ITC 2/55
Information Technology and Control
Vol. 55 / No. 2/ 2026
pp. 511-530
DOI 10.5755/j01.itc.55.2.43456

YOLO-iDAT: An Improved YOLOv11 for PCB Defect Detection

Received 2025/10/29 | Accepted after revision 2026/04/07

HOW TO CITE: He, H. (2026) YOLO-iDAT: An Improved YOLOv11 for PCB Defect Detection. *Information Technology and Control*, 55(2), 511-530. https://doi.org/10.5755/j01.itc.55.2.43456

YOLO-iDAT: An Improved YOLOv11 for PCB Defect Detection

Hanyang He*
School of Huaxin Software, Tianjin University of Technology, 300387 Tianjin, China

Corresponding author: hehanyanng@outlook.com

Printed Circuit Boards (PCB) are a fundamental component of electronic devices, serving to connect and support electronic components. They are widely used in various fields, including Consumer Electronics, Industrial Control, and Aerospace. However, modern PCBs are trending toward miniaturization, lightweight design, and high density. Consequently, tiny surface defects appear more frequently. Defects such as missing holes, mouse bites, and open circuits exhibit low contrast, making them challenging to detect. Traditional inspection methods suffer from several limitations. Manual visual inspection and electrical testing are inefficient. They suffer from low accuracy, poor efficiency, and high costs. The automated optical inspection (AOI) still needs people's assistance. In addition, the early deep learning models like the initial versions of YOLO do not accurately detect small defects. During upsampling, these models lose fine-grained details and fail to adapt to unbalanced data. Consequently, they often fail to detect low-contrast defects. To address these issues, this paper proposes YOLO-iDAT. It is an improved version of YOLOv11. First, this paper changes the C2PSA module into C2PSA_iRMB. This new module combines the iRMB structure, SE attention and residual connections. It makes the model mix the global and local information better. And it significantly enhances the ability to extract the tiny defect feature. Second, this paper uses DySample for up sampling. It is a dynamic method. It prevents the blurring of small defects that often occurs with conventional upsampling methods. It keeps the feature clear. And it does not make the model slow. Third, this paper uses the Adaptive Threshold Focal Loss (ATFL) function. It uses an adaptive threshold and adjusts parameters to adapt to unbalanced data. It significantly reduces the number of missed defects. This paper uses the proposed model to test the public PKU-Market-PCB dataset. The results show that YOLO-iDAT gets a precision of 0.941, the mAP50 of 0.93 and the recall of 0.895. These represent improvements of 0.9%, 1%, and 1.8%, respectively, over the original YOLOv11. And the speed of our model is very fast. It gets FPS of 88.76. This is a sufficient speed for real-time detection in factory settings. These results demonstrate that YOLO-iDAT fits the requirement of industrial PCB defect detection. It provides a more accurate and more efficient solution for PCB defect detection.

KEYWORDS: PCB defect detection; C2PSA_iRMB; DySample; Adaptive Threshold Focal Loss

1. ntroduction

PCBs are the foundation for connecting electronic components and supporting electronic products. With the continuous evolution and upgrading of electronic products, the quality of PCBs directly affects the performance, reliability, and service life of final products [29]. From consumer electronics to manufacturing industries, including industrial control, aerospace and military products, all electronic systems integrated with components use PCBs as the basic support [9]. The development of Industry 4.0 drives PCBs towards miniaturization, lightweight and high density, putting higher demands on their manufacturing process. Thus, PCB defect detection is crucial. Any defect (such as missing holes, mouse bites, or open circuits) existing on the PCB will affect the equipment performance and ultimately prevent the realization of equipment functions [12]. Therefore, researching reliable PCB defect detection methods is essential.

Traditional PCB defect detection mainly includes manual visual inspection and electrical testing [6]. Manual inspection relies on operators' naked eyes and experience (aided by microscopes/magnifying glasses), suffering from low accuracy, discontinuous defect identification and difficult data collection due to subjective judgment. Electrical testing uses probes to detect faults but requires specific needle beds, leading to high costs and inability to meet large-scale production efficiency. Therefore, the traditional PCB defect detection method has many limitations and cannot meet the requirements of industrial application.

To solve these problems, AOI technology has been developed to overcome traditional limitations. With advantages of automation, high precision and non-contact detection [19], it has become an important technical development direction in PCB defect detection. According to the comparison method, AOI method can be divided into three categories: reference comparison method, non-reference comparison method and hybrid method [15]. The reference method compares defect-free sample data with the tested object, with simple hardware and high pixel rate but alignment and color difference issues and large feature matching storage. The non-reference comparison method needs no reference patterns, with high speed and low storage but only identifies partial defects and requires standardized conductor tracks. The hybrid method combines the reference comparison method and the non-reference comparison method. It can identify more types of defects, but the amount of computation is increased and the detection steps are complicated [20].

Although AOI systems reduce labor dependency in factories, the PCB industry still needs manpower to cooperate with AOI systems for quality inspection. To improve detection accuracy and speed while reducing labor costs, many researchers have devoted themselves to the development of detection algorithm based on traditional image processing or machine vision [9]. With the rapid advancement of deep learning, its algorithms have revolutionized computer vision. Deep learning algorithms based on Convolutional Neural Network (CNNs) [7], Recurrent Neural Network (RNNs) [24] and Generative Adversarial Network (GANs) [2] efficiently analyze data features and achieved significant breakthroughs in image recognition and other fields. The Transformer architecture, via self-attention, has further advanced visual benchmarks and promoted multimodal integration. Recently, Liu et al. [10] revisited the multi-scale feature hierarchy within the DETR framework, proposing to introduce a multi-branch structure into the decoder to alleviate computational bottlenecks in the encoder. Yang et al. [21] designed a modal fusion vision Transformer for hyperspectral and LiDAR heterogeneous modality fusion. It generates queries via LiDAR self-attention and uses hyperspectral features as keys/values, thereby reducing cross-modal alignment requirements. Raman et al. [17] combined panoramic segmentation with monocular depth estimation, proposing the PODE network to predict occlusion relationships at the instance level using depth.

These technological advancements provide new approaches for PCB defect detection, helping overcome traditional inspection limitations. In 2021, Kim et al. [6] designed a PCB defect detection system based on skip-connected convolutional autoencoder. By addressing the issues of small sample size and class imbalance in the training and validation sets via image preprocessing and data augmentation, the system achieved better image reconstruction and defect classification performance than ordinary convolutional autoencoders, and the computational amount was lower and real-time inference was available. In 2022, Yu et al. [23] proposed a PCB defect detection method based on data augmentation of Self-Attentive Generative Adversarial Network (SAGAN) and ResNet18. By solving the problem of unbalanced sample distribution, the method reached high accuracy. In 2025, Luo et al. [14] proposed a rotation angle-based principal feature extraction and optimization method to address PCB defect detection under uncertain conditions. This method is implemented through operations such as feature extraction and angle optimization. The method has advantages of small amount of parameters, fast operation speed and excellent detection effect. In 2025, Guo et al. [3] proposed CM-UNetv2, which significantly enhances the accuracy of PCB defect boundary restoration and small object segmentation through parallelized block-aware attention modules, a dual-stream skip-guided mechanism, and a frequency-domain context-guided Mamba decoder.

Among deep learning-based PCB defect detection technologies, end-to-end single-stage object detection networks perform remarkably. The YOLO series, a representative of such networks, is famous for efficient detection and widely used in PCB defect detection. It has evolved into mature versions (YOLOv5, YOLOv8, YOLOv11) that enhance feature extraction and generalization for complex patterns, solidifying its irreplaceable role [5]. In 2023, Zhou et al. [28] proposed MSD-YOLOv5 model. By combining MobileNet-v3 network with CSPDarknet5 network and replacing coupled detection head with decoupled detection head, the model's parameters are reduced by 46% and the detection accuracy is improved by 3.34%. In 2024, Xiao et al. [20] proposed CDI-YOLO-based PCB defect detection model. Built on YOLOv7-tiny, the model integrates the Coordinate Attention (CA) mechanism into defect feature extraction, replaced some ordinary convolutions with DSConv to reduce computational cost, and adopted Inner-CIoU to accelerate bounding box regression and achieved balance between detection accuracy, speed, and parameter quantity. In 2025, Hu et al. [4] proposed YOLOv5s-based ASTKD-PCB-LDD model. It optimizes anchor boxes by k-means++ algorithm, adopted Faster-Ghost as backbone network, used Slim-Neck architecture and SIoU loss function, and adopted LAMP pruning and ASTKD knowledge distillation to achieve efficient PCB defect detection. In 2025, Li et al. [8] proposed EL-PCBNet (C3F-SimAM-YOLO), an efficient and lightweight PCB defect detection model. By designing or introducing dedicated modules, it optimized feature extraction, resisted background interference, reduced false recognition rate, and maintained model lightweight. It can adapt to PCB defect detection requirements with excellent robustness and strong deployment adaptability.

To further improve the detection accuracy and real-time performance of tiny and low-contrast PCB defects (such as missing holes, mouse bites, and open circuits), this study adopts YOLOv11 as the base model and proposes an improved model, YOLO-iDAT. Its core innovation is summarized as follows.

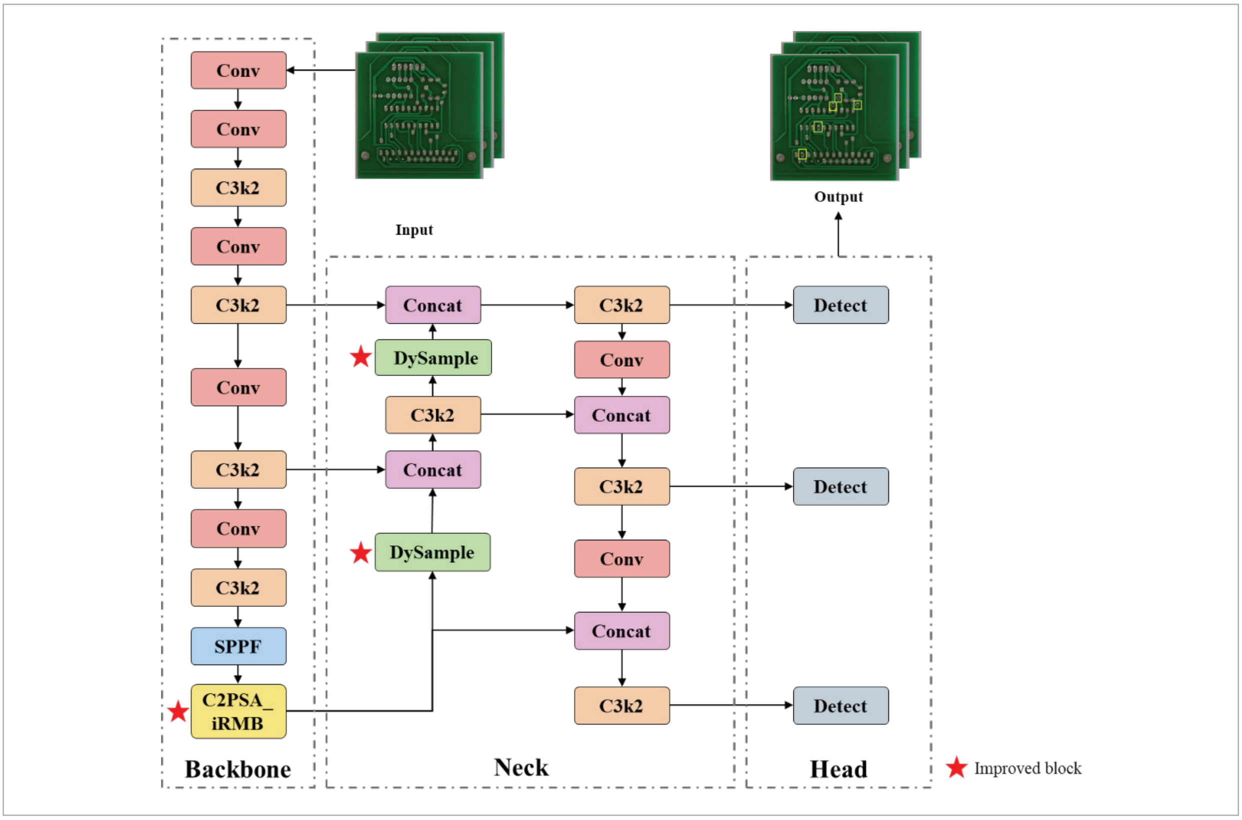
- **Design C2PSA_iRMB module:** The C2PSA module is enhanced by integrating the SE attention mechanism and residual connections into the improved iRMB structure. It can achieve efficient fusion between global dependencies and local detailed features, significantly enhancing the capability to capture key information such as minor edge defects and textures on PCB and addressing the weak detail modeling of the original module.
- **Add DySample dynamic sampler:** A dynamically adjustable sampling strategy is adopted to minimize feature blurring caused by conventional upsampling as much as possible, and keep lightweight inference efficiency while maintaining feature information of tiny defects.
- **Add ATFL function:** Through the method of threshold division and dynamic parameter adjustment, it effectively mitigates model bias induced by class imbalance, enhance the learning weight of low-contrast and minor hard-to-distinguish defects, greatly reduce the miss rate of key defects and improve the detection robustness.

2. Methods

With the current trend toward miniaturization, lightweight design, and high-density integration in PCBs, most defects present challenges in detection due to their small size, low contrast, and strong background interference. Since the original YOLOv11 model in-

Figure 1

Overall Network Structure of YOLO-iDAT.



adequately captures local texture information, defect feature information is prone to distortion during upsampling, and there is a severe imbalance between defect samples and background samples, this study proposes the specific implementation of the improved YOLO-iDAT model to meet the accuracy and real-time performance requirements for PCB defect detection, as illustrated in Figure 1. The improved C2PSA_iRMB module enhances the representation capability of tiny PCB defect regions, preserve defect feature information during upsampling by designing the DySample dynamic upsampler, the ATFL function mitigates the learning bias between defect and background samples, thus constructing an efficient object detection framework suitable for industrial PCB defect detection scenarios.

2.1. Improved C2PSA Module Based on iRMB

The original C2PSA module in YOLOv11 is designed for feature extraction. As shown in Figure 2, its design idea is to combine CSP structure and PSA mechanism. The PSA mechanism can extract cross-level contextual information, which improves the ability of the model to extract multi-scale features. Its functional design focuses on processing spatial dimension information, resulting in the model's weak capability to model local texture features (e.g., edges and tiny objects) and making local detail information prone to loss in detection and segmentation tasks.

To solve this problem, this paper proposes the C2PSA_iRMB module based on the iRMB structure [25]. Retaining the "split-enhance-fuse" paradigm, we replace classic attention with iRMB (Inverted Residual Mobile Block) integrating the improved Expanded Window Multi-Head Self-Attention (EW-MHSA). This hybrid enhancement collaboratively optimizes global dependencies, local details, and channel importance while reducing computational cost.

Our iRMB is inspired by the combination of local modeling ability of CNNs and global correlation

Figure 2

Structure of C2PSA Module.

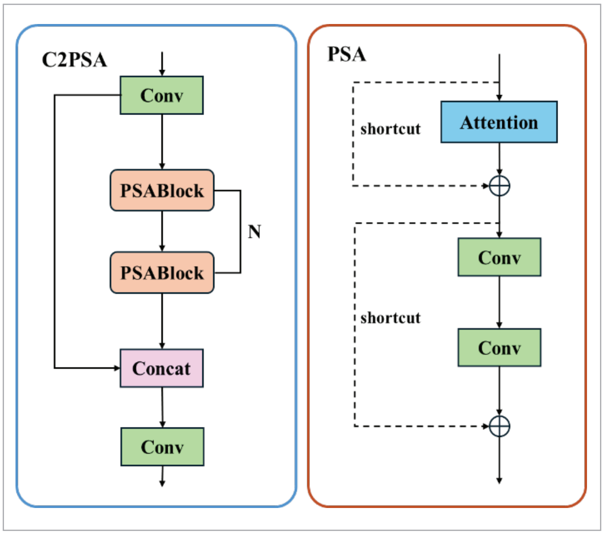
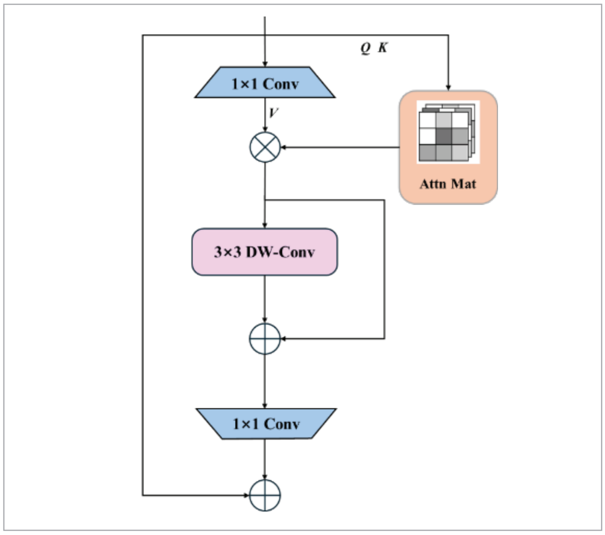


Figure 3

Structure of iRMB Module.



capture ability of Transformers. It enhances the accuracy of feature representation in dense prediction tasks while meeting the requirements for lightweight deployment. In terms of structural design, as shown in Figure 3, iRMB uses the Inverted Residual Block (IRB) of MobileNetv2 as the basic framework and expands its functionality by introducing three key components: First, DW-Conv (Depthwise Convolution): By performing independent convolution on key channels, it can effectively extract local spatial information, avoid cross-channel information interference, and reduce computational complexity. Second, Convolutional layers in channel dimension adjustment: These layers provide sufficient feature representation dimensions for attention computation and local convolution via a two-stage "expansion-shrinkage" transformation. In the expansion stage, the number of channels is increased to λ times the input dimension; in the shrinkage stage, the number of channels is restored to the original. The channel expansion process can be expressed as:

$$X_e = Conv_{1\times1}(X_{norm}) \in R^{B\times\lambda C\times H\times W}, \tag{1}$$

where X_{norm} is the normalized input feature, λ is the channel expansion rate, and B, C, H, W represent the batch size, number of input channels, feature map height, and feature map width, respectively. Third, EW-MHSA: Divides the feature map into fixed-size windows, converting global attention calculation into efficient local window-based operations. It adopts an "asymmetric Q/K/V" generation strategy (generating low-dimensional Q/K from original features and high-dimensional V from expanded features). The attention weight calculation is:

$$Attn(Q, K) = Softmax(\frac{QK^T}{\sqrt{d_h}}) \tag{2}$$

$$X_{attn} = Attn(Q, K) \cdot V, \tag{3}$$

where $Attn$ is the abbreviation of Attention, Q and K are the query and key matrices respectively, and d_h is the attention head dimension. This design reduces quadratic complexity while preserving global semantic correlation information.

Specifically, the feature processing flow of iRMB exhibits a hierarchical characteristic of "dimension adjustment – multi-modal enhancement – residual fusion". First, channel expansion is achieved via convolution to provide sufficient dimensional support for subsequent feature enhancement. Next, EW-MHSA captures long-range semantic dependencies within the window, and DW-Conv refines local spatial textures. The two are cascaded to form a "global-local" dual-path feature enhancement chain. After comparing with CBAM and CoordAttention, the Squeeze-Excite (SE) channel attention mechanism is selected for its superior performance and incorporated to dynamically amplify the response intensity of key channels. The channel weight generation process is expressed as:

$$s = Sigmoid(MLP(GlobalAvgPool(X_{dw}))), \quad (4)$$

where X_{dw} is the output feature of depthwise convolution, and s is the normalized channel weight.

Finally, the channel dimension is reduced through convolution, and residual connections are established with the original input features to ensure the integrity of feature information while alleviating the gradient vanishing problem in deep networks. The residual fusion process is:

$$X_{irmb} = X + DropPath(Conv_{1\times1}(X_{se})), \quad (5)$$

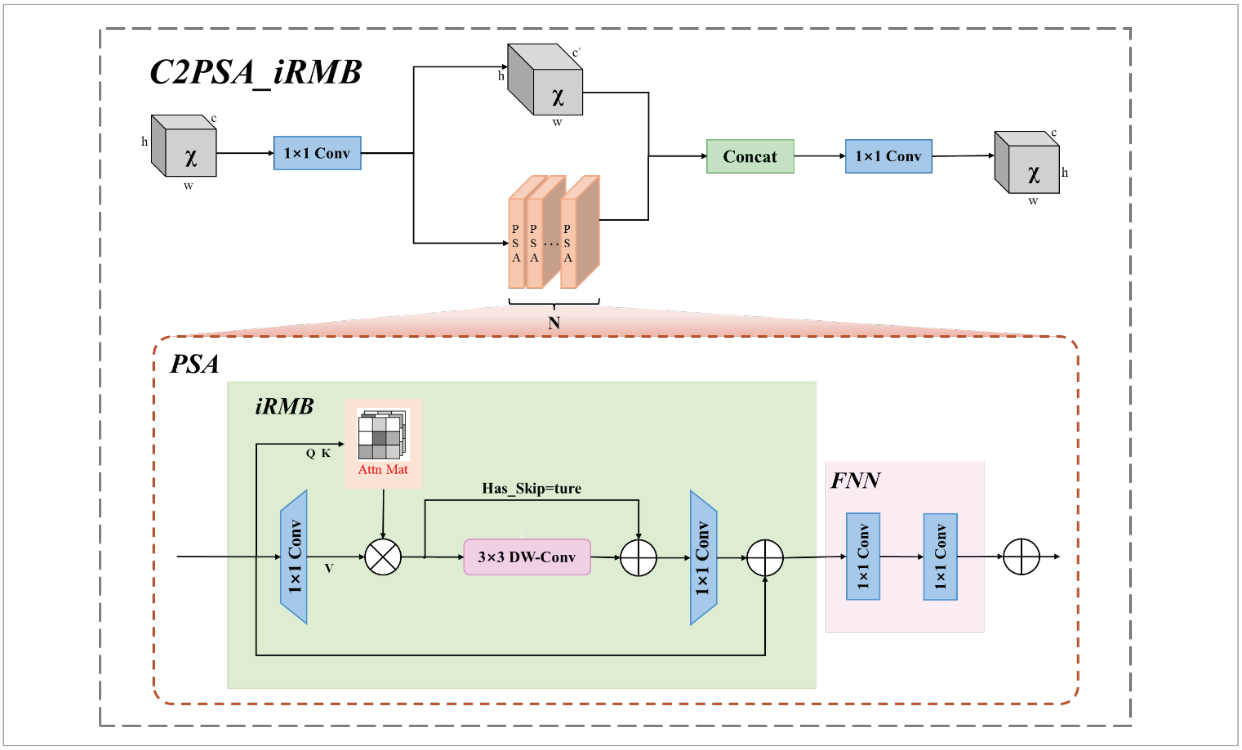
where X_{se} is the output feature of the SE module, and DropPath is a stochastic depth operation to improve model generalization.

Furthermore, compared with the Meta-Mobile Block, the iRMB also possesses the following design idea: Through the configurable expansion rate and operator selection, the module can adapt the structure varying complexity of input images, and then the network could adjust feature processing strategies adaptively according to various visual scenarios.

As shown in Figure 4, in the complete structure of C2PSA_iRMB module, the input feature is first split into two branches in the channel dimension via 1×1 convolution. One branch directly retains the feature for later information preservation. The other branch is achieved by piling N PSAs. Each PSA block consists of an iRMB module and a Feed-Forward Neural Network (FFN), enabling multi-stage feature enhancement. Finally, the features from the two branches are concatenated and fused. A 1×1 convolution is then applied to reduce the channel dimension back to its original size, producing the optimized output feature. It not only inherits the lightweight advantage of the CSP structure but also mitigates the original C2PSA's insufficient focus on channel differences between local details and global infor-

Figure 4

Overall architecture of the C2PSA_iRMB module.



mation by using multi-dimension enhancement ability of iRMB to achieve a large accuracy improvement on PCB defect detection task.

2.2. Upsampling by Dynamic Sampling

DySample is an extremely lightweight and memory-efficient dynamic upsampling operator introduced by Liu et al. [11]. The core design of DySample abandons the paradigm of static interpolation or dynamic convolution kernels and re-design the upsampling from the viewpoint of point sampling. Instead of learning convolution kernels, sampling points are dynamically and directly selected on the feature map, which not only maintains the details of features but also greatly reduces computational and resource overhead. Compared with static upsampling methods such as Bilinear Interpolation, DySample can adjust sampling positions based on feature semantics to avoid feature blurring. Compared with kernel-based dynamic upsampling methods such as CARAFE [11] and FADE [13], it does not require custom CUDA modules and only relies on PyTorch's built-in functions – resulting in significantly fewer parameters and inference time close to bilinear interpolation. It is highly suitable for complex scenarios in PCB detection involving a large number of defect types. The upsampling process of DySample is shown in Figure 5.

The specific implementation logic of DySample is as follows:

1 First, given a low-resolution feature map X with size C×H1×W1, and a sampling set S with size 2g×H2×W2 constructed by the sampling point generator, the PyTorch built-in Grid_sample function is utilized to resample the low-resolution feature map X based on the coordinate information of each point in the sampling set S. Finally, a high-resolution feature map X' with size C×H2×W2 is output:

$$X' = Grid\_sample(X, S), \quad (6)$$

2 The generation of the sampling set S is the key to DySample's dynamic adjustment. Its generation is implemented by the sampling point generator, as shown in Figure 6, which specifically includes two implementation methods: static range factor and dynamic range factor. Taking the static range factor method as an example, for the input feature map X, a linear layer with input channels C and output channels $2gs^2$ is first employed to map the features to initial offset information. To avoid feature ambiguity caused by excessive sampling point offsets, the output result of the linear layer is multiplied by a static range factor of 0.25 to constrain the offset amplitude. Subsequently, a PixelShuffle operation is applied to adjust the spatial dimension of the offset information to match that of the high-resolution feature map, yielding the final offset. Finally, add offset to elements of preset original sampling grid to obtain dynamic sampling set for resampling. The process is as follows:

$$O = 0.25 \times Linear(X) \quad (7)$$

$$S = G + O, \quad (8)$$

where Linear represents the linear layer operation, and 0.25 is the size of the static range factor.

3 Dynamic range factor uses two parallel linear layers. The first linear layer performs a standard linear transformation on the input feature map to learn basic offset information; while the second linear layer is supplemented with a function. Finally, the output results of above two layers

Figure 5

The Upsampling process of DySample.

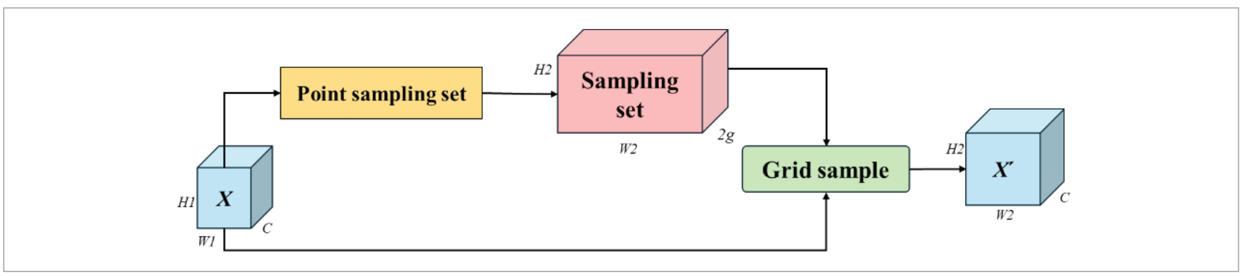
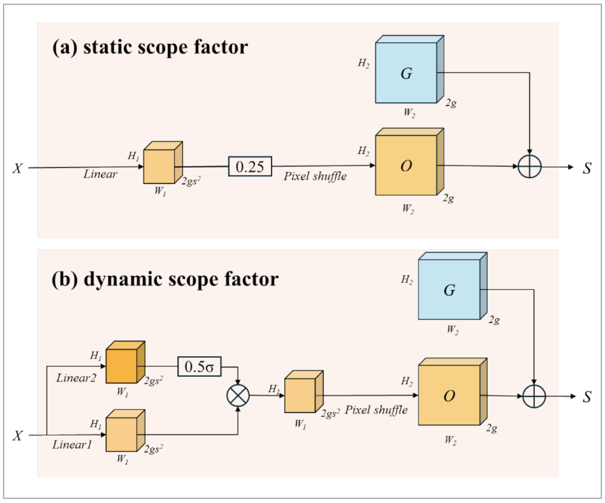


Figure 6

Sampling point generator.



are multiplied together to obtain the final offset, which not only achieves semantic adaptability but also ensures amplitude constraints, thereby enabling more precise adjustment of sampling positions. The process is as follows:

$$O = Linear_1(X) \cdot 0.5sigmoid\left(Linear_2(X)\right), \quad (9)$$

To address the issues of feature blurring for tiny PCB defects and excessive computational overhead caused by traditional upsampling methods. DySample rebuilds the upsampling process and generates sampling points in a dynamic way. It eliminates the need to learn the convolution kernels and instead relies solely on PyTorch's built-in functions to precisely adjust sampling positions based on feature semantics--it can accurately retain the defect features, make inference efficient and apply to complex PCB defect detection more suitably. The generation of its sampling set can control the offset through static range factors or attain fine adjustment through dynamic range factors, which can guarantee the integrity of features after upsampling further.

2.3. Adaptive Threshold Focal Loss

In PCB defect detection, a core issue similar to infrared small target detection is that defective regions occupy only a tiny proportion of image pixels, while normal background regions are dominant, causing target-background distribution imbalance. Thus, this study introduces the ATFL function to solve this imbalance and enhance feature learning of tiny, hard-to-distinguish PCB defects.

The ATFL function [22] is designed based on classic cross-entropy loss and Focal Loss, and it solves the limitation of traditional loss in imbalanced sample scenarios. First, the classic cross-entropy loss function computes the base loss by measuring the discrepancy between predicted probabilities and true label, as shown in Formula (10).

$$L_{BCE} = -\left(ylog(p) + (1-y)\ log\ log\,(1-p)\right), \quad (10)$$

where p represents the probability that the predicted region is a defect, and y represents the true label. To simplify calculations, the effective prediction probability p_t is introduced, defined as:

$$p_t = \begin{cases} p, & if\ y = 1 \\ 1-p, & others \end{cases}. \quad (11)$$

At this point, the cross-entropy loss can be simplified to $LBCE = -\log(p_t)$. However, cross-entropy loss cannot distinguish the loss weights of easy-to-classify samples and hard-to-classify samples, leading the model to over-learn background features during training. The classic focal loss alleviates this problem by introducing a modulation factor $(1-p_t)^{\gamma}$ (where γ is a fixed focusing parameter). Its expression is:

$$FL(p_t) = -(1-p_t)^{\gamma}\ log\,(p_t)\,. \quad (12)$$

However, the fixed focusing parameter of Focal Loss uniformly synchronously weakens the loss weights of hard-to-classify samples, which is detrimental to the learning of typical hard-to-classify targets (e.g., tiny PCB defects)—this becomes the core starting point for the improvement of the ATFL function.

The ATFL function adapts to the needs of PCB defect detection through two mechanisms: threshold-based sample decoupling and adaptive weight adjustment. Its core lies in dividing samples into easy-to-classify and hard-to-classify categories via a threshold, adopting differentiated loss calculation methods for the two sample types, and dynamically adjusting the focusing parameter to avoid the cost of manual parameter tuning and adaptability issues. Its mathematical expression is:

$$ATFL = \begin{cases} -(\lambda - p_t)^{-ln(p_t)} \, log\,(p_t), p_t \leq 0.5 \\ -(\lambda - p_t)^{-ln(\hat{p}_c)} log\,(p_t), p_t > 0.5 \end{cases}, \tag{13}$$

where λ is a hyperparameter used to amplify the loss weight of hard-to-classify samples, and $\hat{p}_c$ is the average predicted probability of the next round, used to dynamically adjust the focusing parameter of easy-to-classify samples. Its calculation formula is:

$$\hat{p}_c = 0.05 \times \frac{1}{t-1} \sum_{i=0}^{t-1} \overline{p}_i + 0.95 \times p_t\,, \tag{14}$$

where $\hat{p}_i$ is the average predicted probability of PCB defect regions in the first t-1 training rounds, p_t is the average predicted probability of the current round, and the weight coefficients of 0.05 and 0.95 are used to balance historical and current training information, making more stable and preventing individual abnormal samples from affecting the loss calculation in a single training round.

Standard Focal Loss effectively suppresses easy samples via its fixed modulation factor, but this same mechanism excessively suppresses gradients for extremely difficult samples – a limitation particularly evident in PCB defect detection where hard samples are rare yet critical.

ATFL can solve this problem to a certain extent. ATFL separates easy/hard samples via the pt threshold and employs dynamic parameter adjustment: for hard samples ($p_t \leq 0.5$), the $(\lambda - p_t)^{-ln(pt)}$ modulation term significantly amplifies loss weights when pt is small, enabling prioritized learning of subtle defect features; for easy samples ($p_t > 0.5$), mild modulation is retained. Meanwhile, ATFL dynamically adjusts focus intensity through weighted fusion of historical and current average prediction probabilities ($\hat{p}_c$), avoiding gradient oscillation caused by a fixed γ. This design reinforces hard sample learning without losing foundational features of easy samples, making it particularly suitable for PCB scenarios characterized by ″low proportion of hard samples and weak features,″ thereby improving optimization effectiveness and convergence stability.

3. Experiment

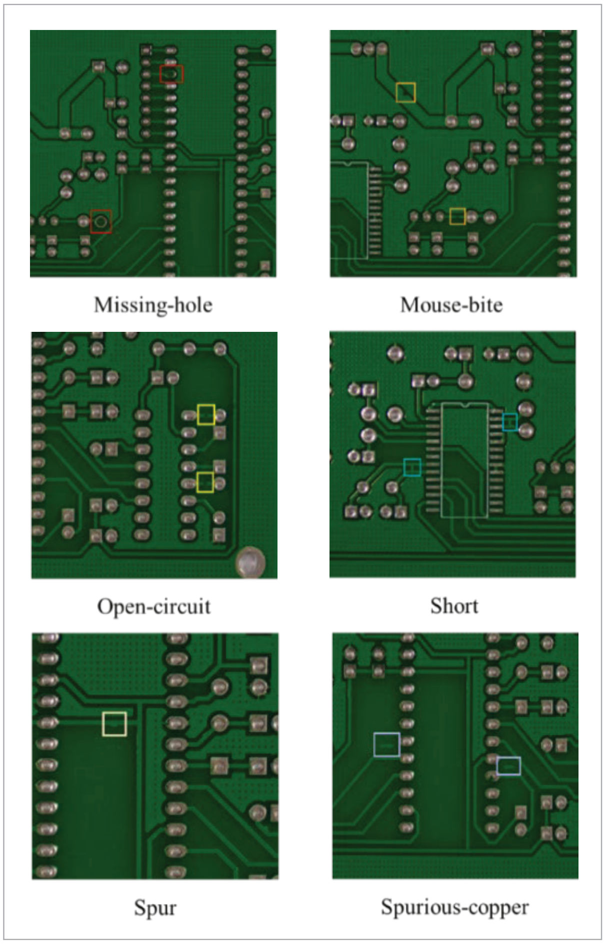
3.1. Data Preprocessing

This study utilizes the public PKU-Market-PCB dataset [16], released by the Intelligent Robot Open Laboratory of Peking University. The dataset contains industrial camera-captured bare PCB images, including 1386 defective ones with six common defects: missing hole, mouse-bite, open circuit, short circuit, spur, and spurious copper (as illustrated in Figure 7). For experiments, 693 detection-suitable images were randomly sampled and split into a training set (593 images) and a validation set (100 images).

To address small training dataset constraints, enhance model robustness, and avoid noise for minute/low-contrast defects, we adjusted three key parameters: hue,

Figure 7

PKU-Market-PCB dataset.



brightness, and rotation angle. All augmentations were applied during training using YOLO's built-in methods. Cross-validation was conducted to evaluate model stability and generalization under different data partitions; results are presented in Section 4.1.

3.2. Experimental Environments

The experiment's training and testing environment is configured as follows: Python 3.12, PyTorch 2.7.1+cu128, Windows 11, AMD Ryzen 9 8945HX CPU (with integrated Radeon Graphics), 16GB memory, and NVIDIA GeForce RTX 5060 GPU (8GB memory). Specific model training parameters are shown in Table 1.

Table 1

Experimental environment and hyperparameter configuration.

Parameter	Value
Input Image Size	640×640
Epochs	100
Batch Size	16
Optimizer	SDG
Workers	4
Learning Rate	0.01

3.3. Evaluation Metrics

Network performance evaluation mainly relies on mean average precision (mAP) monitoring during training and comprehensive validation on the validation set after training. Thus, this study uses Precision, Recall, mAP, and FPS as the primary metrics to evaluate model performance, with their calculation formulas as follows:

$$Precison = \frac{TP}{TP+FP} . \tag{15}$$

Precision: It is the ratio of the number of samples predicted as positive by the model and the number of positive samples.

$$Recall = \frac{TP}{TP + FN} . \tag{16}$$

Recall: The ratio of the number of actually positive samples predicted as positive by the model to the number of positive samples.

$$AP_i = \int_0^1 P_i(R_i) d R_i = \sum_{k=0}^{n-1} P_i(k) \Delta R_i(k) = \frac{\sum P_i}{\sum R_i} \tag{17}$$

$$mAP = \frac{1}{C} \sum_{i=1}^{C} AP_i . \tag{18}$$

mAP: It is a commonly used evaluation metric in object detection, and it is the average value of AP value of all the defect categories. It reflects the average precision of the algorithm at different recall rate, where C represents the number of categories.

FPS: It is used to evaluate the speed of the model detection, which represents the number of images processed in 1 s.

4. Results and Discussion

4.1. Detection Results

To systematically analyze the detection metrics of YOLO-iDAT model on various types of PCB defects, the model was trained on the training set in turn until convergent. Table 2 presents the detection results for each category, quantifying the model's performance in detecting six typical PCB defects

Table 2

Category-specific detection results.

Metric	Category					
	Missing hole	Mouse bite	Open circuit	Short	Spur	Spurious copper
Precision	0.968	0.965	0.904	0.932	0.964	0.914
Recall	0.973	0.811	0.88	0.969	0.816	0.918
mAP_{50}	0.994	0.876	0.901	0.982	0.887	0.939
mAP_{50-95}	0.642	0.442	0.503	0.48	0.40	0.515

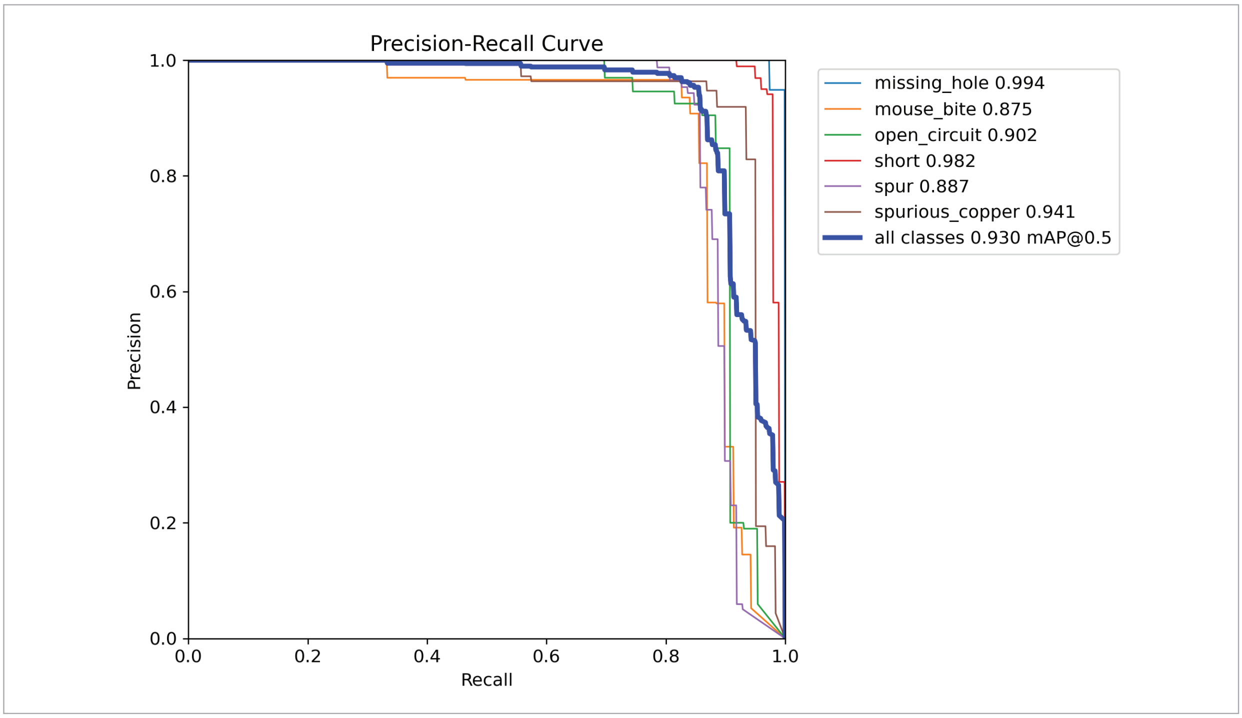
As shown in Table 2, YOLO-iDAT achieves varying detection performance on PCB defects with distinct feature attributes. Generally, it has high detection accuracy on PCB defects with obvious features and good adaptability on PCB defects with weak features, and the detection metrics of all categories satisfy the requirements of industrial-grade PCB defect detection. Specifically, missing holes and shorts have the best detection effect, owing to the prominent visual features of missing holes and short circuits. Missing holes present regular circular features and clear pixel contrast, and therefore the spatial positioning and category determination accuracy of missing holes for the model are the highest: P = 96.8%, R = 0.973, and mAP_{50} are further improved to 0.994, approaching the theoretically optimal detection performance optimal detection level. Due to the large difference in the visual part of metal connection area and normal circuits, the models can capture connectivity features of shorts through the improved C2PSA_iRMB module efficiently: P = 0.932, R = 0.969, and mAP_{50} are stably maintained at 0.982, which also proves that C2PSA_iRMB can effectively capture local distinct features and spatial position information.

Unlike that, the detection accuracy of mouse bites and spurs are a slightly lower. This is because mouse bite defects feature irregular edges and are easily confused with PCB circuit texture, and spur defects are slender metal spur with small pixel proportion, so the model misses a lot of spur defects in cases of complex background. However, their P values are still above 0.96. This means that the model still has a good control on the false detection rate of these two defects (always <4%), and it will not mistakenly recognize normal regions as defects due to weak features. This aligns well with the design of DySample dynamic sampler (preserve detailed features) and ATFL (balance sample weights).

Open circuit and spurious copper defects reflect that the model's feature extraction capability for these two defect types is moderate. Open circuits have the feature of circuit break gaps: P reaches 90.4%, mAP_{50} is 0.901, but R is 88% due to the influence of some tiny break gaps. Spurious copper has weak differences in color and texture from normal copper foil, requiring deep feature semantics for distinction. Through the channel enhancement mechanism of the C2PSA_iRMB module, the model achieves a P of 91.4% and an

Figure 8

PR curves of YOLO-iDAT detection results.



R of 91.8%—demonstrating good performance in deep semantic modeling of such tiny features.

As shown in Figure 8, the PR curves visually verify the category-specific detection performance of the YOLO-iDAT model from a visual perspective. The PR curves of all six defect categories are generally close to the upper-right corner of the coordinate system, and the curves are smooth without sharp fluctuations – indicating stable model training and no abnormal fluctuations in detection performance, with strong generalization ability.

To improve the robustness of the model, we designed two different data augmentation schemes by adjusting three key parameters: hue (hsv_h), brightness (hsv_v), and rotation (degrees). Scheme 1 uses strong disturbances (hsv_h=0.03, hsv_v=0.5, degree=±30°) to simulate significant scene changes; Scheme 2 uses slight disturbances (hsv_h=0.015, hsv_v=0.4, degree=±15°) to match common PCB imaging conditions, as shown in Table 3.

Under Scheme 1, the proposed model's mAP50 decreased by 1.2% compared to the baseline, exhibiting reduced stability – likely due to occlusion or distortion of defect features caused by stronger perturbations. Under Scheme 2, the proposed model achieved a significant 2.7% improvement in mAP50, with both models gaining Recall. This indicates that mild perturbations provide beneficial diversity while preserving the integrity of minor defect features.

To further evaluate the model's stability and generalization capability under different data partitions, we conducted 5-fold cross-validation. The cross-validation results are shown in Table 4.

Table 3

Comparison of Data Augmentation Methods.

Data augmentation	Model	P	R	mAP_{50}	mAP_{75}	$mAP_{50\text{-}95}$
Scheme 1	yolo11n	0.962	0.817	0.878	0.386	0.446
	ours	0.934	0.824	0.866	0.334	0.427
Scheme 2	yolo11n	0.935	0.873	0.91	0.431	0.479
	ours	0.954	0.898	0.937	0.456	0.502

Table 4

Cross-validation results.

Fold	Model	P	R	mAP_{50}	$mAP_{50\text{-}95}$	FPS
1	yolo11n	0.930	0.875	0.918	0.488	90.50
	ours	0.938	0.892	0.928	0.495	88.20
2	yolo11n	0.934	0.880	0.922	0.492	91.20
	ours	0.942	0.897	0.932	0.498	88.90
3	yolo11n	0.928	0.872	0.916	0.486	90.80
	ours	0.940	0.894	0.929	0.496	88.50
4	yolo11n	0.935	0.878	0.921	0.490	91.00
	ours	0.943	0.896	0.931	0.499	88.70
5	yolo11n	0.931	0.876	0.919	0.489	90.70
	ours	0.941	0.895	0.930	0.497	88.60
Mean ± Std	yolo11n	0.932 ± 0.003	0.876 ± 0.003	0.919 ± 0.002	0.489 ± 0.002	90.84 ± 0.25
	ours	0.941 ± 0.002	0.895 ± 0.002	0.930 ± 0.001	0.497 ± 0.001	88.58 ± 0.24

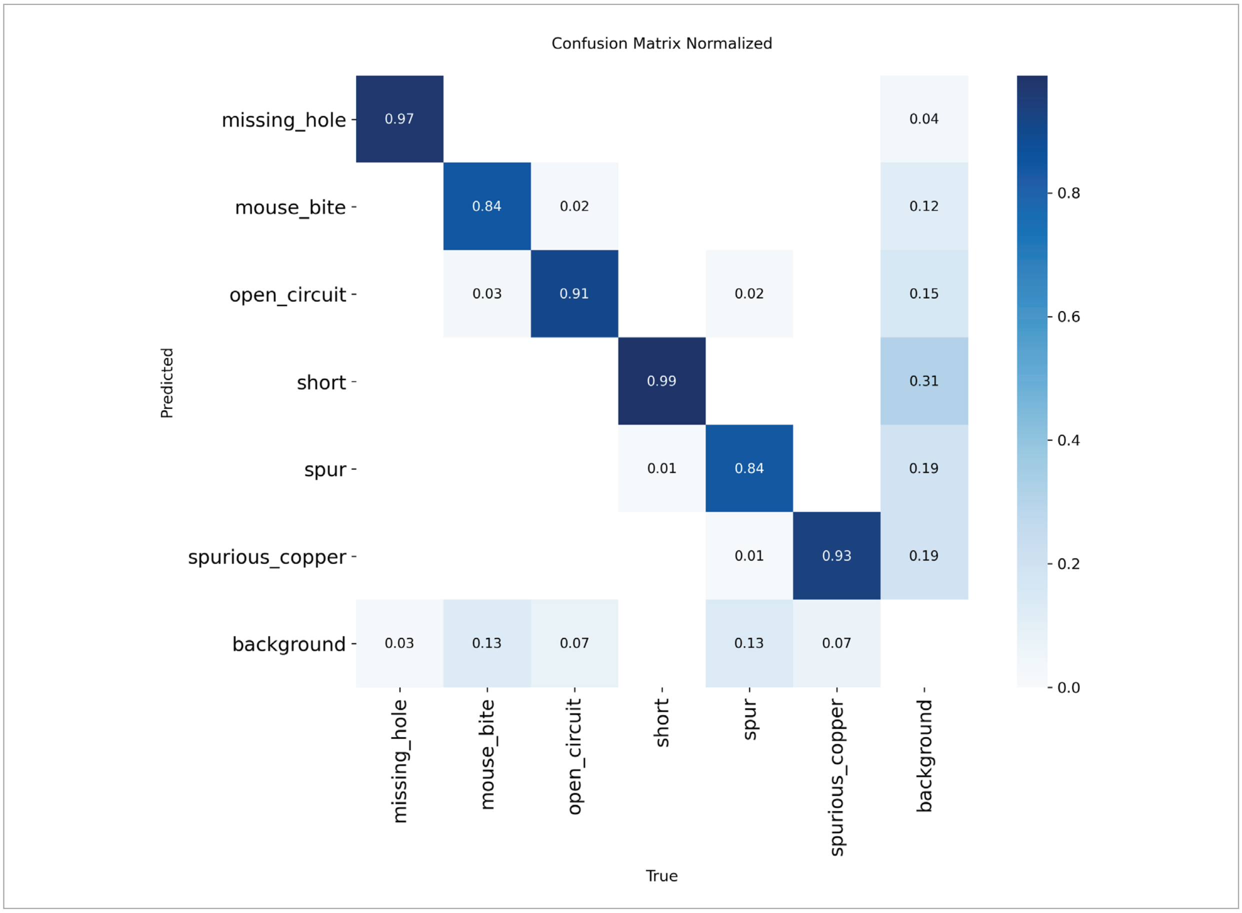
YOLO-iDAT achieved an average mAP_{50} of 0.930±0.001 at 50% confidence level, a Recall of 0.895±0.002, an $mAP_{50\text{-}95}$ of 0.497±0.001, and an FPS of 88.58±0.24. all metrics outperforming the baseline YOLOv11n with minimal standard deviation. This indicates the model's robustness to data partitioning, effectively mitigating risks of overfitting or partitioning bias. These results confirm that the proposed YOLO-iDAT achieves reliable and stable performance, thereby alleviating concerns regarding overfitting or partitioning dependency bias.

As a key tool for evaluating a model's classification performance, the confusion matrix reflects the correspondence between model predictions and true labels, and supports the calculation of evaluation metrics to assess classification effectiveness. As shown in Figure 9, the normalized confusion matrix for YOLO-iDAT indicates that the model achieves high classification accuracy for most PCB defect categories, particularly for defects with distinct features (such as missing holes and short circuits), where its classification performance is outstanding. For defects with relatively complex or weak features (such as rodent bite marks and copper clutter) the model's classification accuracy decreases slightly but remains within an acceptable range. The results of the confusion matrix align with the conclusions regarding model performance derived from analyses based on metrics such as precision and recall, further validating the effectiveness and reliability of YOLO-iDAT in PCB defect detection.

Based on the above analysis, it is evident that YOLO-iDAT effectively balances the two core requirements: "high detection accuracy for prominent defects" and "low false positive rate for subtle defects." Its detection performance is stable and reliable,

Figure 9

Confusion matrix of YOLO-iDAT detection result.



making it suitable for automatic PCB defect detection tasks in industrial environments.

4.2. Comparative Experiments

To further validate the effectiveness of YOLO-iDAT in PCB defect detection, this section compares it with mainstream lightweight YOLO models (YOLOv3-tiny, YOLOv5n, YOLOv8n etc.) and representative Transformer-based (RT-DETR-l) and Mamba-based models (RS-Mamba, ChangeMamba) on the same PCB defect dataset. The results are shown in Table 5.

In terms of Recall, YOLO-iDAT achieves 0.895, matching ChangeMamba and surpassing RS-Mamba (0.888) and RT-DETR-l (0.89), indicating a lower missed detection rate. For mAP_{50}, YOLO-iDAT reaches 0.930, comparable to ChangeMamba (0.932) and significantly outperforming RT-DETR-l (0.9). These results demonstrate that YOLO-iDAT achieves accuracy on par with emerging Mamba-based architectures.

In terms of precision, YOLO-iDAT achieves 0.941, only slightly below YOLOv8n's 0.942. Combined with its higher recall, it delivers superior overall performance. In terms of speed, YOLO-iDAT achieves 88.76 FPS, meeting industrial real-time detection requirements while maintaining significant accuracy improvements over earlier lightweight models.

Regarding computational efficiency, YOLO-iDAT requires only 2.61M parameters and 8.2 GFLOPs—comparable to YOLOv11n and significantly lower than RT-DETR-l (32M params, 103.5 GFLOPs; 12× params, 13× FLOPs), RS-Mamba (27.9M params, 50.2 GFLOPs; 11× params, 6× FLOPs), and ChangeMamba (49.94M params, 28.7 GFLOPs; 19× params, 3.5× FLOPs). This demonstrates that YOLO-iDAT achieves extremely low computational overhead while maintaining competitive accuracy, making it suitable for resource-constrained industrial PCB defect detection scenarios.

4.3. Ablation Experiments

To validate the effectiveness of the proposed YOLO-iDAT model for PCB defect detection – particularly in high-reliability scenarios (e.g., military, aerospace, automotive electronics, and medical equipment) where missed detections may lead to equipment malfunctions, safety incidents, or substantial economic losses, imposing stringent requirements on Recall – the core detection objective is to minimize missed defects. Based on this, ablation experiments were conducted on the PKU-Market-PCB dataset. The results are shown in Table 6. The experiments used YOLOv11 as the base model, and by gradually adding the different modules mentioned earlier, the performance of each model in terms of P, R, mAP_{50}, and FPS was compared to evaluate the role of each module and the effect of multi-module fusion.

YOLOv11 is the base model of this experiment, and its metrics are as follows: P = 0.932, R = 0.877, mAP_{50} = 0.92, and FPS = 91.07. Model_1 is based on YOLOv11

Table 5

Comparison between YOLO-iDAT and other models.

Model	P	R	mAP_{50}	mAP_{50-95}	FPS	Param(M)	FLOPs(G)
YOLOv3-tiny	0.726	0.644	0.693	0.309	**146.52**	12.13	18.9
YOLOv5n	0.94	0.893	0.922	0.504	90.18	2.5	7.1
YOLOv8n	**0.942**	0.875	0.92	0.486	104.96	3.01	8.1
YOLOv9t	0.859	0.73	0.797	0.388	57.12	1.97	7.6
YOLOv10n	0.932	0.854	0.916	0.496	97.53	2.27	6.5
YOLOv11n	0.932	0.877	0.92	0.491	91.07	2.58	6.3
RT-DETR-l [27]	0.905	0.890	0.9	0.450	34.5	32	103.5
RS-Mamba [26]	0.938	0.888	0.928	0.499	68.6	27.9	50.2
ChangeMamba [1]	**0.942**	0.895	**0.932**	**0.505**	51.31	49.94	28.7
YOLO-iDAT(ours)	0.941	**0.895**	0.93	0.497	88.76	2.61	8.2

Table 6

Ablation experiments on key modules in YOLO-iDAT.

Model	C2PSA_iRMB	DySample	ATFL	P	R	mAP_{50}	FPS
Baseline				0.932	0.877	0.92	91.07
Model_1	√			0.935	0.884	0.914	86.69
Model_2		√		0.944	0.879	0.92	91.67
Model_3			√	0.957	0.864	0.925	**95.44**
Model_4	√	√		0.951	0.88	0.925	87.4
Model_5	√		√	0.944	0.872	0.928	90.28
Model_6		√	√	**0.967**	0.887	0.926	92.54
Model_7(ours)	√	√	√	0.941	**0.895**	**0.93**	88.76

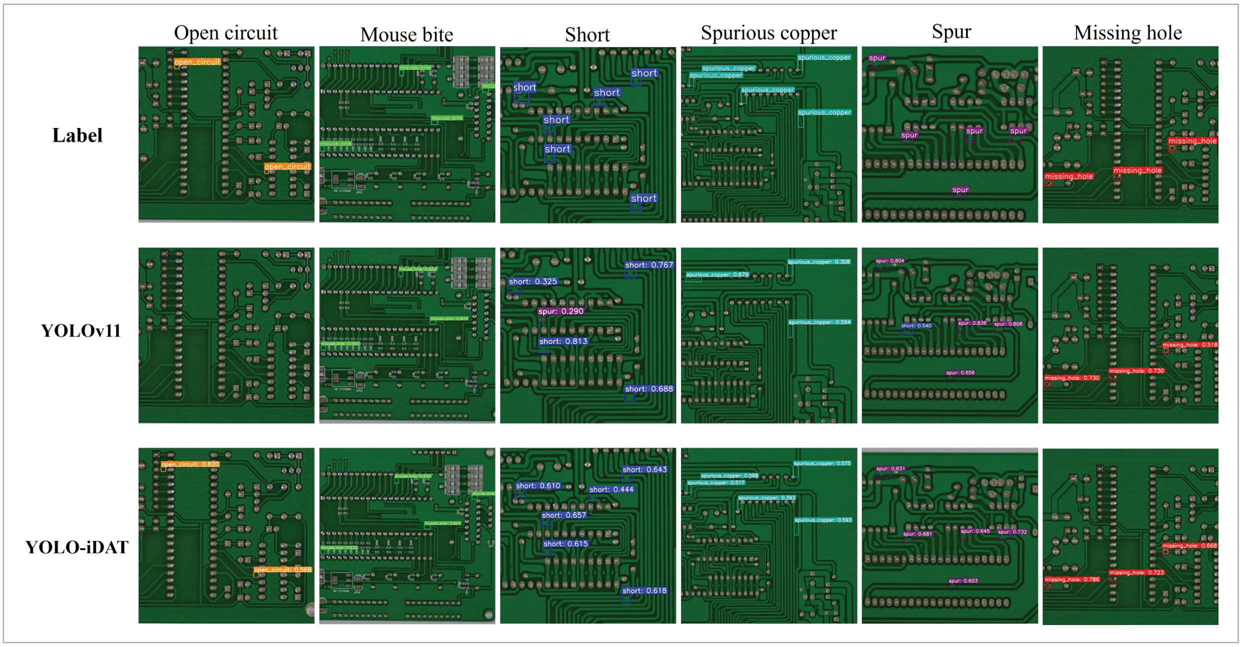
and introduces C2PSA_iRMB module. Compared with Baseline, P increases from 0.932 to 0.935 and R rises from 0.877 to 0.884 – indicating that, compared with Baseline, C2PSA_iRMB module can improve the ability of model to extract and distinguish PCB defect features to some extent, which is conducive to improving the accuracy and P and R of model and partial average precision. However, FPS decreases to 86.69, indicating that, compared with Baseline, this module has certain limitation in aspect of model detection speed. Model_2 builds on Model_1 by integrating the DySample upsampling method, with Precision increasing significantly from 0.935 to 0.944 and FPS is slightly improved—reflecting that, compared with Baseline, DySample upsampling method plays a role in optimizing sample utilization and improving model precision. Model_3 is based on Model_1 and introduces only ATFL loss function, and P increases significantly from 0.932 to 0.957 and FPS increases from 91.07 to 95.44 – showing that, compared with Baseline, ATFL loss function plays a critical role in improving model precision, specific average precision and detection speed. However, R decreases compared with Baseline, indicating that, compared with Baseline, ATFL loss function has certain limitation in aspect of Recall.

As the number of integrated modules increases (e.g., Model_4, Model_5, Model_6), each variant exhibits varying changes across metrics but suffers from performance limitations. For example, the FPS of Model_4 is only 87.4, which is far lower than that of all single module models and other combination modules; the R of Model_5 is 0.872, which is an extremely bad performance; the R of Model_6 is 0.887, which still has a place for improvement. This indicates that the Recall for PCB defect detection still has room for further enhancement.

When the above three modules (C2PSA_iRMB + DySample + ATFL function) are combined into the model, the following advantages are presented. In the case of Recall, Model_7 can achieve 0.895, which is the highest among all models – namely, this model can identify more types of PCB defects, greatly improving the ability to detect defects on PCBs and eliminating missed detection. In the case of the mAP_{50}, Model_7 can achieve 0.93, which is also the highest among all models – namely, this model can achieve the optimal average accuracy of detecting all defects on PCBs when the matching threshold is moderate, and the defect identification is relatively accurate. In addition, the precision of Model_7 is 0.941, which is slightly lower than the 0.967 of Model_6. However, combined with its highest Recall and mAP_{50}, Model_7 achieves a better balance between precision and recall: it can accurately identify PCB defects while recalling defects to the greatest extent, effectively avoiding false detections and missed detections. Considering the FPS indicator comprehensively, although Model_7 is not the highest, its outstanding performance in other core indicators is sufficient to prove that the fusion of the three modules achieves complementary advantages. The roles of the C2PSA_iRMB (enhancing defect feature extraction and distinction capabilities), DySample (optimizing sample utilization), and ATFL (enhancing the comprehensive performance of the overall model) are fully exerted – enabling YO-

Figure 10

Comparison of detection results between YOLOv11 and YOLO-iDAT.



LO-iDAT to achieve the optimal overall performance in key indicators of PCB defect detection and fully verifying the effectiveness of multi-module fusion and the excellent performance of YOLO-iDAT in PCB defect detection tasks.

Figure 10 shows a comparison of the detection results of YOLO-iDAT and YOLOv11n for six types of PCB defects. The first row marks the positions of defects, with each image containing exactly one type of defect; the second row shows the detection results of the YOLOv11 model; and the third row shows the detection results of the YOLO-iDAT model. Each column corresponds to defects of open circuit, mouse bite, short, spurious copper, spur, and missing hole, respectively.

Observing the detection results in Columns 1, 2, and 4 reveals that the YOLOv11 model exhibits missed detections in these three scenarios. Comparing the detection results in Columns 3 and 4 further reveals that the YOLOv11 model also has false detection issues. In the detection results of Column 6, the confidence level output by the YOLOv11 model is generally lower than that of the YOLO-iDAT model. Comprehensive comparison results indicate that the detection performance of the YOLO-iDAT model is superior to that of the YOLOv11 model.

Table 7

Recall Rates of 6 Defect Types.

Defect Type	YOLOv11 Recall	YOLO-iDAT Recall	Improvement
Missing hole	1	0.975	-1.50%
Mouse bite	0.8	0.811	1.00%
Open circuit	0.777	0.88	10.30%
Short	0.956	0.969	1.30%
Spur	0.814	0.816	0.20%
Spurious copper	0.902	0.918	1.60%

As shown in Table 7, the recall rate of the YOLO-iDAT model for the "Open circuit" defect increased by 10.30% compared with the baseline. The recall rates for the defects "Mouse bite", "Short", "Spur" and "Spurious copper" increased by 1%, 1.3%, 0.2% and 1.6% respectively, and the overall recall rate for difficult-to-detect defects increased by an average of approximately 2.5%. Only the recall rate of the "Missing hole" defect fluctuated slightly by 0.015. This is because the recall rate of the baseline model on this defect has reached 100%. In high baseline scenarios, the difference between the training random seed and the sample distribution is easily magnified, which is a normal statistical fluctuation and not a degradation of model performance.

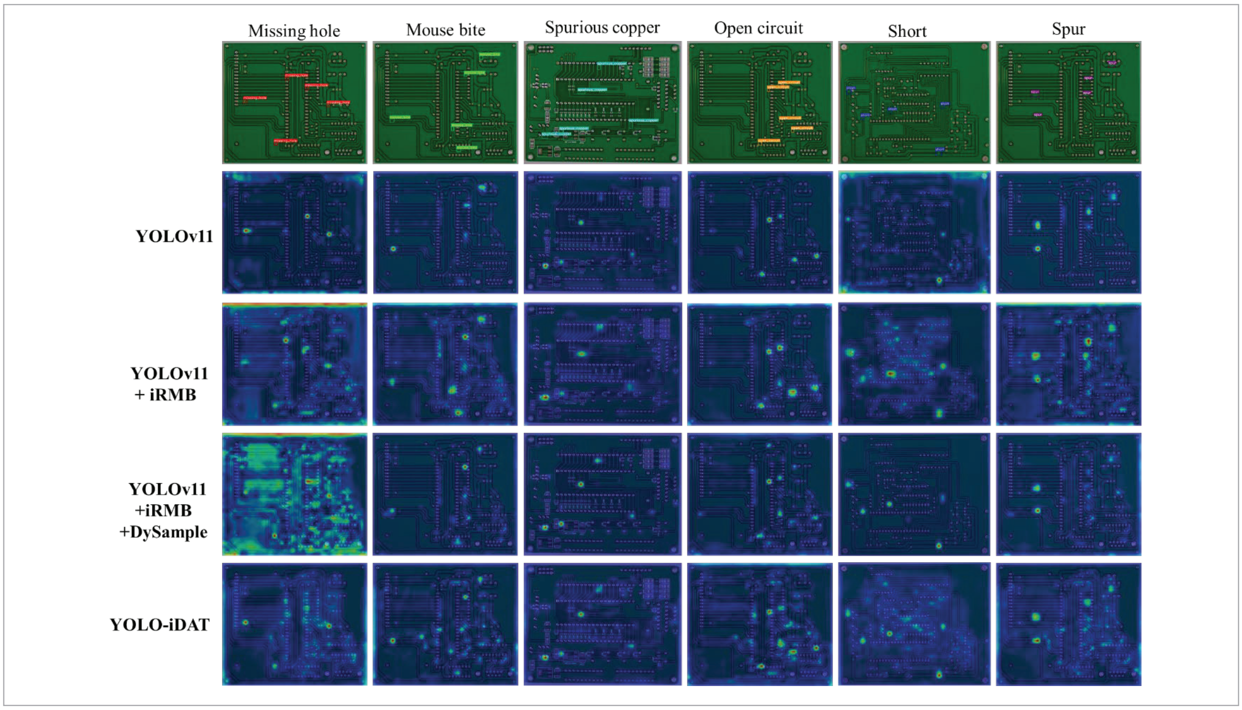
It can be seen from this that the YOLO-iDAT model has achieved a significant improvement in the recall rate of most difficult-to-detect defects. Only in high-baseline defect scenarios does statistical fluctuations occur that have no actual performance impact. The overall detection capability is superior to that of the baseline model.

To clarify the feature transfer relationship and complementary mechanism among the three modules (C2PSA_iRMB, DySample, and ATFL), the Grad-CAM method was employed to generate feature response heatmaps of PCB defect regions (as shown in Figure 11). A comparative analysis was conducted on the feature perception capabilities of the original YOLOv11, models improved with a single module, models combined with two modules, and the final YOLO-iDAT model.

The comparison results indicate that the original YOLOv11 baseline model exhibits scattered response regions and low intensity for tiny defects (such as spurs and spurious copper), making it difficult to effectively distinguish defect contours. This reflects the model's insufficient ability to capture features of low-contrast and small-scale defects. After introducing only the C2PSA_iRMB module, the responses in defect regions begin to concentrate. The heatmap boundaries of defects with obvious features (e.g., missing holes and shorts) tend to be clear, demonstrating that this module enhances the extraction of local detailed features. However, for hard samples with messy edges like mouse bites, the responses remain relatively scattered, and the improvement in feature attention is limited.

Figure11

Grad-CAM-Based Heatmap Visualization.



On this basis, after adding the DySample dynamic upsampling mechanism, the response intensity of all defect types is further enhanced, and the overlap between the heatmap range of tiny defects and the actual defect contours is significantly improved. This is because DySample can adaptively adjust sampling positions based on the local features enhanced by C2PSA_iRMB, avoiding feature distortion caused by traditional upsampling and realizing the effective retention and transfer of enhanced features.

When all three modules are integrated to form the YOLO-iDAT model, the heatmaps of all defect types show the characteristics of highly concentrated response regions and significantly enhanced intensity. In particular, the feature attention to hard samples such as mouse bites and spurs is greatly improved. This indicates that the ATFL loss function dynamically focuses on the learning weight of hard-to-classify samples, making up for the insufficient ability of C2PSA_iRMB to capture hard samples and forming a good complementary effect with the preceding modules.

In summary, C2PSA_iRMB is responsible for the enhanced extraction of underlying local detailed features, DySample realizes the efficient transfer and retention of enhanced features, and ATFL strengthens the learning of hard samples that are difficult to be captured by the preceding modules. The three modules form a collaborative chain from feature enhancement to feature transfer, and then to hard sample focusing, ultimately improving the model's detection performance for tiny and low-contrast PCB defects.

4.4. Discussion

Although this study focuses on PCB defect detection, the core module of YOLO-iDAT is designed to address fundamental challenges in industrial vision. Among them, the enhanced local detail and global context fusion capability of the C2PSA_iRMB module is crucial for any refined detection task; The problem of small target feature blurring alleviated by the sampler on DySample is not exclusive to PCB defect detection. The problem of foreground and background imbalance handled by the ATFL loss function is a common pain point in the field of defect detection. Therefore, we believe that YOLO-iDAT has strong cross-domain adaptability potential and can be applied to scenarios with similar requirements such as semiconductor wafer inspection, textile defect identification or surface quality assessment.

Meanwhile, the construction and augmentation of PCB defect detection datasets remain a research hotspot in the field of industrial vision. Core challenges stem from two aspects: First, defect samples are inherently scarce in industrial settings, with high annotation costs. Existing public datasets (such as PKU-Market-PCB) have limited sample sizes and differ from the complex scenarios encountered in real production environments, making it difficult to support improvements in model generalization capabilities. Second, the trend toward higher density and miniaturization in PCB design demands finer-grained feature characterization in datasets. Traditional datasets struggle to capture low-contrast, irregularly shaped micro-defects. Current research primarily focuses on two directions: First, defect image generation using techniques like diffusion models. For instance, the Strength-Controlled Defect Synthesis (SDAS) approach can selectively augment scarce defect samples to enhance data diversity. Second, developing fine-grained data augmentation methods tailored to defect features, such as super-resolution generation and adaptive parameter tuning to enhance the expression of minute defect characteristics. These explorations aim to overcome the limitations of small sample sizes. However, achieving precise alignment between synthetic samples and real-world scenarios while minimizing feature distortion during augmentation remains a critical challenge requiring ongoing optimization.

5. Conclusion

Addressing the challenges of insufficient detection accuracy and poor real-time performance for tiny, low-contrast defects (e.g., missing holes, mouse bites, open circuits) in the context of PCBs developing towards miniaturization, lightweight, and high density, this study optimizes YOLOv11 as the base model and proposes the improved YOLO-iDAT model from three core dimensions: feature extraction, upsampling mechanism, and loss function. The goal is to provide an efficient technical solution for industrial-grade PCB defect detection. Its core contributions and value are as follows:

In terms of feature extraction and upsampling optimization, the designed C2PSA_iRMB module integrates the iRMB structure, SE channel attention mechanism, and residual connections – compensating for the weak local detail modeling in the original model, enabling efficient fusion of global dependencies and local exture features, and significantly enhancing the ability to capture tiny edge defects and fine texture information of PCBs. This provides a good grain feature basis for the latter accurate detection. This paper proposes DySample dynamic sampler to overcome the limitation of feature blurring in tiny defect caused by static upsampling method. By dynamically changing sampling positions and sampling strategies, DySample accurately preserves the grain feature of tiny defect and improves the inference efficiency of model without significantly increasing the cost" – improves the model's ability to recognize low-contrast defects.

In terms of loss function design, compared with other functions, the ATFL function can solve the problem of model learning bias caused by sample imbalance in PCB defect detection. From the perspective of easy-to-classify and hard-to-classify sample division based on threshold, and parameter adjustment, it enhances the learning degree of tiny, low-contrast, and hard-to-classify defects, reduces the missed detection rate of key defects, and improves the robustness of model detection. It ensures that the model has stable detection performance in the following complex scene: backgrounds occupy a large proportion and defects occupy a small proportion.

In terms of the overall performance, by optimizing multiple parts, YOLO-iDAT can achieve a relatively good balance between detection accuracy and real-time performance. Based on the comparison with other models, this model has obvious advantages in the basic indicators of Recall and mAP50, and can be applied to meet the requirements of automated detection of PCB defects in industrial applications. This study offers a novel technical approach to address the challenges of low accuracy, poor efficiency, and limited adaptability to complex defects in PCB defect detection. Subsequent work will validate and optimize this framework using relevant datasets, further verify its cross-domain robustness, establish its position as a universal solution for microscopic defect detection, and promote the broader and more stable application of improved models in actual PCB production lines. This will provide more effective technical support for quality control in electronic manufacturing.

Conflicts of Interest

The author declares that there is no conflict of interest.

References

1. Chen, H., Song, J., Han, C., Xia, J., Yokoya, N. ChangeMamba: Remote Sensing Change Detection with Spatiotemporal State Space Model. IEEE Transactions on Geoscience and Remote Sensing, 2024, 62, 1-20, Article No. 4409720. https://doi.org/10.1109/TGRS.2024.3417253
2. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. Generative Adversarial Nets. In, Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS), 2014, 2672-2680. DOI: 10.1145/3422622.3422626.
3. Guo, Q., Chen, Y., Zhu, Y., Chen, D. CM-UNetv2: An Enhanced Semantic Segmentation Model for Precise PCB Defect Detection and Boundary Restoration. Sensors, 2025, 25(16), 4919. https://doi.org/10.3390/s25164919
4. Hu, Z., Zhang, Z., Liu, S., Zhao, D., Zheng, L., Tang, L. ASTKD-PCB-LDD: High-Performance PCB Defect Detection Model with Align Soft-Target Knowledge Distillation and Lightweight Network Design. Journal of Supercomputing, 2025, 81(4). https://doi.org/10.1007/s11227-025-07045-9
5. Huang, J., Zhao, F., Chen, L. Defect Detection Network in PCB Circuit Devices Based on GAN Enhanced YOLOv11. Applied and Computational Engineering, 2025, 133, 129-135. https://doi.org/10.54254/2755-2721/2025.20610
6. Kim, J., Ko, J., Choi, H., Kim, H. Printed Circuit Board Defect Detection Using Deep Learning via a Skip-Connected Convolutional Autoencoder. Sensors, 2021, 21(15), 4968. https://doi.org/10.3390/s21154968
7. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE, 1998, 86(11), 2278-2324. https://doi.org/10.1109/5.726791
8. Li, D., Bai, F., Li, S., Liu, B., He, L. EL-PCBNet: An Efficient and Lightweight Network for PCB Defect Detection. Measurement, 2025, 253, 117719. https://doi.org/10.1016/j.measurement.2025.117719

9. Ling, Q., Isa, N. A. M. Printed Circuit Board Defect Detection Methods Based on Image Processing, Machine Learning and Deep Learning: A Survey. IEEE Access, 2023, 11, 15921-15944. https://doi.org/10.1109/ACCESS.2023.3245093
10. Liu, F., Zheng, Q., Tian, X., Shu, F., Jiang, W., Wang, M., Elhanashi, A., Saponara, S. Rethinking the Multi-Scale Feature Hierarchy in Object Detection Transformer (DETR). Applied Soft Computing, 2025, 175, 113081. https://doi.org/10.1016/j.asoc.2025.113081
11. Liu, W., Lu, H., Fu, H., Cao, Z. Learning to Upsample by Learning to Sample. In, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023, 6004-6014. https://doi.org/10.1109/ICCV51070.2023.00554
12. Liu, X. An Adaptive Defect-Aware Attention Network for Accurate PCB-Defect Detection. IEEE Transactions on Instrumentation and Measurement, 2024, 73, 1-11, Article No. 5040811. https://doi.org/10.1109/TIM.2024.3488158
13. Lu, H., Liu, W., Fu, H., Cao, Z. FADE: Fusing the Assets of Decoder and Encoder for Task-Agnostic Upsampling. Lecture Notes in Computer Science, 2022. https://doi.org/10.1007/978-3-031-19812-0_14
14. Luo, Z. D., Lei, L., Li, H. X. Rotation-Angle-Based Principal Feature Extraction and Optimization for PCB Defect Detection Under Uncertainties. IEEE Transactions on Industrial Informatics, 2025, 21(1), 932-939. https://doi.org/10.1109/TII.2024.3470849
15. Moganti, M., Ercal, F. Automatic PCB Inspection Systems. IEEE Potentials, 1995, 14(3), 6-10. https://doi.org/10.1109/45.464686
16. Open Lab on Human Robot Interaction, Peking University. Dataset Resources. Available at, https://robotics.pkusz.edu.cn/resources/dataset/.
17. Raman, R., Jiang, W., Bakshi, S. PODE: Panoptic-Aware Object-Wise Depth Estimation for Occlusion Prediction. IEEE Transactions on Consumer Electronics, 2025. https://doi.org/10.1109/TCE.2025.3610083
18. Wang, J., Chen, K., Xu, R., Liu, Z., Loy, C. C., Lin, D. CARAFE: Content-Aware ReAssembly of FEatures. In, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, 3007-3016. https://doi.org/10.1109/ICCV.2019.00310
19. Wang, W. C., Chen, S. L., Chen, L. B., Chang, W. J. A Machine Vision Based Automatic Optical Inspection System for Measuring Drilling Quality of Printed Circuit Boards. IEEE Access, 2017, 5, 10817-10833. https://doi.org/10.1109/ACCESS.2016.2631658
20. Xiao, G., Hou, S., Zhou, H. PCB Defect Detection Algorithm Based on CDI-YOLO. Scientific Reports, 2024, 14, 7351. https://doi.org/10.1038/s41598-024-57491-3
21. Yang, B., Wang, X., Xing, Y., Cheng, C., Jiang, W., Feng, Q. Modality Fusion Vision Transformer for Hyperspectral and LiDAR Data Collaborative Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2024, 17, 17052-17065. https://doi.org/10.1109/JSTARS.2024.3415729
22. Yang, B., Zhang, X., Zhang, J., Luo, J., Zhou, M., Pi, Y. EFLNet: Enhancing Feature Learning Network for Infrared Small Target Detection. IEEE Transactions on Geoscience and Remote Sensing, 2024, 62, 1-11, Article No. 5906511. https://doi.org/10.1109/TGRS.2024.3365677
23. Yu, X., He, Y. PCB Defect Detection Based on GAN Data Generation with Self-Attentive Mechanism. In, Proceedings of the International Conference on Frontiers of Electronics, Information and Computation Technologies (ICFEICT), 2022, 55-60. https://doi.org/10.1109/ICFEICT57213.2022.00018
24. Zaremba, W., Sutskever, I., Vinyals, O. Recurrent Neural Network Regularization. arXiv Preprint arXiv:1409.2329, 2014. DOI: 10.48550/arXiv.1409.2329.
25. Zhang, J., Li, X., Li, J., Liu, L., Xue, Z., Zhang, B., Jiang, Z., Huang, T., Wang, Y., Wang, C. Rethinking Mobile Block for Efficient Attention-Based Models. In, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023, 1389-1400. https://doi.org/10.1109/ICCV51070.2023.00133
26. Zhao, S., Chen, H., Zhang, X., Xiao, P., Bai, L., Ouyang, W. RS-Mamba for Large Remote Sensing Image Dense Prediction. IEEE Transactions on Geoscience and Remote Sensing, 2024, 62, 1-14, Article No. 5633314. https://doi.org/10.1109/TGRS.2024.3425540
27. Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Cheng, J. DETRs Beat YOLOs on Real-Time Object Detection. In, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, 16965-16974. https://doi.org/10.1109/CVPR52733.2024.01605
28. Zhou, G., Yu, L., Su, Y., Xu, B., Zhou, G. Lightweight PCB Defect Detection Algorithm Based on MSD-YOLO. Cluster Computing, 2024, 27(3), 3559-3573. https://doi.org/10.1007/s10586-023-04156-x
29. Zhou, Y., Yuan, M., Zhang, J., Ding, G., Qin, S. Review of Vision-Based Defect Detection Research and Its Perspectives for Printed Circuit Board. Journal of Manufacturing Systems, 2023, 70, 557-578. https://doi.org/10.1016/j.jmsy.2023.08.019