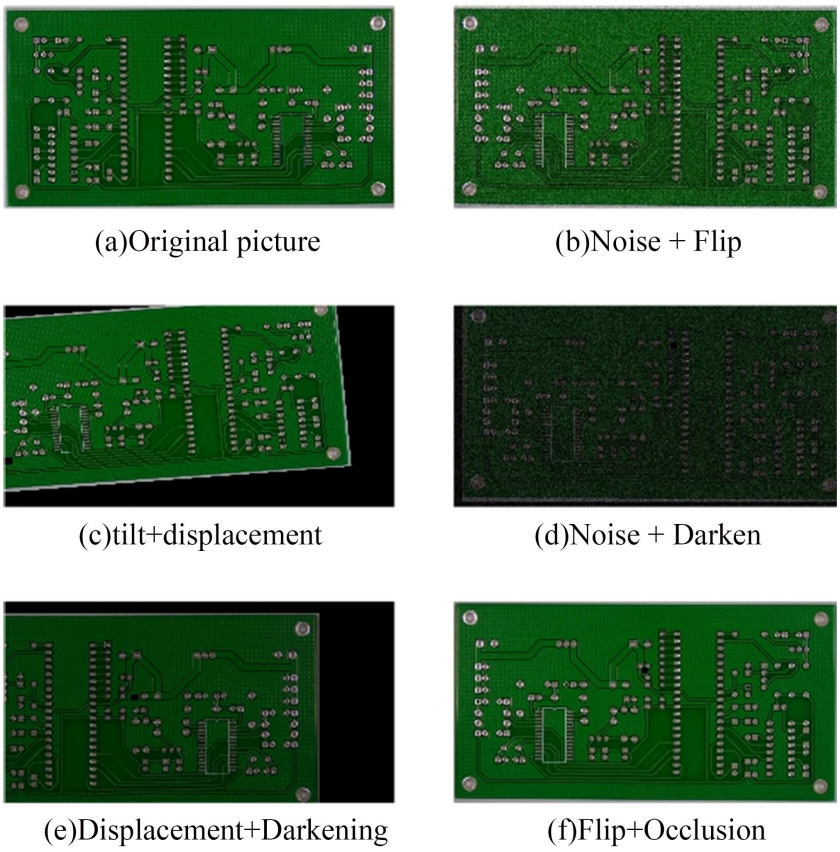


Figure 9. Image enhancement dataset.

4.3. Evaluation Indicators

To evaluate the detection capabilities of the improved network model on different types of images, experiments were conducted to compare the performance of the network before and after the improvements. The comparison was made based on the following metrics: Precision-Recall (P-R) curves, Average Precision (AP), mean Average Precision (mAP), Frame Per Second (FPS), and the F1 (F-Measure) factor. The Equation 5、6、7、8 for calculating these metrics are as follows:

$$P = \frac{TP}{TP + FP} \times 100\% \tag{1.5}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{1.6}$$

$$AP = \int_0^1 P(R) dR \tag{1.7}$$

$$F_a = \frac{\left(a^2 + 1\right) \times P \times R}{a^2 \left(P + B\right)} \tag{1.8}$$

The above Equation are used to measure the detection performance of the model for PCB bare board defects. Higher values indicate better detection performance. In object detection algorithms, FPS (Frames Per Second) is used to measure the detection speed of the algorithm, representing the number of images that can be processed per second. The calculation formula is shown in Equation 9:

$$FPS = \frac{1000ms}{PP + Inference + NMS} \tag{1.9}$$

In Equation 9, PP represents the time for image preprocessing, which includes operations such as resizing images while maintaining aspect ratio and padding, channel transformation, and dimensionality expansion. $Inference$ represents the inference speed, indicating the time from inputting preprocessed images into the model to obtaining model output results. NMS (Non-Maximum Suppression) denotes the time taken for non-maximum suppression, which primarily involves post-processing to refine the model output results. Comparing FPS requires conducting experiments under the same hardware conditions.

4.4. Attention Mechanism Experiments

To validate the effectiveness of the Bi-Former attention mechanism for small object detection, ECA, SE, CBAM, and Bi-Former attention mechanisms were added to YOLOv7 and compared using the AI-TOD dataset. The experimental results are shown in Table 2.

Table 2. Hardware environment and model parameters.

Algorithm	mAP@0.5 (%)	mAP@0.5:0.9 (%)
YOLOv7	40.02	19.63
YOLOv7 + SE	41.95	19.91
YOLOv7 + ECA	43.40	20.36
YOLOv7 + CBAM	43.89	20.61
YOLOv7 + Bi-Former	43.97	20.96

After incorporating attention mechanisms, the detection accuracy of YOLOv7 generally increased. Compared to the original network, adding the Bi-Former attention mechanism improved mAP@0.5 by 1.41% and mAP@0.5:0.9

by 3.95%. Compared to adding CBAM, mAP@0.5 increased by 0.35% with Bi-Former while mAP@0.5:0.9 remained essentially unchanged, demonstrating the significant performance enhancement provided by Bi-Former. Additionally, the performance of the algorithm with Bi-Former outperformed other attention mechanisms.

4.5. Loss Function Experiments

To determine the optimal combination of NWD and IoU and to validate the effectiveness of NWD, comparative experiments were conducted with different ratio values, as shown in Equation 4. The experimental results are detailed in Table 3.

Table 3. Effect of ratio value on detection accuracy.

Ratio Weight	mAP@0.5	mAP@0.5:0.9
0	92.10	49.94
0.2	92.27	50.19
0.4	92.41	50.47
0.6	92.58	50.64
0.8	92.63	50.72
1.0	91.93	49.89

From Table 2, it can be observed that when $ratio = 0$, there is an increase of 0.17% in mAP@0.5 and a 0.05% increase in mAP@0.5:0.9 compared to $ratio = 1$, confirming the effectiveness of NWD. As $ratio$ increases, the mAP values also increase. When $ratio = 0.8$, the mAP value reaches its maximum, with a 0.7% increase in mAP@0.5 and a 0.83% increase in mAP@0.5:0.9 compared to $ratio = 1$. These experiments demonstrate that introducing NWD and IoU to optimize the loss function can reduce sensitivity to small target position deviations and improve the detection of small PCB defects. For subsequent experiments, $ratio = 0.8$ will be used for further improvements.

IoU is more sensitive to large targets and targets with high overlap, and can better optimize the overlap of the overall frame. NWD is more sensitive to small targets and frame offsets, and can better optimize the positioning of small targets.0.8 not only ensures the optimization ability of IoU for large targets and overall frame overlap, but also allows NWD to play an auxiliary role in the positioning of small targets. This combination can take into account the detection accuracy of large and small targets.

Actual tests found that 0.8 has the best effect, indicating that the proportion of small targets in the data set is not extremely high, but the improvement of small target detection accuracy has a significant help to the overall mAP. The performance curves of different ratio values are shown in Figure 10 below.

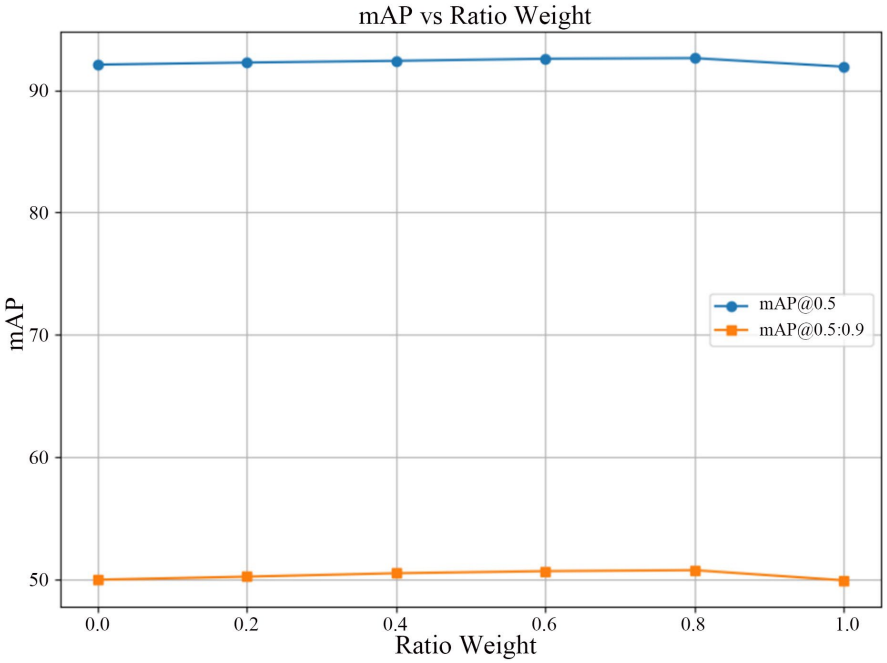


Figure 10. Performance curves for different ratio values.

4.6. Ablation Experiments

Using the aforementioned experimental setup, training experiments were conducted on the improved algorithm, and the convergence of various losses and accuracies during the training process was recorded. The specific results are shown in Figure 11.

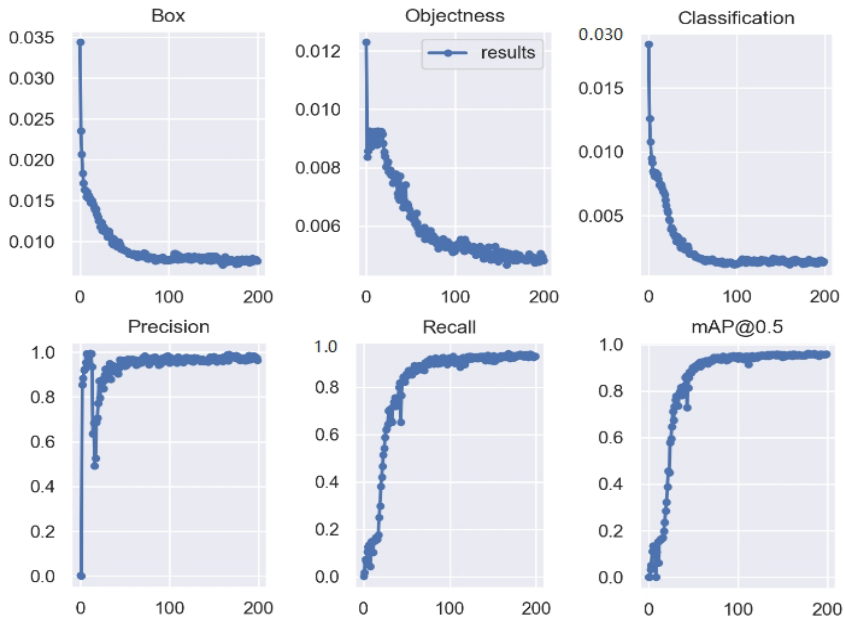


Figure 11. Training loss and precision convergence curve.

It can be observed that after approximately 50 epochs of iteration, the rates of

change in various losses and accuracies begin to slow down. After 100 epochs, the losses and accuracies tend to stabilize and no longer exhibit significant changes. Therefore, it can be determined that the model has converged, and it is advisable to save its weights to ensure the reproducibility of the model post-training and to facilitate predictions or further optimizations when needed.

Table 4. Analysis of ablation results.

Num	Improvement module				mAP@0.5	F1	Params	FLOPs	FPS
	Bi-Former	CARAFE	P2	NWD	(%)	(%)	(M)	(G)	
1					91.93	91.17	36.9	105.1	52.17
2	√				92.98	92.03	37.6	106.6	48.31
3		√			92.79	92.45	37.4	105.3	48.02
4			√		92.35	92.12	37.5	106.7	47.93
5				√	92.63	92.34	37.2	105.2	47.75
6	√	√			93.36	93.15	37.5	106.3	46.83
7	√		√		93.63	93.13	37.5	107.1	45.72
8	√			√	93.52	93.23	37.6	106.6	46.56
9		√	√		93.61	93.28	37.8	107.5	46.65
10		√		√	93.93	93.62	37.4	106.4	46.88
11			√	√	94.21	93.76	37.5	106.3	46.12
12	√	√	√	√	95.25	95.08	38.1	107.6	44.84

Several improvements were proposed targeting the original YOLOv7 algorithm by introducing enhancement modules. To validate the effectiveness of each enhancement module, 12 ablation experiments were designed. The effectiveness of the enhancement modules was evaluated using metrics such as mAP@0.5, parameter count (Params), computational complexity (FLOPs), frames per second (FPS), and the F1 score. The results of the ablation experiments showed significant improvements in the aforementioned variables, as detailed in Table 4.

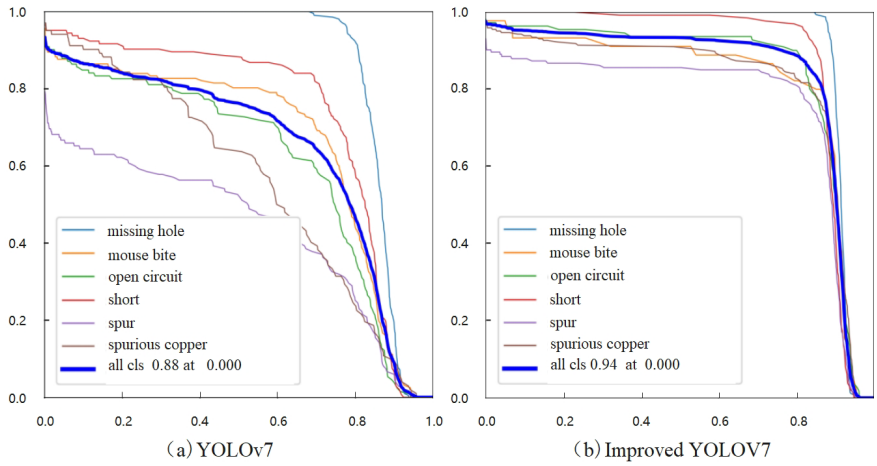
In Table 4, '√' indicates the introduction of the respective module in that experimental group. Compared to the original network, in the second experimental group, introducing the Bi-Former attention mechanism resulted in a slight increase in parameters and computational complexity, but it improved mAP@0.5 by 1.05%, demonstrating its ability to improve detection performance while achieving more flexible content perception. In the third, fourth, and fifth experimental groups, replacing the CARAFE operator, NWD loss function, and introducing the P2 detection head respectively, the parameter count remained basically unchanged, while mAP@0.5 and the F1 score showed varying degrees of improvement. This indicates that these improvements effectively enhance the detection accuracy without increasing model complexity. In the sixth, seventh, and eighth experimental groups, CARAFE operator, CMPANet,

and NWD loss function were respectively added on top of the Bi-Former attention mechanism. Among these three groups, the seventh group showed the highest improvement, with a 1.7% increase compared to the first group. In the twelfth experimental group, the Bi-Former attention mechanism, CARAFE operator, CMPANet, and NWD loss function were all simultaneously introduced. This resulted in a 3.32% improvement in mAP@0.5 and a 3.97% improvement in the F1 score, with only a slight increase in parameter count (1.2 M) and computational complexity (2.5 G). Moreover, the FPS reached 44.84, meeting the real-time testing requirement of over 30 FPS, thereby satisfying industrial production inspection needs.

The detailed mAP results for various types of defect detection ablation experiments are shown in Table 5. The improved algorithms all demonstrate enhanced detection capabilities for various types of defects on PCB bare boards, with the greatest improvement observed for scattered small target defects, showing a 9.3% increase. Since scattered defects are generally smaller and more challenging to identify compared to other types of defects, the significant improvement in defect recognition after improving the YOLOv7 network validates the effectiveness of the improved network for small target detection.

Table 5. Results of various defects in ablation experiments.

Defect Type	Original mAP@0.5/%	Improved mAP@0.5/%	Improvement/%
Mouse Bite	93.6	96.4	2.8
Missing Hole	97.7	99.8	2.1
Open Circuit	93.9	96.1	2.2
Short	93.7	95.6	1.9
Spur	76.3	85.6	9.3
Spurious Copper	94.7	95.9	1.2



(a) YOLOv7 (b) Improved YOLOV7

Figure 12. Comparison of various defect detection recall.

The significant increase in mAP@0.5 is primarily due to the notable improvement in the recall rate R for small targets, as illustrated in Figure 12.

Figure 12 illustrates the comparison of recall rates R for various PCB defects, where the average recall rate has increased from 88% in the original YOLOv7 to 94%, representing a significant improvement of 6%. The BCN-YOLO algorithm maintains high precision while stabilizing parameter and computational requirements, and notably enhances the recall rate. Detailed results of defect detection for each category are depicted in Figure 13.

Figure 13 displays the detection results of the original YOLOv7 algorithm and the BCN-YOLO on the PCB bare board defect dataset. The left side of the image showcases the detection results of the original YOLOv7 model, while the right side illustrates the results of the improved algorithm.

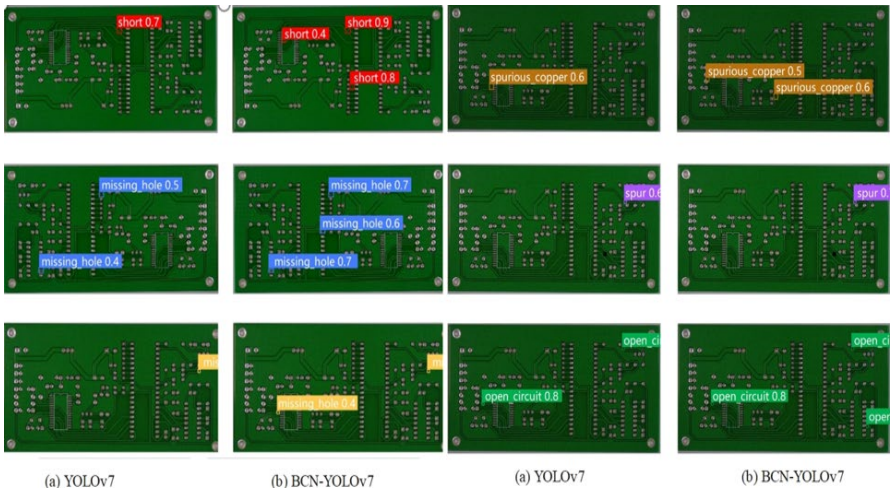


Figure 13. Comparison of various defect detection results.

Figure 13 displays the detection results of the original YOLOv7 algorithm and the BCN-YOLO on the PCB bare board defect dataset. The left side of the image showcases the detection results of the original YOLOv7 model, while the right side illustrates the results of the improved algorithm. Different colored bounding boxes represent detected defects of various types. By comparing the detection results of the two algorithms, it is evident that the improved algorithm enhances both precision and recall for detecting small PCB bare board defects.

In summary, the superiority and effectiveness of the improved algorithm in PCB defect detection tasks have been further validated. It provides detection personnel with a more reliable and accurate method for identifying and categorizing small target defects on PCB bare boards. It is believed that the improved algorithm will play a practical role in the field of PCB defect detection, especially for detecting small target defects.

4.7. Comparative Experiments

To verify the superiority of the BCN-YOLO detection performance, comparative

experiments were conducted against current mainstream single-stage and two-stage networks. The performance of YOLOv7 and BCN-YOLO on different datasets was compared to validate the robustness of the network. The experimental results are presented in Table 6 and Table 7. According to the data in Table 6, compared to the original YOLOv7 network, the improved algorithm achieved a 3.32% and 2.86% increase in mAP@0.5 and mAP@0.5:0.9, respectively. The precision increased by 3.28%, and the recall rate improved by 6.18%, demonstrating the effectiveness of the improved algorithm. In the task of PCB bare board defect detection, the mAP@0.5 value of the improved algorithm is comparable to that of YOLOX-WSC, but with significantly lower parameter and computational requirements compared to the latter network. In the 7 sets of comparative experiments, the recall rate and mAP@0.5 value of the improved algorithm ranked first, and other parameters also showed significant advantages. Considering the comprehensive comparison of model complexity, computational requirements, and application scenarios, the improved algorithm is more suitable for PCB bare board defect detection tasks.

Table 6. Analysis of experimental results.

Algorithm	Precision/%	Recall/%	mAP@0.5/%	mAP@0.5:0.9/%	Params/M	FLOPs/G	FPS
YOLOv7	91.63	88.74	91.93	49.89	36.9	105.1	52.17
YOLOv5-L	92.73	89.53	92.56	50.47	47.1	114.9	49.94
YOLOX-L	94.03	90.24	93.79	50.60	54.2	155.6	51.74
YOLOX-WSC	95.23	94.57	95.45	52.85	70.3	163.4	51.84
Faster R-CNN	94.86	92.35	93.54	51.58	42.6	180.2	28.45
YOLOv7-CA-SIoU	93.53	92.31	93.84	51.41	37.6	113.5	46.72
YOLOv8-L	94.21	93.42	94.81	52.27	43.7	165.7	54.13
BCN-YOLO	94.91	94.92	95.25	52.75	38.1	107.6	44.84

The detailed detection results are illustrated in Figure 14, where the improved network shows significant enhancements in mAP@0.5 across three selected small target datasets compared to YOLOv7. Specifically, in the AI-TOD dataset, mAP@0.5 and mAP@0.5:0.9 improved by 4.67% and 2% respectively; in the Tiny-Person dataset, they improved by 4.55% and 2.03% respectively; and in the Wide-Person dataset, they improved by 4.31% and 2.42% respectively. These results validate the effectiveness of the improved network for small target detection, its generalization capability across different input data, and its enhanced applicability for detecting defects on PCB bare boards.

Table 7. YOLOV7 and BCN-YOLO are compared in different data sets.

Dataset	Algorithm	Params/M	mAP@0.5/%	mAP@0.5:0.9/%	FPS
AI-TOD	YOLOv7	84.3	40.02	19.63	23.21
	BCN-YOLO	90.4	44.69	21.13	15.25

Continued

Tiny-Person	YOLOv7	47.1	36.71	17.42	22.93
	BCN-YOLO	54.3	41.26	19.45	13.56
Wider-Person	YOLOv7	64.8	56.25	24.52	32.41
	BCN-YOLO	72.3	60.56	26.94	21.76

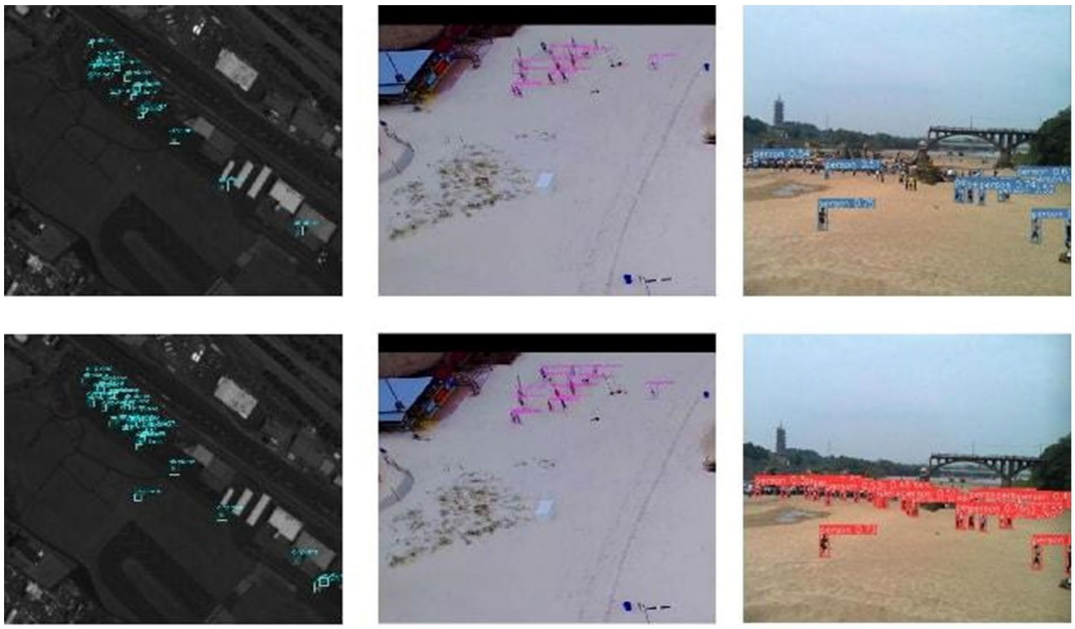


Figure 14. Detailed response about the detection results.

4.8. System Performance Evaluation Based on Real Production Images

The previous content of this article verified the testing performance of the algorithm on the open PCB dataset of Peking University. To further test its usability in actual production environments, a cooperation was established with Dalian Riji Electronics Company. The proposed BCN-YOLO algorithm was applied to the actual PCB board production line, and a comprehensive evaluation of the algorithm was conducted by using real production inspection images.

This study selected four representative types of PCB boards from Dalian Rijia Electronics Co., Ltd. for practical testing. These four PCB boards covered different production processes and complexities to ensure the comprehensiveness and representativeness of the experiment. The actual test results are shown in Figure 13.

As shown in Figure 15, this study's algorithm demonstrates the detection performance comparison of BCN-YOLO on different PCB boards. The algorithm achieves ideal results when detecting six common defects. Based on actual test results, the improved BCN-YOLO basically meets the performance requirements for practical production.

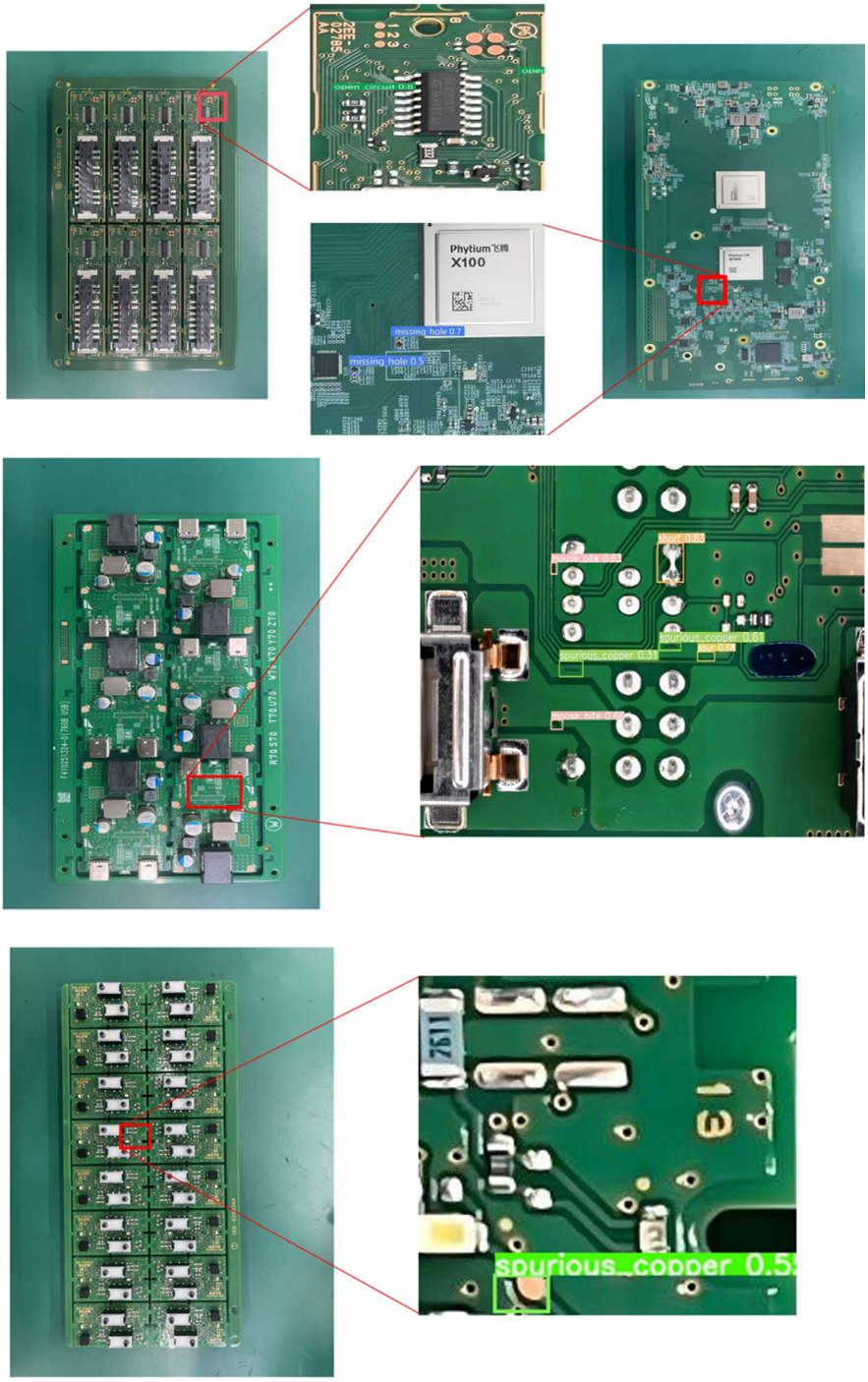


Figure 15. Real production image inspection results.

5. Conclusion and Future Work

Addressing the issues of low accuracy, false positives, and missed detections in the detection of small target defects on PCB bare boards, an improved detection algorithm based on YOLOv7 has been proposed. The improvements include the introduction of a Bi-Former dynamic sparse attention mechanism based on BRA and the CARAFE operator to enhance detection accuracy. Additionally, a robust

feature pyramid network called CMPANet is incorporated to improve the capability of extracting shallow information from small targets. The combination of NWD and IoU loss functions is employed to detect more small targets and reduce false and missed detections during the process. Compared with other classic object detection algorithms, experimental results demonstrate that the proposed algorithm not only outperforms mainstream algorithms in terms of defect detection effectiveness but also shows better generalization capabilities. This makes it particularly valuable for industrial deployment in detecting small target defects on PCB bare boards.

In this paper, we proposed an improved YOLOv7-based PCB bare board defect detection model, BCN-YOLO, which addresses the challenges of missed and false detections in small defect detection. The experimental results demonstrate that the BCN-YOLO model achieved significant improvements in detection accuracy compared to the baseline YOLOv7 model, particularly for small PCB defects. Future work will focus on further optimizing the model architecture and exploring additional loss functions to improve detection performance on challenging PCB defect datasets.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Yuan, Y. (2021) PCB Market Iterative Upgrade Road. *China Information Word*, **51**, 11-29.

[2] Wu, Y.-Q., Zhao, L.-Y., Yuan, Y.-B. and Yang, J. (2022) Research Status and the Prospect of PCB Defect Detection Algorithm based on Machine Vision. *Chinese Journal of Scientific Instrument*, *China*, **8**, 1-17.

[3] Huang, H.-X. and Jin, X. (2021) Small Target Defect Detection Based on YOLOV4. *Electronics World*, **5**, 146-147.

[4] Redmon, J. and Farhadi, A. (2017) YOLO9000: Better, Faster, Stronger. 2017 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Honolulu, 21-26 July 2017, 6517-6525. https://doi.org/10.1109/cvpr.2017.690

[5] Redmon, J. and Farhadi, A. (2018) YOLOV3: An Incremental Improvement. https://doi.org/10.48550/arXiv.1804.02767

[6] Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y.-M. (2020) YOLOV4: Optimal Speed and Accuracy of Object Detection. arXiv:2004.10934.

[7] Li, Y. (2022) Research and Implementation of PCB Defect Detection Based on Deep Learning. Inner Mongolia University.

[8] Su, J., Jia, X.-Y. and Hou, W.-M. (2024) YOLO-J Based PCB Defect Detection Algorithm. *Computer Integrated Manufacturing Systems*, **30**, 3984-2998.

[9] Wang, S.-Q., Zhang, Z.-Y., Zhu, W.-X., Liu, Y.-F., Wang, J. and Li, Q.-Y. (2023) Surface Defect Detection of PCB Based on Improved YOLOV5. *Instrument Technique and Sensor*, **5**, 106-111.

[10] Tuo, B., Huang, L.-W., Tang, X., Chen, L.-Y. and Zhou, J. (2023) Research on PCB Defect Detection Algorithm Based on YOLOX-WSC. *Computer Engineering and*

Applications, **59**, 236-243.

[11] Mohammed, A., Kashif, M., Zama, M.H., Ansari, M.A. and Ali, S. (2023) Master GAN: Multiple Attention Is All You Need: A Multiple Attention Guided Super Resolution Network for Dems. *IGARSS* 2023-2023 *IEEE International Geoscience and Remote Sensing Symposium*, Pasadena, 16-21 July 2023, 5154-5157. https://doi.org/10.1109/igarss52108.2023.10283196

[12] Zhu, L., Wang, X., Ke, Z., Zhang, W. and Lau, R. (2023) Biformer: Vision Transformer with Bi-Level Routing Attention. 2023 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Vancouver, 17-24 June 2023, 10323-10333. https://doi.org/10.1109/cvpr52729.2023.00995

[13] Wang, J., Chen, K., Xu, R., Liu, Z., Loy, C.C. and Lin, D. (2019) CARAFE: Content-Aware Reassembly of Features. 2019 *IEEE/CVF International Conference on Computer Vision* (*ICCV*), Seoul, 27 October 2019-2 November 2019, 3007-3016. https://doi.org/10.1109/iccv.2019.00310

[14] Jiang, Y.-Q., Tan, Z.-Y., Wang, J.-Y., Sun, X.-Y., Lin, M. and Li, H. (2022) GiraffeDet: A Heavy-Neck Paradigm for Object Detection. arXiv:2202.04256.

[15] Liu, S., Qi, L., Qin, H., Shi, J. and Jia, J. (2018) Path Aggregation Network for Instance Segmentation. 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 18-23 June 2018, 8759-8768. https://doi.org/10.1109/cvpr.2018.00913

[16] Tan, M., Pang, R. and Le, Q.V. (2020) EfficientDet: Scalable and Efficient Object Detection. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Seattle, 13-19 June 2020, 10778-10787. https://doi.org/10.1109/cvpr42600.2020.01079

[17] Kim, K. and Lee, H.S. (2020) Probabilistic Anchor Assignment with IOU Prediction for Object Detection. *Computer Vision-ECCV* 2020: 16*th European Conference*, Glasgow, 23-28 August 2020, 355-371.

[18] Ge, Z., Liu, S.-T. Wang, F., Li, Z.-M. and Sun, J. (2021) YOLOX: Exceeding Yolo Series in 2021. arXiv:2107.08430.

[19] Decreusefond, L. (2008) Wasserstein Distance on Configuration Space. *Potential Analysis*, **28**, 283-300. https://doi.org/10.1007/s11118-008-9077-5