ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 3 - 2026

FHIM-YOLO: Balancing Accuracy and Efficiency for Real-Time PCB Surface Defect Detection

Hao Cui[1,*]

[1] School of Physics and Electronic Information Engineering, Henan Polytechnic University, Jiaozuo 454003, China

SUMMARY: *To meet the high demands for quality and reliability in the manufacturing of electronic products, surface micro-defect inspection of printed circuit boards (PCBs) is now required. At present, the methods available do not meet the demand for light-weight deployment and high-accuracy small-target detection in complicated industrial environments. Therefore, an efficient lightweight detector called FHIM-YOLO based on YOLOv8n has been introduced. C2f_Faster is a modification of C2f in the backbone that reduces the number of parameters and memory access; it is thus more efficient in computation. An optimised feature enhancement module (FEM) is added to enhance the local contextual representation of fine defects through multi-branch dilated convolutions and address feature degradation in deep networks. A High-level Screening-feature Fusion Pyramid Network (HS-FPN) is used in the neck to improve the efficiency of multi-scale feature screening and fusion, thereby better isolating micro-defects from background clutter. Finally, CIoU is replaced with Inner-MPDIoU to improve the accuracy of bounding-box regression in a cluttered background and speed up the training process. On the HRIPCB dataset, FHIM-YOLO has reached a top-5 mAP of 95.2% and reduced the number of parameters to 0.95M, which is 68.4% lower than that of the YOLOv8n baseline. Although the FLOPs are as high as 15.6G, the model can still run at 65.19 FPS and meet the requirements for real-time industrial applications. Additional experiments on the DeepPCB dataset have shown that, compared to the baseline, Recall, mAP@50, and mAP@50-95 have all improved and exhibit good generalisation to different data distributions. In short, FHIM-YOLO offers a good balance among model size, detection accuracy and real-time performance, and is thus very suitable for resource-limited online PCB defect inspection.*

KEYWORDS: *PCB defect detection; YOLOv8n; FHIM-YOLO; feature enhancement; bounding box regression loss*

1 Introduction

PCBs are used to build many electronic devices now, so they have a direct impact on the work environment and life span of these devices. There will be certain kinds of surface defects on the PCBs during production, such as open circuits, mouse bites and spurs, etc. The above defects will reduce the extended service life of the equipment. Therefore, to improve the production efficiency and quality of products, high-precision and high-sensitivity PCB defect-inspection technologies with low false-alarm rates are needed.

However, PCB defect detection is still a challenge: minor defects occupy a very small area in the image and are thus difficult to find; background clutter and texture interference prevent

*ch@home.hpu.edu.cn
https://doi.org/10.65102/is20261264
Published: 10 June, 2026

the extraction of discriminative features; and an excellent balance of efficiency and high accuracy has not been reached.

Before the spread of deep learning, only manual visual inspection and traditional machine vision could be used to identify PCB defects. Manual methods are subject to errors such as subjectivity, low efficiency and potential misdetection or missed detection [1]. Traditional machine vision is also unable to recognise small and complex defect features, and it has poor cross-scene generalization [2]. On the other hand, deep learning-based detectors are now widely used due to their superior accuracy, robustness and real-time performance [3].

Existing deep learning detection technologies can be broadly divided into two categories based on region extraction (such as Faster R-CNN) and one-stage approaches that prioritize inference speed (e.g., YOLO and SSD). Many have been studied so far. Ren et al. [4, 5] introduced an RPN in the R-CNN framework for cooperative region proposal and classification to balance detection speed and accuracy. Ding and others used k-means clustering to opti[6]mise Faster R-CNN with adaptive anchors and improved the detection performance of small defects. Liu and Wen [7, 8] proposed MobileNet-YOLO-Fast to improve the speed of real-time applications by reducing parameters. Kang and others replaced the SSD backbone with ResNet50 and added a special small-target layer for improved feature extraction. Two-stage detectors are usually more accurate, but they are also relatively slow and not suitable for real-time applications. YOLO is a single-stage detector that is both accurate and high-speed, thus suitable for practical use in PCB defect detection. Recently, some researchers have begun to adapt the YOLO architecture for specific domains. [9] improved the bounding box regression of YOLOv4 with GsIoU. [10] proposed PCB-YOLO based on YOLOv5 to enhance the speed of PCB surface defect detection. [11] added coordinate attention (CA) to YOLOv7-tiny to achieve more precise feature extraction. Li and others [12] introduced an adown module to reduce the number of parameters for multi-class defect detection in ASF-YOLO. Despite the above progress, the current technology still has a small-target detection problem, which generally shows a lack of large receptive fields and limited fusion effectiveness among different spatial scales.

YOLOv8 will be taken as the basic framework of this paper; it has a well-developed ecosystem and relatively good performance, and has been used to conduct extensive experiments on PCB defect detection[13] and other high-precision inspection problems. YOLOv8 uses a backbone network with CSP and SPPF, and also has a PAN-FPN feature fusion module in the neck [14]. The Design will improve detection accuracy and inference speed, and thus YOLOv8 will be widely applied to PCB defect recognition. Ling and others [?] strengthened the combined local and global feature extraction of the YOLOv8 backbone by introducing C2Focal. Chen and Xie proposed Wise-IoU and W-YOLOv8 to improve the capacity for key feature extraction. To reduce the computational cost of PCB detection, Zhang et al. [15-17] proposed YOLO-RRL and used an RFD module to enhance downsampling and optimise multi-scale feature fusion. YOLOv8 still lacks the ability to detect fine-grained features and is thus not very small or very accurate.

In light of the above limitations, this study proposes FHIM-YOLO with the following contributions.

(1) A new detection framework called FHIM-YOLO will be built on the YOLOv8n foundation model. Architectural optimisation has been carried out to meet the strict speed requirements of industrial applications, and at the same time, a significant reduction in the number of model parameters and a substantial improvement in detection accuracy have been achieved.

(2) Optimised FEM is used in the backbone network to enhance perception and feature representation for small-defect detection. Many Dilations are used to avoid losing the subtle

spatial features of the image. At the same time, the light-weight C2f_Faster can be used instead of C2f to reduce both computation and memory-access costs.

(3) HS-FPN is used to redesign the neck network to maintain the accuracy of small defect detection while reducing the computational and parameter count significantly.

(4) Inner-MPDIoU is used to improve the detection accuracy and convergence speed for small defects by adjusting the scale of auxiliary bounding boxes dynamically.

The arrangement of this work is as follows. Section 2 introduces the FHIM-YOLO model and how it is improved. Section 3 presents the results of the experiment and comparative analysis, as well as ablation studies. Section 4 is the research results, and the space for further studies will be discussed here.

2 Methods

2.1 General Structure

YOLOv8 [18] generally performs well in object detection due to a CSP backbone and PAN-FPN mechanism [19], but it has been facing problems in PCB defect recognition. Specifically, the backbone provides only weak representations of fine details, and the dense C2f blocks increase the computational cost; thus, it cannot meet the low-latency and high-accuracy requirements of industry at the same time [20]. The neck also lacks adaptive screening for multi-scale features and thus reduces the detection rate of micro-targets. Finally, for PCB defects with an extremely high aspect ratio, the convergence speed and localisation accuracy of CIoU can be limited.

Based on the YOLOv8n architecture, backbone, neck and loss functions have been improved to create FHIM-YOLO. As shown in Figure 1, lighter C2f_Faster blocks are used instead of C2f blocks, and the C2f structure in the neck remains unchanged. The improved FEM is employed in the backbone, and multiple branches of dilated convolutions are used to enhance the robustness of feature representation for small defects. HS-FPN is used to redesign the neck for adaptive feature selection and efficient multi-scale fusion; at the same time, small defects can be enhanced, background noise suppressed, and model complexity reduced. Finally, CIoU loss is replaced by Inner-MPDIoU loss to improve the speed of convergence and regression accuracy. The above modifications will improve the overall detection ability of the PCB defects.

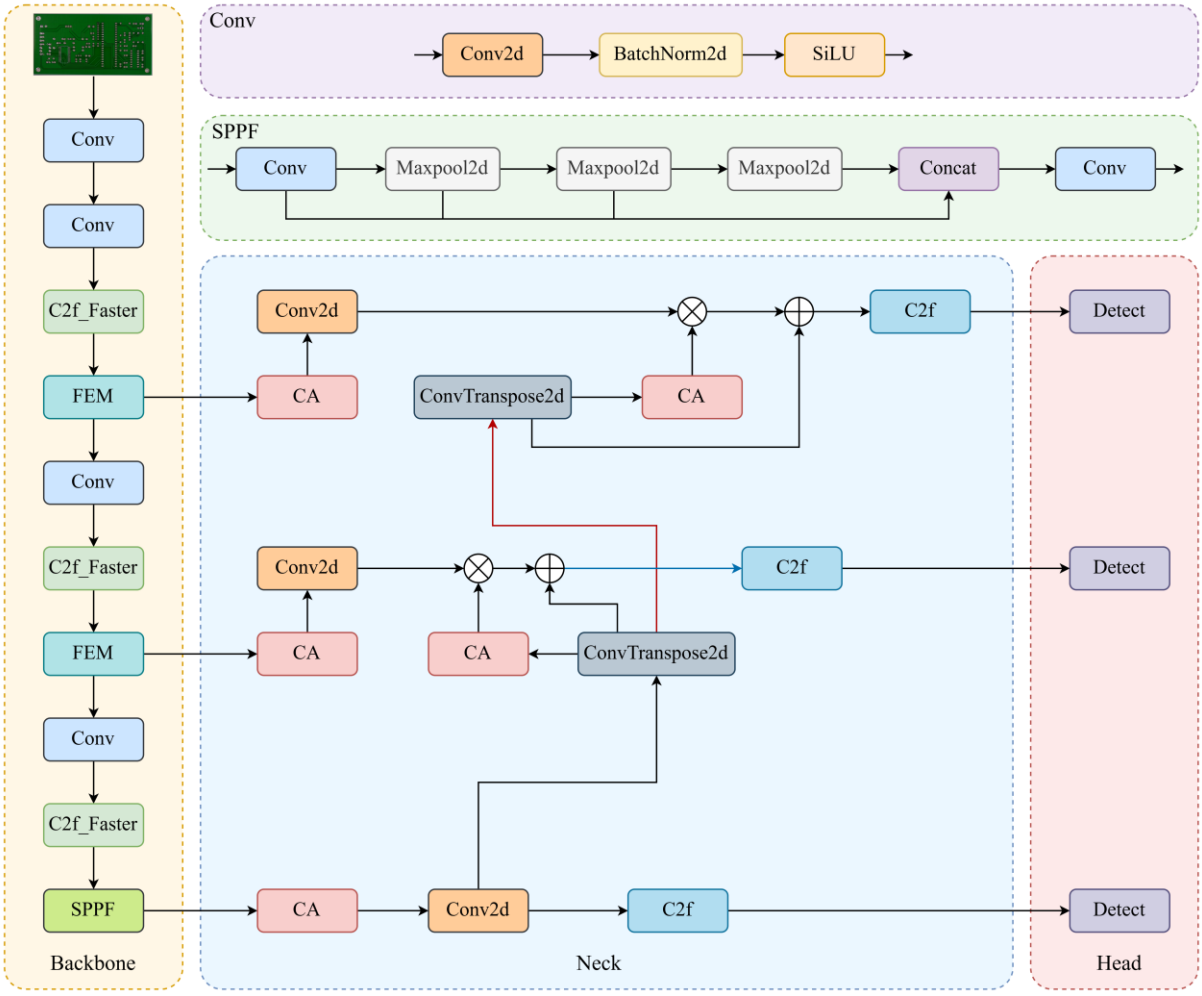

Figure 1: Framework of FHIM-YOLO.

2.2 Optimization of the Backbone Network

2.2.1 FEM

The semantic extraction capability of the YOLOv8n backbone is relatively weak, and thus it may be prone to feature confusion in the detection of small PCB defects. Based on the FEM proposed by Zhang et al. [21], this paper further optimises it for PCB surface defect detection. Figure 2 is shown below. Instead of the single, large dilation rate (Rate=5) of the original FEM, the optimised FEM now uses several smaller dilation rates (Rate=2, 3). Therefore, this Design will have a relatively high sampling rate and be more likely to detect small defects. In addition, this study also maintains the multi-branch convolution structure and, thus, captures semantic information at multiple depths in the features. The two above can improve the representation of minor-defect features and enhance general perception quality through FEM.

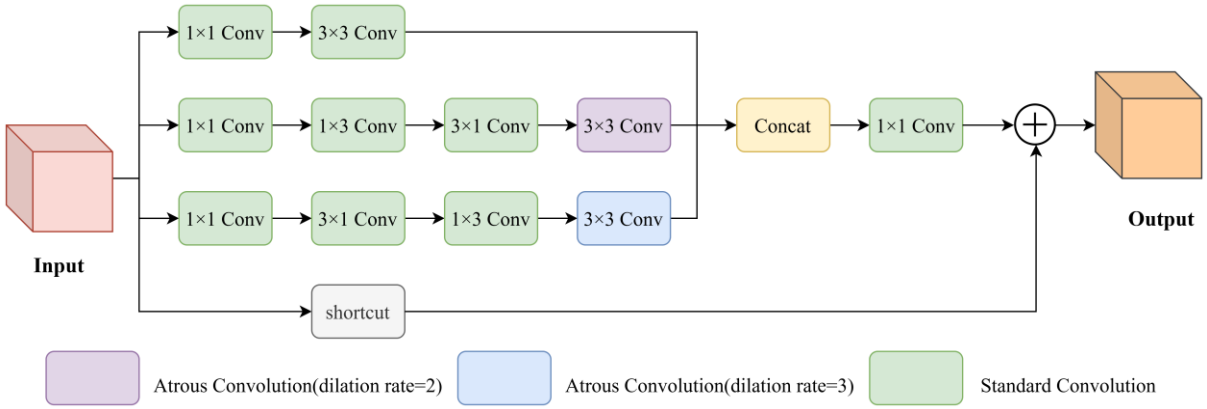

Figure 2: Framework of FEM. Shortcut is a residual connection operation, and Concat is the operation of feature map concatenation.

FEM is as follows mathematically:

$$C_1 = f_{conv}^{3\times3}[f_{conv}^{1\times1}(Input)] \tag{1}$$

$$C_2 = f_{diconv}^{3\times3}\{f_{conv}^{3\times1}\{f_{conv}^{1\times3}[f_{conv}^{1\times1}(Input)]\}\} \tag{2}$$

$$C_3 = f_{diconv}^{3\times3}\{f_{conv}^{1\times3}\{f_{conv}^{3\times1}[f_{conv}^{1\times1}(Input)]\}\} \tag{3}$$

$$Output = f_{conv}^{1\times1}[Cat(C_1, C_2, C_3)] \oplus Shortcut(Input) \tag{4}$$

Among them, $f_{conv}^{1\times1}$, $f_{conv}^{3\times3}$, $f_{conv}^{3\times1}$, and $f_{conv}^{1\times3}$ respectively denote standard convolutions with 1×1, 3×3, 3×1, and 1×3 kernels. $f_{diconv}^{3\times3}$ represents a dilated convolution operation with a variable dilation rate. $\oplus$ indicates the element-level addition operation of the feature map. C_1, C_2, and C_3 depict the first three branches' output feature maps. “Input” and “Output” are the input- and output-feature maps of FEM.

2.2.2 C2f_Faster

To improve detection performance and maintain inference speed, C2f_Faster is proposed here. It is based on the core idea of Partial Convolution (PConv) [22]. As shown in Figure 3, the bottleneck component of the original C2f part has been replaced by FasterBlock. C2f_Faster reduces redundant computations and memory accesses without harming the accuracy of feature extraction; thus, it has improved inference efficiency and is highly suitable for real-time interactions and compute-intensive scenarios.

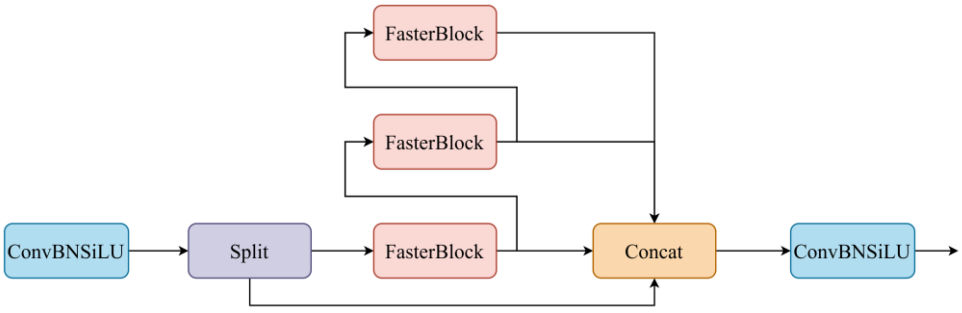

Figure 3: Lightweight C2f_Faster Module Architecture Diagram Integrated with FasterBlock.

2.3 HS-FPN

HS-FPN is proposed to handle the complex background and small defects of PCBs that have higher requirements for feature fusion. Industrial edge devices have limited computing power; therefore, the actual application needs to be a light-weight design. Therefore, this study will use HS-FPN[23] to replace the original neck design. HS-FPN can combine and optimise multi-scale features effectively, and has a small number of parameters and computational complexity. The two components are feature fusion and selection. Together, these modules can enhance the semantic representation of features to improve the detection performance of small-object detection. The framework of HS-FPN can be seen in Figure 4.

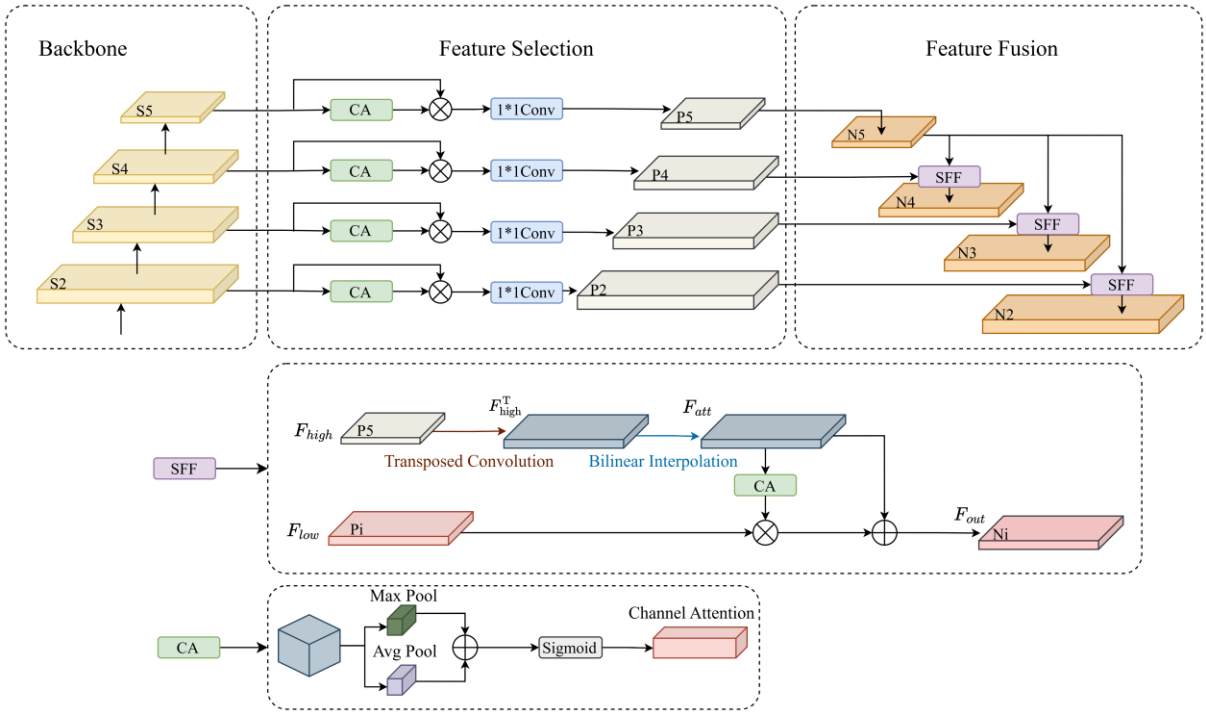

Figure 4: Framework of HS-FPN. SFF is for selective feature fusion, and CA is the channel attention mechanism.

Feature Selection Module: A new Screening method will be added to this module. Dynamically generate channel-wise weights via a Channel Attention (CA) module, then multiply these weights with the input features element-wise. A 1x1 convolution is used to reduce the number of channels.

Feature fusion module: The core module for feature fusion is the selective feature fusion (SFF) operation. Filters and selects the relevant semantic information from low-level features according to high-level features as weights. The input consists of high- and low-level feature maps that have been generated by feature selection. The fusion process is given by (5)-(6).

$$F_{att} = BL(T - Conv(F_{high})) \tag{5}$$

$$F_{out} = F_{low} * CA(F_{att}) + F_{att} \tag{6}$$

In the formula, BL shorts for bilinear interpolation; $T - Conv$ stands for transposed convolution; CA stands for channel attention mechanism; Feature map $F_{att} \in R^{C \times H1 \times W1}$; Output the feature map $F_{out} \in R^{C \times H1 \times W1}$.

2.4 Inner-MPDIoU

Minor defects and background interference make PCB defect detection difficult, and conventional localization loss functions may not be suitable for both detection accuracy and optimisation behaviour. Therefore, to improve both the generalisation ability and convergence speed of this paper, the Inner-MPDIoU loss function will be adopted. Inner-IoU controls the scale of the auxiliary bounding box via a scale factor [24], and thus helps to learn defect information at all scales effectively. The related formula is:

$$B_l = x_c - \frac{w*ratio}{2}, B_r = x_c + \frac{w*ratio}{2} \tag{7}$$

$$B_t = y_c - \frac{h*ratio}{2}, B_b = y_c + \frac{h*ratio}{2} \tag{8}$$

$$B_l^{gt} = x_c^{gt} - \frac{w^{gt}*ratio}{2}, B_r^{gt} = x_c^{gt} + \frac{w^{gt}*ratio}{2} \tag{9}$$

$$B_t^{gt} = y_c^{gt} - \frac{h^{gt}*ratio}{2}, B_b^{gt} = y_c^{gt} + \frac{h^{gt}*ratio}{2} \tag{10}$$

$$S_{inter} = \left(min\left(B_r^{gt}, B_r\right) - max\left(B_l^{gt}, B_l\right)\right) * \left(min\left(B_b^{gt}, B_b\right) - max\left(B_t^{gt}, B_t\right)\right) \tag{11}$$

$$S_{union} = (w^{gt} * h^{gt}) * (ratio)^2 + (w * h) * (ratio)^2 - S_{inter} \tag{12}$$

$$IoU^{inner} = \frac{S_{inter}}{S_{union}} \tag{13}$$

where (x_c, y_c) represents the shared center point of the anchor box and the inner anchor box, and (x_c^{gt}, y_c^{gt}) means the shared center point of the ground-truth box and the auxiliary bounding box. h and w mean the height and width of the anchor box, respectively, while h^{gt} and w^{gt} mean those of the ground-truth box. B_l, B_r, B_b, and B_t represent the center points of the left, right, bottom, and top boundaries of the anchor box, respectively, while B_l^{gt}, B_r^{gt}, B_b^{gt}, and B_t^{gt} denote the corresponding points of the ground-truth box.

MPDIoU is employed to reduce the distance between the lower-right (upper-left) corners of the predicted bounding box and those of the ground-truth box [25], thus improving the accuracy of localisation and enhancing the stability of regression.

$$d_1^2 = (x_1^{prd} - x_1^{gt})^2 + (y_1^{prd} - y_1^{gt})^2 \tag{14}$$

$$d_2^2 = (x_2^{prd} - x_2^{gt})^2 + (y_2^{prd} - y_2^{gt})^2 \tag{15}$$

$$MPDIoU = IoU - \frac{d_1^2}{w^2+h^2} - \frac{d_2^2}{w^2+h^2} \tag{16}$$

Among them, (x_1^{gt}, y_1^{gt}) and (x_2^{gt}, y_2^{gt}) denote the coordinates of the upper-left and lower-right corners of the actual bounding box, respectively, while (x_1^{prd}, y_1^{prd}) and (x_2^{prd}, y_2^{prd}) refer to those of the estimated box. w and h denote the input image's width and height, respectively.

The loss function of MPDIoU is as follows: Equation (17).

$$L_{MPDIoU} = 1 - MPDIoU \tag{17}$$

MPDIoU and Inner-IoU are used to construct an inner-MPDIoU loss that enhances the fitting of low-quality samples by including IoU calculation for auxiliary bounding boxes. Inner-MPDIoU generally reduces the problems of complex background textures and small defects, thus improving the accuracy and convergence stability of the model. The Formula is:

$$L_{Inner-MPDIoU} = L_{MPDIoU} + IoU - IoU^{inner} \tag{18}$$

Among them, IoU^{inner} is the ratio of intersection to union of auxiliary bounding boxes.

3 Experiments and Results

3.1 Experimental Setup

All the experiments were carried out in Ubuntu 20.04, and an NVIDIA GeForce RTX 4090 GPU with 24 GB of memory and an Intel Xeon Gold 6430 CPU (16 vCPUs) were used. All the models were built in Python 3.8 and PyTorch 2.0.0. All the models in the ablation experiments were trained from scratch under the same training conditions. The resolution of the input image is 640×640; the batch size is 64; the number of training epochs is 200; the random seed is set to 0; and AdamW is used as the optimizer with an initial learning rate of 0.01 and a weight decay of 0.0005. The above are the training parameters. The models used for comparison were obtained from the official GitHub releases. Given the same input size, batch size and training epochs, the default hyperparameter settings of all repositories were used to ensure fairness and strictness in the comparison. The confidence threshold and IoU threshold for the test were set to 0.001 and 0.7, respectively; they are the same as those in the ablation study.

3.2 Data Sources

HRIPCB [26] is the primary dataset; it is a public PCB defect dataset from Peking University. It has 693 images and corresponding annotation files of the six types of defects (see Figure 5). Most of the defects in each image are relatively small and account for about 0.24% of the total pixel area [27]. The ratio of the 8:1:1 split for the training set, validation set and test set in this study is used. This division will introduce some rare fault types in all divisions to help generalisation.

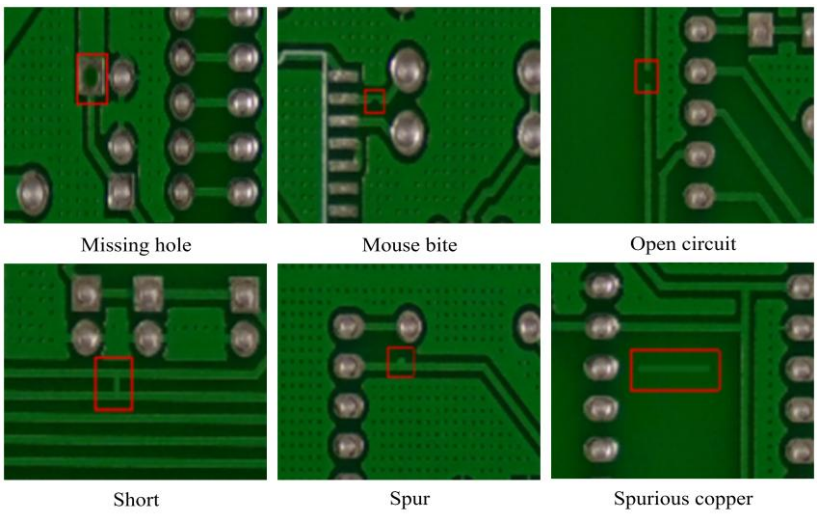

Figure 5: Six Typical PCB Defects.

Test the above model on DeepPCB and check its cross-datas[28]et generalisation performance. The six kinds of defects in DeepPCB are short, open, spur, copper, mouse bite and pin-hole (which are the same as the missing-hole defects in HRIPCB). 1,500 images (640×640) have been annotated with accurate bounding boxes. DeepPCB has more uniform images and less background interference than HRIPCB, so the detection difficulty is relatively lower [29]. This paper uses the 8:1:1 training/validation/test split for this dataset.

3.3 Indicators

To comprehensively and statistically evaluate the suggested PCB fault detection model, precision (P), recall (R), average precision (AP), and mean average precision (mAP) were taken as assessment indicators in this study. Model size is the number of parameters in the model. The quantity of floating-point operations per second (GFLOPs) indicates computational difficulty, and frames per second (FPS) shows inference speed. The formulas of the above three indices are as follows: TP = True positives, FP = False positives, and FN = False negatives.

$$P = \frac{TP}{TP+FP} \tag{19}$$

$$R = \frac{TP}{TP+FN} \tag{20}$$

$$AP = \int_0^1 P(R)dR \tag{21}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{22}$$

3.4 Ablation Studies

This section presents ablation studies of FHIM-YOLO to verify that each proposed part. First, the baseline YOLOv8n was used. Then, components FEM+C2f_Faster, HS-FPN and the Inner-MPDIoU loss were added sequentially to study their individual contributions. Finally, the components were paired and then all of them were examined together. As shown in Table 1, the eight ablation settings are as follows: Model I is YOLOv8n. All the experiments were conducted in the same training environment for fair comparison.

Method (VIII) is reasonably light and has good detection accuracy. Adding FEM+C2f_Faster (II) increased FLOPs due to the 1×1 dimensionality-reduction convolution performed by FEM on the concatenated high-dimensional feature maps, but it significantly reduced the parameter count and raised mAP@50 from 89.1% to 96.1%. HS-FPN (III) and Inner-MPDIoU (IV) further increased the accuracy by 3.6% and 5.9%, respectively, and HS-FPN also reduced the model size. Synergistically, the mAP@50 values for Model V and Model VII were slightly lower than those of Model II and Model IV. It is probably because HS-FPN has significantly reduced both FLOPs and parameters, and thus does not cause a problem. However, it was still better than that of the baseline. When all the components were added (VIII), the parameters dropped to 0.95 M from 3.01 M (a decrease of 68.4%) and mAP@50 increased to 95.2% (an absolute increase of 6.1 over the baseline). Although the FPS decreased to 65.19 due to an increase in FLOPs, the model was still feasible for real-time detection. Model VIII is generally the best all-round trade-off among the above settings.

Table 1: Comparison of the Results of Ablation Experiments.

Model	FEM+ C2f_Faster	HS-FPN	Inner-MPDIoU	mAP@50/%	Params/M	FPS(f/s)	FLOPs/G
I				89.1	3.01	88.50	8.1
II	√			96.1	1.98	67.11	21.8
III		√		92.7	1.96	81.83	6.5
IV			√	95.0	3.01	89.93	8.1
V	√	√		94.7	0.95	60.83	15.6
VI	√		√	95.8	1.98	67.02	21.8
VII		√	√	93.7	1.96	84.89	6.5
VIII	√	√	√	95.2	0.95	65.19	15.6

√ means adding the module.

As shown in Figure 6, P-R curves of the HRIPCB test set before and after optimisation are presented here. As shown in Figure 6(a), YOLOv8n had an mAP@50 of only 89.1% and was not suitable for small defect detection. The lower two categories (mouse bites and spurs) had a lower starting value of 83.9% and 78.1%. As shown in Figure 6(b), FHIM-YOLO increased mAP@50 to 95.2%. Among the spur type that was lower than the others in the baseline, only one had a higher accuracy of 91.5% and the mouse-bite type rose by 9.7%. After optimisation, the envelope area of the P-R curve is larger and more concentrated in distribution. The range of the worst-to-best category performance has decreased to 7.5%, indicating that most defect categories can now be detected more reliably.

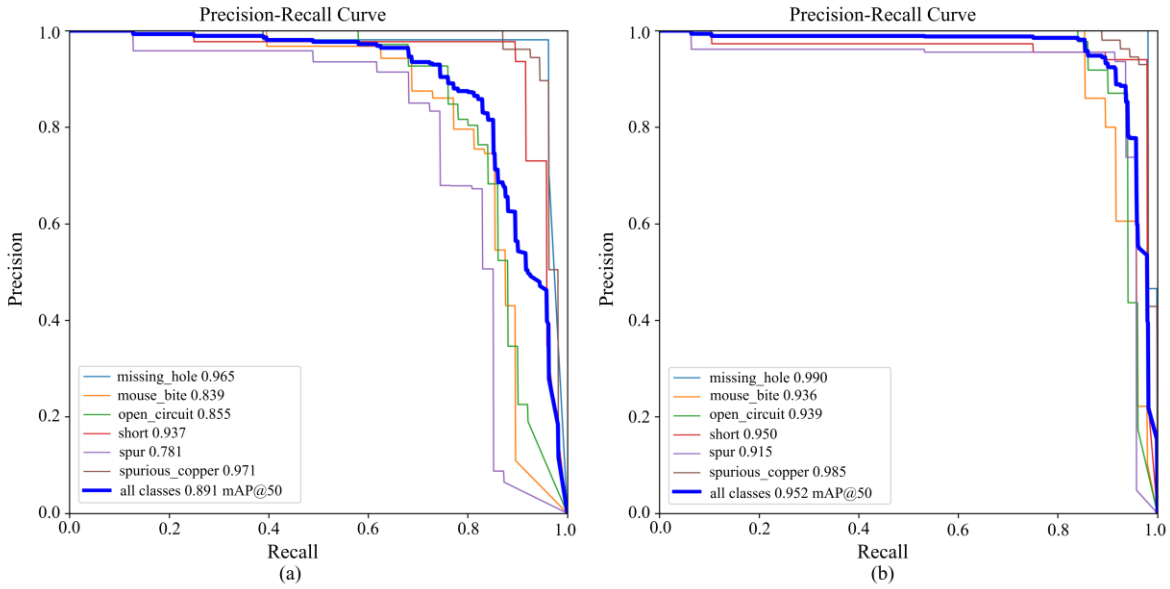

Figure 6: PR curves of YOLOv8n and FHIM-YOLO on HRIPCB. (a) PR curve of YOLOv8n. (b) PR curve of FHIM-YOLO.

Some Modules are designed to provide the necessary functions of this model.

3.4.1 FEM+C2f_Faster

To further verify the effectiveness of FEM+C2f_Faster for small defects, this paper grouped the defects by size and compared the mAP@50 improvements. Spur, open circuit and mouse bite are regarded as small objects; short defects are considered medium objects; and spurious

copper and missing holes are considered large objects. With FEM+C2f_Faster, the mAP@50 values for missing hole, mouse bite, open circuit, short, spur, and spurious copper were 98.3%, 94.5%, 95.0%, 99.3%, 90.8%, and 98.8%, respectively. Based on the above data, the average mAP@50 values of the original and improved models for each group of defects were computed. As shown in Figure 7, the mAP@50 for small objects increased from 82.5% to 93.4% (an absolute increase of 10.9%). The mAP@50 for medium objects rose to 99.3% from 93.7%, and this increase rate was lower than that for small objects. The large objects were still highly accurate and received a small improvement. Based on the above, FEM+C2f_Faster can reduce the impact of feature loss in small targets and improve multi-scale robustness.

It can be seen from the above that the hierarchical receptive fields produced by FEM are better at retaining fine morphological features of small abnormalities that are likely to be lost in standard continuous pooling operations.

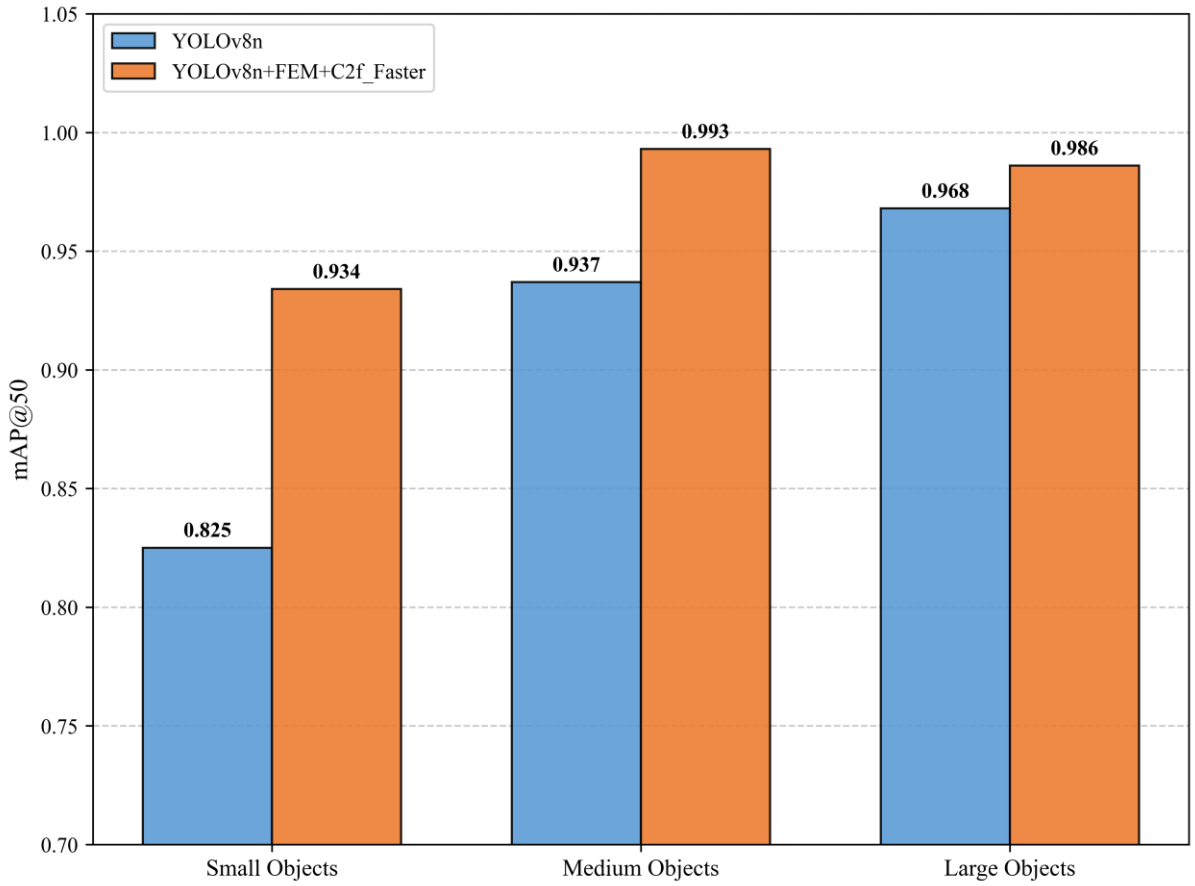

Figure 7: Mean of defects in each group before and after improvement mAP@50.

Figure 8 shows the quality results. The baseline model was unable to identify flaws such as spurs and mouse bites in some pictures because they were too small and had a low feature response. Although the improved model did not eliminate the missed detections entirely, it clearly improved the detection results and reduced false positives compared to the baseline, especially for micro-defects such as mouse bites and open circuits.

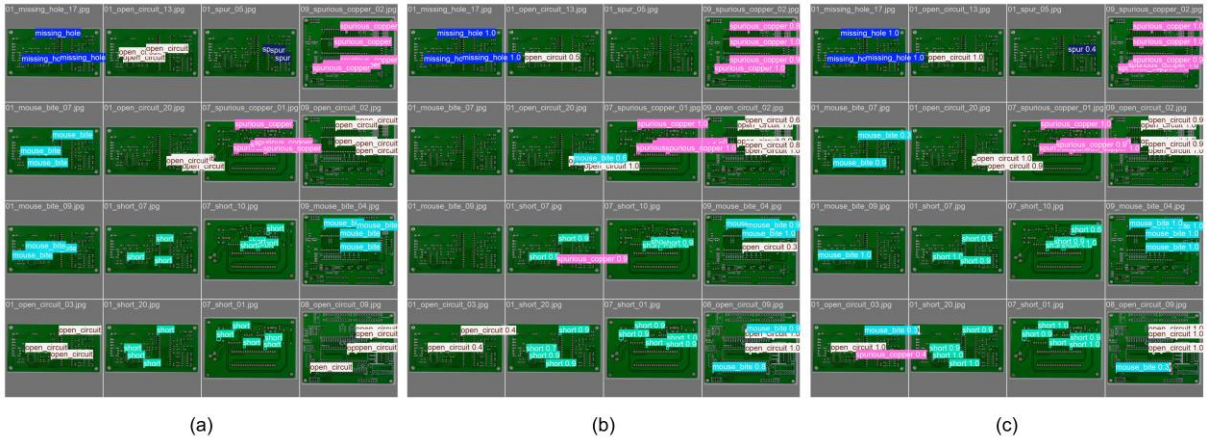

Figure 8: Qualitative detection performance on HRIPCB: (a) Ground truth annotations; (b) Predictions of YOLOv8n; (c) Predictions of the model with FEM+C2f_Faster.

3.4.2 HS-FPN

To verify that HS-FPN can preserve the important features of small defects, the confusion matrices are shown in Figure 9. Diagonal entries are recall values; Background is not a detection. Mouse-bite and short-circuit defects benefited most from HS-FPN. The proportion of missed mouse-bite samples in the baseline was 17%; after adding HS-FPN, it dropped to 10%. The recall rates of short and open circuits increased by 11% and 4%, respectively, and thus the detection sensitivity of slender linear features also rose. In the baseline, 8% and 2% of the short-circuit cases were incorrectly identified as spur and spurious copper defects, respectively; this confusion has been significantly reduced by the improved model. The above results show that HS-FPN learns better-discriminative representations to reduce category confusion. HS-FPN dynamically adjusts feature weights at different spatial scales to reduce the semantic gap between deep semantic layers and shallow high-resolution layers, and thus isolate genuine defect signals from background trace patterns.

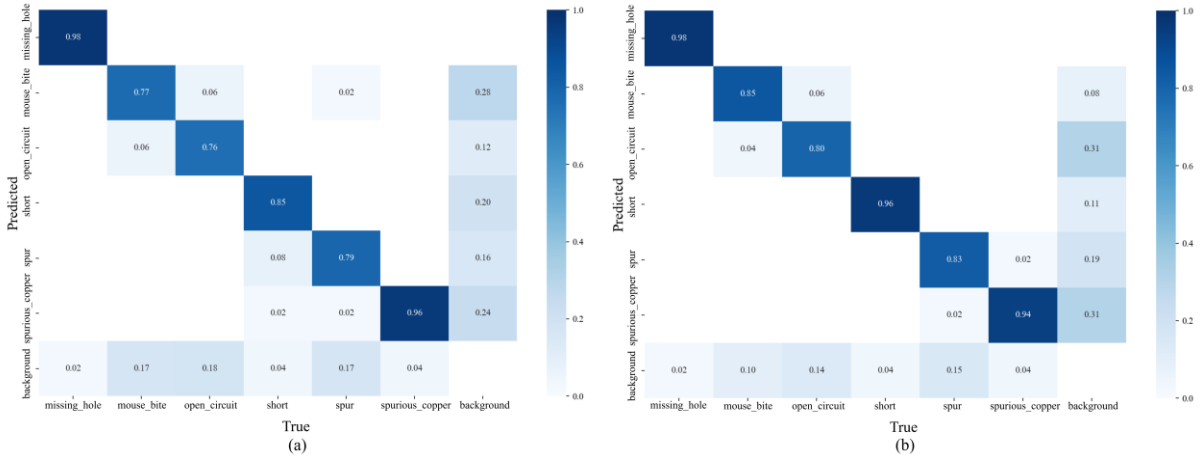

Figure 9: Confusion Matrices of YOLOv8n and YOLOv8n+HS-FPN. (a) YOLOv8n; (b) YOLOv8n+HS-FPN.

HS-FPN improves efficiency as well; YOLOv8n with HS-FPN has only 1.96M parameters and 6.5 GFLOPs, which is significantly lower than the baseline. Figure 10 shows the mAP@50 performance and parameter count of several models. The improved model with HS-FPN maintained a high accuracy and had a smaller number of parameters. And Table 1 shows that

FLOPS of III are 1.6G lower than those of YOLOv8n (I). Therefore, HS-FPN has contributed to the lightweighting project.

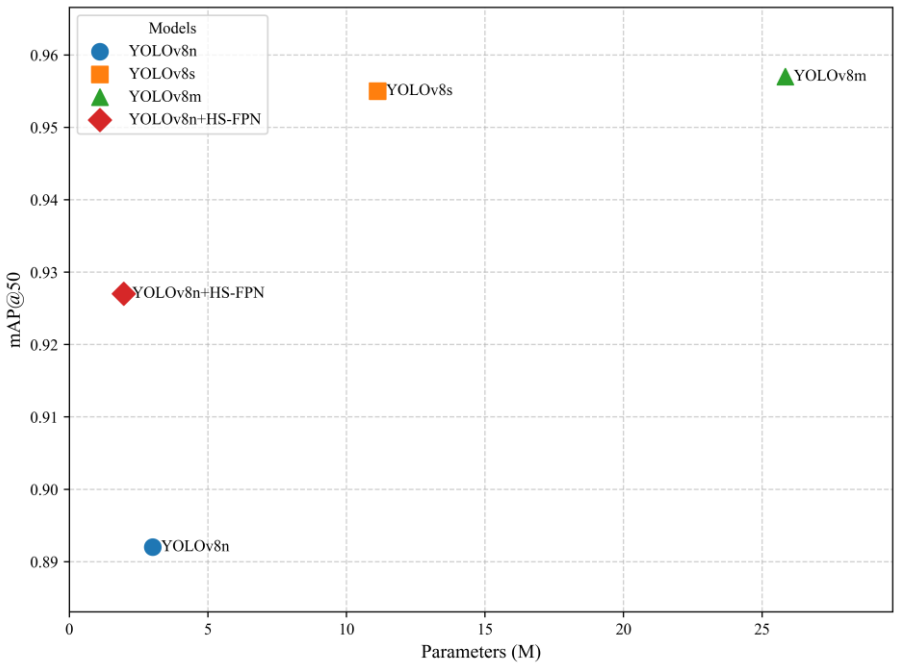

Figure 10: Trade-off between mAP@50 and parameter count of the compared models.

3.4.3 Inner-MPDIoU

To verify that the inner-MPDIoU loss has improved detection accuracy and accelerated convergence, the training dynamics of the improved method and the baseline were also compared in this paper. As shown in Figure 11(a), during the initial training stage (20-50 epochs), the mAP@50 curve of the proposed model increased more rapidly and was continuously higher than that of the baseline. Therefore, Inner-MPDIoU accelerates feature learning and achieves a faster convergence. Figure 11(b) shows mAP@50-95 with a higher IoU threshold and thus better localisation accuracy. After 150 epochs, the mAP@50-95 curve gradually increased and surpassed that of the baseline, showing improved boundary localisation for small defects.

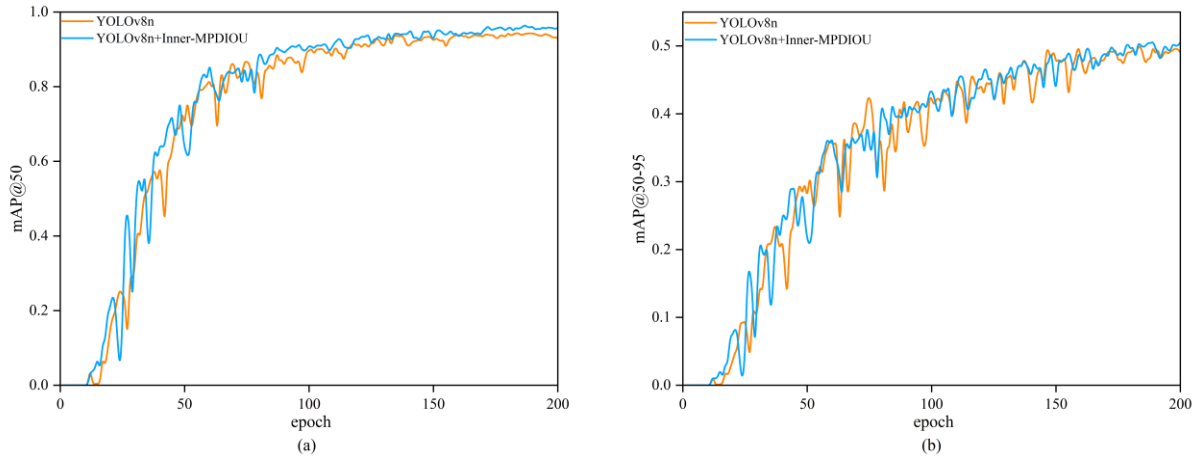

Figure 11: Comparison of Training Processes. (a) mAP@50 curves; (b) mAP@50-95 curves.

3.5 Contrast Experiment

To show the advantages of FHIM-YOLO in more detail, this paper compares it with some other high-performing detectors. As shown in Table 2, FHIM-YOLO had the highest mAP@50 of 95.2% and mAP@50-95 of 51.9% among all the compared methods. The above results are better than those of the older YOLOv5s (90.8%/44.6%) and YOLOv6s (88.6%/43.9%). Compared with a large model such as YOLOv6s (18.50M parameters and 45.2GFLOPs), FHIM-YOLO has reduced the number of parameters by 94.9% and computation by 65.5% while maintaining a high mAP@50 of 95.2%. In short, according to the experimental results, FHIM-YOLO has good detection accuracy and is relatively fast.

Table 2: Comparison of the Performance of Several Methods.

Model	Params/M	mAP@50/%	mAP@50-95/%	FLOPs/G
YOLOv3-tiny	8.68	90.2	43.7	12.9
YOLOv5s	7.03	90.8	44.6	15.8
YOLOv6s	18.50	88.6	43.9	45.2
YOLOv7-tiny	6.02	40.1	15.1	13.1
YOLOv8n	3.01	89.1	47.4	8.1
YOLOv9-t	2.62	77.4	36.7	10.7
YOLOv10n	2.70	78.9	39.0	8.2
FHIM-YOLO	0.95	95.2	51.9	15.6

To study the generalisation and robustness of the model further, this paper also conducted an experiment on the publicly available DeepPCB dataset. As shown in Table 3, FHIM-YOLO has achieved comparable or slightly better performance on DeepPCB compared to YOLOv8n, and there is no significant drop in cross-dataset transfer.

Table 3: Comparison Results on DeepPCB.

Model	Recall/%	mAP@50/%	mAP@50-95/%
YOLOv8n	93.1	97.7	74.2
FHIM-YOLO	94.0	98.2	74.5

3.6 Analysis of Detection Results

Figure 12 shows the qualitative comparison of the three representative minor defects. FHIM-YOLO increased the confidence of a certain detection in the short-circuit defect from medium (0.77) to high (0.99) for short-circuit defects. FHIM-YOLO has also successfully detected the short- and open-circuit instances that the baseline failed to find with high confidence scores of 0.98 and 0.96, respectively. The base line also had a false-positive that mistook mouse bites for an open circuit. FHIM-YOLO corrected the error to a high-confidence value of 0.98. Therefore, based on the above qualitative analysis, the new model reduces both false detections and missing detections and offers more reliable confidence scores.

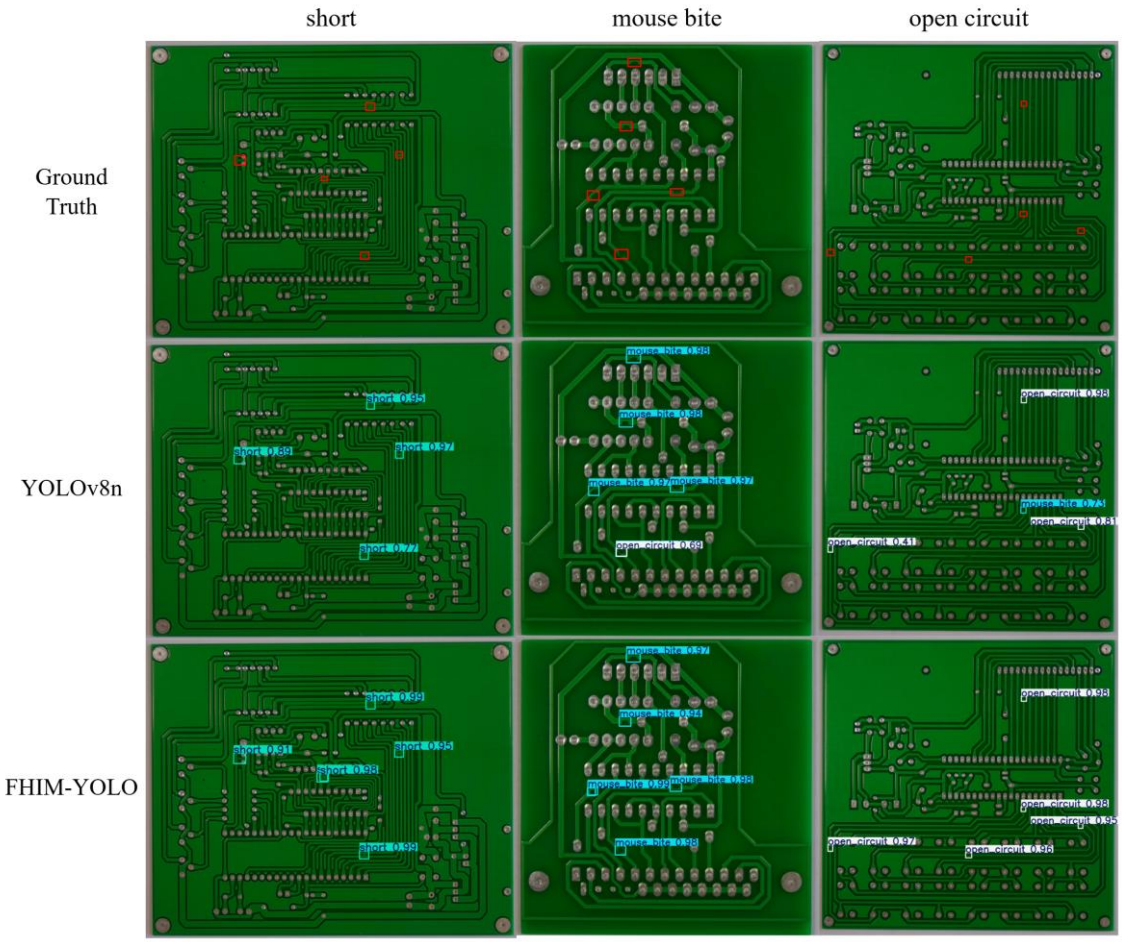

Figure 12: Qualitative Visualization of Detection Performance: YOLOv8n vs. FHIM-YOLO on the HRIPCB dataset.

3.7 Mechanism Analysis and Interpretability Discussion

In order to understand why FHIM-YOLO performs better than the benchmark model more deeply, this section will conduct a theoretical analysis of the internal operating mechanism of the model based on experimental results.

3.7.1 Evolution of Receiving Domains in FEM

Standard convolution in YOLOv8n often has the problem of a limited reception domain, and as a result, the network fails to maintain global context continuity for small targets, such as open circuits. Optimised FEMs have been employed to construct a balanced hierarchical reception domain for the network. Specifically, by setting multiple expansion rates (ratio = 2, 3), a dense sampling mode is achieved that can effectively cover the typical spatial range of microdefects without the problem of sparse sampling. The 10.9% increase in mAP for small targets is a direct result of this reception-domain optimisation; therefore, even minute pixel variations in texture can be incorporated into local contextisation.

3.7.2 Optimization of the Signal-to-Noise Ratio in HS-FPN.

PCB images show a high-frequency background noise due to copper wires and screen printing. This noise in the benchmark PAN-FPN can reduce the strength of a small defect's signal. The HS-FPN in this paper serves as a semantic filter. Channel Attention in the 'feature selection'

module suppresses uncorrelated background channels and improves the signal-to-noise ratio of the feature map. This way has reduced class confusion substantially; now, the model no longer considers short circuits false alarms and has learned to distinguish the morphological "signals" specific to each class from the noisy background.

3.7.3 Feature Redundancy and Information Flow in C2f_Faster

The new C2f_Faster also has a lower number of parameters. PConv is used to reduce feature redundancy that occurs in deep CSP structures in FHIM-YOLO. Many convolution kernels in the standard C2f blocks learn highly correlated and redundant features, thus wasting memory bandwidth and generating noise. C2f_Faster pushes the network to learn a more compact and non-redundant representation. Therefore, a bottleneck occurred in the information; consequently, the model focused only on the most discriminatory attributes, and thus Model VIII retained a high accuracy of 95.2% after reducing the parameter count by 68.4%. Therefore, it will not overfit the background texture and can instead generalize according to the geometry of the defect.

4 Conclusion

The first problem that needs to be solved in this paper is how to balance the requirements for high accuracy, a small-sized design, and real-time operation in detecting small PCB defects in automated manufacturing. Therefore, FHIM-YOLO is put forward, which has added an improved feature enhancement module, a lightweight C2f_Faster module, a modified neck architecture based on HS-FPN, and a good localization loss function (Inner-MPDIoU). On the HRIPCB dataset, FHIM-YOLO reaches a top-1 mAP@50 of 95.2% with only 0.95 million parameters and is 6.1 percentage points higher than YOLOv8n; at the same time, it maintains a real-time inference speed of 65.19 FPS. DeepPCB also has a stable gain and may be more generalisable across different datasets.

Our evaluation still has some limitations; mainly due to the use of public datasets, we have not covered all the variations in real-world production conditions, such as different images, devices, and noise interference. In the future, we will build and annotate more industrially representative datasets, carry out cross-domain and cross-device generalization studies, and improve the efficiency and robustness of edge deployment further by combining model compression with hardware-aware optimisation. Several Random Seeds will be used in the experiment to test the training stability under a highly variable deployment environment.

Author Contributions

Conception, H.C.; Methodological design, H.C.; Software Development, H.C.; Validation and verification, H.C.; Formal analysis, H.C.; Data collection and curation, H.C.; Original writing, H.C.; Review and revision of written content, H.C.; Visualisation, H.C. The author has read and agreed to the published edition of the manuscript.

Finance

This study did not receive other funding.

Institutional Review Board Statement

Not Applicable.

Informed Consent Statement

Not Applicable.

Data Availability Statement

The datasets are available to the public.

TopEdit LLC has provided language editing and proofreading services for the preparation of this paper.

Conflicts of Interest

No conflict of interest.

Disclaimer/Publisher's Note

References

[1] Li, X., Xu, Y., Ji, Z., Zhang, Y., Chen, M., Wang, S., & Zhao, Q. (2025). A lightweight cross-axis semantic interaction network with receptive-field-based attention for industrial surface defect detection. Expert Systems with Applications, 285, 127949.

[2] Chen, Y., Ding, Y., Zhao, F., Zhang, E., Wu, Z., & Shao, L. (2021). Surface defect detection methods for industrial products: A review. Applied Sciences, 11(16), 7657.

[3] Lv, S., Liang, T., Zhang, K., Jiang, S., Ouyang, B., Li, Q., & Li, X. (2025). A lightweight hierarchical aggregation task alignment network for industrial surface defect detection. Expert Systems with Applications, 263, 125727.

[4] Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137-1149.

[5] Ding, R., Dai, L., Li, G., & Liu, H. (2019). TDD-Net: A tiny defect detection network for printed circuit boards. CAAI Transactions on Intelligence Technology, 4(2), 110-116.

[6] Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhaija, B., & Heming, J. (2023). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in

the era of big data. Information Sciences, 622, 178-210.

[7] Liu, G., & Wen, H. (2021). Printed circuit board defect detection based on MobileNet-YOLO-Fast. Journal of Electronic Imaging, 30(4), 043004.

[8] Kang, L., Ge, Y., Huang, H., & Zhao, M. (2022). Research on PCB defect detection based on SSD. In Proceedings of the 2022 IEEE 4th International Conference on Civil Aviation Safety and Information Technology (ICCASIT) (pp. 1315-1319). IEEE.

[9] Liu, X., Hu, J., Wang, H., Zhang, Z., Lu, X., Sheng, C., Song, S., & Nie, J. (2022). Gaussian-IoU loss: Better learning for bounding box regression on PCB component detection. Expert Systems with Applications, 190, 116178.

[10] Tang, J., Liu, S., Zhao, D., Tang, L., Zou, W., & Zheng, B. (2023). PCB-YOLO: An improved detection algorithm of PCB surface defects based on YOLOv5. Sustainability, 15(7), 5963.

[11] Xiao, G., Hou, S., & Zhou, H. (2024). PCB defect detection algorithm based on CDI-YOLO. Scientific Reports, 14, 7351.

[12] Li, Z., Zhan, J., Qu, C., Chen, X., & Zhang, L. (2024). Lightweight PCB defect detection algorithm and deployment based on ASF-YOLO. In Proceedings of the 2024 7th International Conference on Computer Information Science and Application Technology (CISAT) (pp. 36-40). IEEE.

[13] Lou, H., Duan, X., Guo, J., Liu, H., Gu, J., Bi, L., & Chen, H. (2023). DC-YOLOv8: Small-size object detection algorithm based on camera sensor. Electronics, 12(10), 2323.

[14] Dang, L., Huangfu, P., Hou, Y., Liu, Y., & Han, H. (2023). A path aggregation network based on residual feature enhancement for object detection in remote sensing imagery. Remote Sensing Letters, 14(6), 598-608.

[15] Ling, Q., Isa, N. A. M., & Asaari, M. S. M. (2023). Precise detection for dense PCB components based on modified YOLOv8. IEEE Access, 11, 116545-116560.

[16] Chen, P., & Xie, F. (2023). A machine learning approach for automated detection of critical PCB flaws in optical sensing systems. Photonics, 10(9), 984.

[17] Zhang, T., Zhang, J., Pan, P., & Zhang, X. (2024). YOLO-RRL: A lightweight algorithm for PCB surface defect detection. Applied Sciences, 14(17), 7460.

[18] Wang, X., Gao, H., Jia, Z., & Li, Z. (2023). BL-YOLOv8: An improved road defect detection model based on YOLOv8. Sensors, 23(20), 8361.

[19] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint, arXiv:2004.10934.

[20] Chen, J., Ji, C., Zhang, J., Feng, Q., Li, Y., & Ma, B. (2024). A method for multi-target segmentation of bud-stage apple trees based on improved YOLOv8. Computers and Electronics in Agriculture, 220, 108876.

[21] Zhang, Y., Ye, M., Zhu, G., Liu, Y., Guo, P., & Yan, J. (2024). FFCA-YOLO for small object detection in remote sensing images. IEEE Transactions on Geoscience and Remote Sensing, 62, 1-15.

[22] Chen, J., Kao, S., He, H., Zhuo, W., Wen, S., Lee, C. H., & Chan, S. H. G. (2023). Run, don't walk: Chasing higher FLOPS for faster neural networks. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 12021-12031). IEEE.

[23] Chen, Y., Zhang, C., Chen, B., Huang, Y., Sun, Y., Wang, C., Fu, X., Dai, Y., Qin, F., Peng, Y., Wang, J., Song, Z., Li, Y., Xu, S., Zhou, Z., Cai, H., Shi, J., Gong, T., & Wang, G. (2024). Accurate leukocyte detection based on Deformable-DETR and multi-level feature fusion for aiding diagnosis of blood diseases. Computers in Biology and Medicine, 170, 107917.

[24] Zhang, H., Xu, C., & Zhang, S. (2023). Inner-IoU: More effective intersection over union loss with auxiliary bounding box. arXiv preprint, arXiv:2311.02877.

[25] Ma, S., & Xu, Y. (2023). MPDIoU: A loss for efficient and accurate bounding box regression. arXiv preprint, arXiv:2307.07662.

[26] Huang, W., Wei, P., Zhang, M., & Liu, H. (2020). HRIPCB: A challenging dataset for PCB defects detection and classification. The Journal of Engineering, 2020(13), 303-309.

[27] Shi, P., Zhang, Y., Cao, Y., Sun, J., Chen, D., & Kuang, L. (2025). DVCW-YOLO for printed circuit board surface defect detection. Applied Sciences, 15(1), 327.

[28] Tang, S., He, F., Huang, X., & Yang, J. (2019). Online PCB defect detector on a new PCB defect dataset. arXiv preprint, arXiv:1902.06197.

[29] Deng, C., Zhang, Y., Wu, Z., Wu, Y., Sun, X., & Wang, S. (2025). EH-YOLO: Dimensional transformation and hierarchical feature fusion-based PCB surface defect detection. Applied Sciences, 15(20), 10895.