Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

SHNet: A Lightweight Network Focusing on Shallow High-Resolution Features for PCB Defect Detection

YU LIU[1], CHENGJIE FANG[2], and LIKUN LU[1]

[1]School of Information Engineering, Beijing Institute of Graphic Communication, Beijing 102600, China (e-mail: friedabigc@gmail.com; lklu@bigc.edu.cn)
[2]Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences, Hangzhou 310024, China (e-mail: fangchengjie24@mails.ucas.ac.cn)

Corresponding author: Likun Lu (e-mail: lklu@bigc.edu.cn).

This work was supported in part by the Research Project of Beijing Institute of Graphic Communication under Grant E6202405; in part by the Disciplinary Construction and Postgraduate Education Project of Beijing Institute of Graphic Communication under Grants 21090525014 and 21090325003; in part by the Doctoral Degree Programs Cultivation Project of the First-level Discipline of Information and Communication Engineering of Beijing Institute of Graphic Communication under Grant 21090525004; and in part by the Research Platform Construction Project of Beijing Institute of Graphic Communication under Grant KYCPT202509.

Yu Liu and Chengjie Fang are co-first authors.

ABSTRACT Automatic defect detection for printed circuit boards (PCBs) is crucial for manufacturing quality and reliability, yet remains challenging due to tiny defect sizes, low visual contrast, and deployment constraints. We propose SHNet, a lightweight YOLOv8s-based detector that emphasizes shallow high-resolution representations. Instead of relying on deep low-resolution features, SHNet removes the low-resolution detection head and uses the 16× feature only as semantic guidance in the neck, reducing complexity while preserving fine structural cues. To enhance detection performance, we introduce three modules: the Multi-kernel Adaptive Module (MKAM) for scale-aware representation, the Spatial-Semantic Alignment Module (SSAM) for cross-layer feature alignment, and the Wavelet Enhancement Module (WEM) for strengthening weak defect responses in the frequency domain. Experiments show that SHNet improves mAP@0.5:0.95 by 9.1% and 20.8% over YOLOv8s on two datasets, respectively, while achieving a model size of only 2.3 MB with 87% fewer parameters and 56% fewer FLOPs. Code is available at https://github.com/Frieda73/SHNet.

INDEX TERMS Printed circuit boards, defect detection, lightweight framework, semantic alignment, Wavelet enhancement

I. INTRODUCTION

Printed circuit boards (PCBs) are fundamental components in modern electronic devices [1]. However, defects such as burrs and missing holes often occur in the manufacturing process [2]. Thus, it has significance to detect surface defects accurately and efficiently [3]. Actually, this task presents a set of challenges in industrial applications. Different from general object detection, PCB defects have extremely high class imbalance and extremely small size. Most defects only account for 0.005% to 0.07% of the entire image area. In addition, the visual similarity between defects and complex backgrounds make it hard to distinguish between categories [4].

Traditional methods use techniques such as template matching, filter enhancement and image difference, and rely on manual extraction of features [5]. While these methods may be effective in highly canonical environments, they have limited robustness in the face of complex backgrounds or microscopic defects. Moreover, despite their potential advantages for rapid detection, they lack the adaptability required for modern production lines. Machine learning methods, like Principal Component Analysis [6], random forests [7], and reinforcement learning [8], heavily rely on the performance of hand-designed classifiers. These models usually have high computational complexity and slow inference speed, which limits real-time applications.

In recent years, deep learning technology has developed fast, and Convolutional Neural Network (CNN) has performed well. For example, two-stage networks like R-CNN series [9–11], prioritize accuracy by generating candidate regions. Single-stage networks, like YOLO series [12–14], SSD [15] and EfficientDet [16], optimize for speed by directly predicting categories and bounding boxes. Recently,

Vision Transformers (ViTs) [17] have been gradually introduced into vision tasks because of their long-range dependency modeling ability and parallelization advantages. Efficient sequence modeling methods based on Mamba architecture [18], have also begun to attract attention.

Despite these remarkable advances, applying deep learning to PCB defect detection still faces many challenges. YOLO-based detectors have achieved great success in general object detection by progressively expanding the receptive field through stacked convolutional and downsampling layers. However, for PCB defect detection, repeated downsampling may prematurely discard high-resolution details, leading to missed detections or inaccurate localization of tiny defects. On the one hand, the morphology of defects is diverse. At the same time, the visual contrast between defects and complex backgrounds is usually very low. On the other hand, many high-precision models are too complex [19–22] with too many parameters and computational costs, which limits their applicability in real industrial environments.

To solve these problems, this paper proposes a lightweight network focused on shallow high-resolution features (SHNet). The core idea of SHNet is to get rid of the dependency on deep, low-resolution features. First, to cope with the large differences in defect scales, we design the Multi-kernel Adaptive Module (MKAM). The module extracts features from multiple branches with different receptive fields in parallel and fuses them through an adaptive gating mechanism, allowing the network to flexibly capture defects ranging from tiny burrs to large scratches. Second, we propose the Spatial-Semantic Alignment Module (SSAM), which combines local fine-grained texture recognition with global semantic modeling to ensure SHNet can understand the context of the circuit layout and suppress background noise. Furthermore, the Wavelet Enhancement Module (WEM) is deeply integrated in the feature fusion path. WEM uses wavelet transform to analyze and reconstruct features in the frequency domain, which significantly enhances weak defect signals that are easily confused with background textures.

- We propose SHNet, a lightweight PCB defect detector that emphasizes shallow high-resolution features and removes the $32\times$ detection head, making it more suitable for tiny and low-contrast defects.
- We design MKAM to compensate for the reduced deep receptive field by adaptively integrating point-wise, local, large-kernel, and asymmetric convolutional branches.
- We introduce SSAM and WEM to improve spatial-semantic alignment and enhance weak defect-related frequency components under complex PCB backgrounds.

The remaining part of this article will be organized in the following way. Section 2 reviews the work related to PCB defect detection and Section 3 details the proposed SHNet. Section 4 present experimental results and discussion, and finally in Section 5, conclusions and perspectives for future work are given.

II. RELATED WORK

A. SCALE-AWARE FEATURE LEARNING FOR OBJECT DETECTION

Defects on PCB board have a wide range of sizes, from large scratches covering multiple components to subtle short circuits only a few pixels wide, which puts high requirements on the scale adaptability of the model. Classical solutions for multi-scale problems are to employ Feature Pyramid Network (FPN) and its variants, such as PANet [23] and BiFPN [16]. However, when dealing with PCB defects exhibiting extreme scale variations, these approaches face a fundamental limitation: for pixel-level micro defects, their visual features are often severely degraded during the initial downsampling stages of the backbone. Although subsequent pyramid structures can fuse multi-scale features, they cannot recover spatial details lost during early resolution reduction. KAN-YOLO [24] integrates Kolmogorov–Arnold Network modules into YOLOv8 and improves the detection accuracy at a higher computational cost. MV-YOLO [25] uses a Mamba-based model with the BHFM module for multi-scale feature extraction, but remains ineffective for extremely small targets because of downsampling information loss. CoTNet [26] introduces Contextual Transformer (CoT) blocks, which use local contextual encoding of keys through 3×3 convolution to guide dynamic attention learning. This design enhances contextual representation for visual recognition tasks. However, CoTNet was originally proposed as a general-purpose visual recognition backbone rather than a PCB-specific defect detector; therefore, its relevance to PCB defect detection mainly lies in contextual feature modeling rather than explicit preservation of shallow high-resolution defect details.

B. TEXTURE CONFUSION IN COMPLEX LOW-CONTRAST BACKGROUNDS

In complex PCB images, defect areas are often highly similar to dense circuit traces, pads and other background in texture and gray level, and the contrast is very low, resulting in very weak feature signals, which are easily drowned by strong background noise, making the detection model easy to be confused, resulting in missed detections and false positives. To enhance discriminative capability in such scenarios, the deformable Transformer-FPN method [27] combines deformable Transformers with an FPN to better localize irregular defects while increasing computational cost. The coordinate-based refinement module [28] improves weak defect recognition but has limited generalization. The local–global context modeling method [29] enhances PCB defect detection but remains less effective on complex spatial layouts. The dual-layer template matching with ECA-enhanced YOLOv7 method [30] improves detection accuracy but remains sensitive to low-contrast textures. The unsupervised reconstruction-based method [31] detects defects as deviations from normal texture patterns.

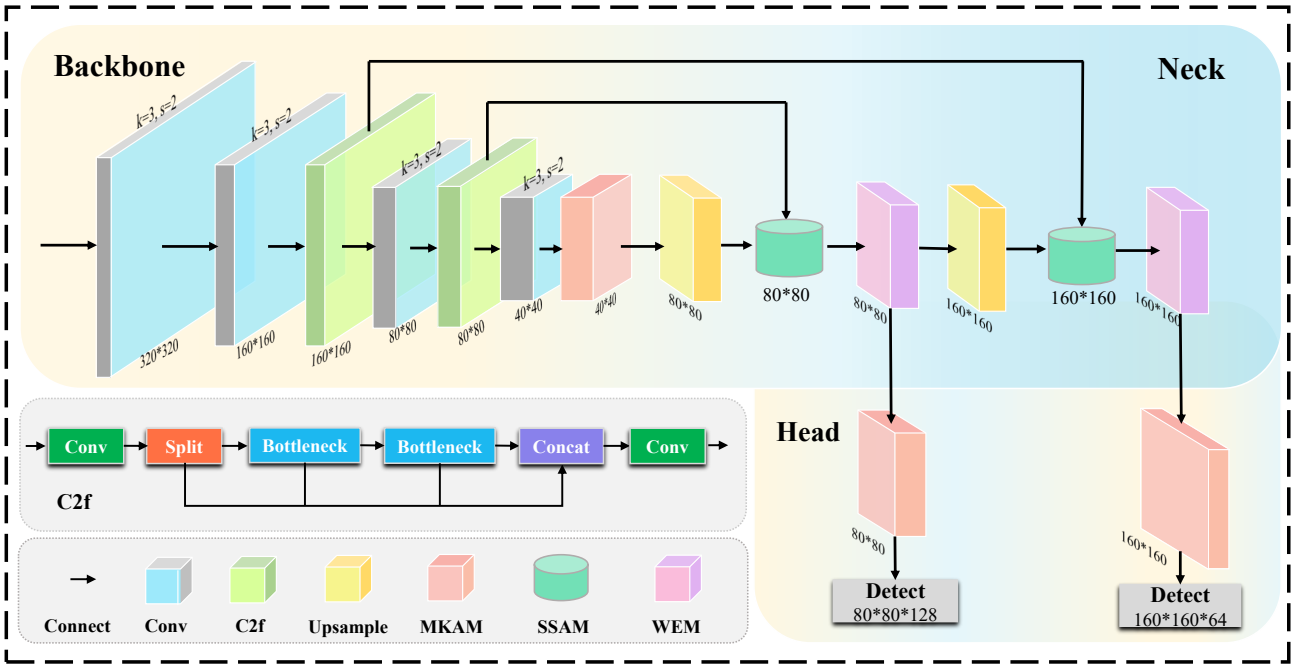


FIGURE 1. Overall architecture of SHNet. MKAM, SSAM, and WEM denote the Multi-Kernel Adaptive Module, Spatial-Semantic Alignment Module, and Wavelet Enhancement Module, respectively. Conv denotes a convolutional layer, C2f denotes the standard C2f block used in YOLOv8, and Detect denotes the decoupled detection head. Feature-map dimensions are expressed as $H \times W \times C$, where H, W, and C denote height, width, and channel number, respectively.

C. LIGHTWEIGHT NETWORK

Real-time is very important in industrial inspection. Therefore, it is an urgent need to develop lightweight models to complete the deployment of edge computing platforms. One mainstream strategy is to optimize the network architecture by reducing the number of parameters and computational cost while improving inference speed through efficient convolutional designs or lightweight modules. GC-Neck [32] combines GSConv and CAA modules to get a favorable trade-off between lightweight design. However, the depthwise separable convolution (DSC) component in GSConv decouples spatial and channel operations, potentially losing cross-channel structural information despite reducing computation. The HS-LSKA and global-EMA method [33] balances accuracy and speed under small-object, complex-background scenarios. While HS-LSKA improves small-object sensitivity, it may misclassify background textures; global-EMA mitigates this but can increase false positives. MT-YOLO [34] replaces the backbone C3 with PConv for faster FPS and adds DGConv in the neck for irregular defect features. Ablation studies show slightly lower accuracy than YOLOv5s, traded for speed and smaller size.

III. METHOD

To address the challenges of PCB defect detection, including large scale variation, low contrast between defects and backgrounds, and the need for efficient deployment, we propose SHNet, a lightweight detector that emphasizes shallow high-resolution features. As shown in Fig. 1, SHNet consists of three components: a backbone for hierarchical feature extraction, a neck for feature enhancement and multi-scale fusion, and a standard decoupled detection head.

The backbone extracts multi-level semantic and texture features using stacked convolutional layers and C2f blocks. Different from conventional detectors that continuously downsample features to obtain deeper semantics, SHNet removes the P5 layer with 32× downsampling and retains features only up to the P4 level with 16× downsampling. This design is motivated by the small spatial extent of PCB defects. After excessive downsampling, tiny defects may be compressed into very few feature cells or confused with complex circuit backgrounds. Therefore, SHNet preserves the 4× and 8× high-resolution branches and uses the 16× feature as the deepest semantic representation, balancing fine structural details and contextual information.

The neck further enhances and fuses features from different levels. It follows a bidirectional pyramid structure and embeds three key modules: the Multi-kernel Adaptive Module (MKAM), the Spatial-Semantic Alignment Module (SSAM), and the Wavelet Enhancement Module (WEM). In the top-down path, MKAM enlarges the receptive field of deep features and strengthens contextual representation. SSAM then aligns shallow and deep features through global-local collaborative attention, improving both spatial consistency and semantic complementarity. WEM introduces wavelet-domain enhancement to suppress background noise and strengthen

defect responses while preserving low-frequency structural information. In the bottom-up path, features are progressively downsampled and fused to reinforce semantic consistency across scales. As a result, the neck produces multi-scale features with rich texture details, clear defect responses, and balanced semantic representation.

Finally, the decoupled detection head separates classification and localization tasks, enabling SHNet to detect both tiny defects and large-area anomalies on PCB surfaces.

A. MULTI-KERNEL ADAPTIVE MODULE

PCB defects vary in size and shape, from pinholes to scratches, which requires a high feature extraction ability of the model. The traditional convolution module usually uses a fixed receptive field, which is difficult to flexibly adapt to this scale diversity. To solve this problem, the multi-kernel adaptive module (MKAM) is designed, as shown in Fig. 2.

Formally, assume that the input feature map is $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, MKAM first processes the input through a set of parallel convolutional transformations. We define the operations of these four branches as sets of transformations $\Phi = \{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \mathcal{F}_4\}$, corresponding to 1×1 Conv for preserving fine-grained information, 3×3 depthwise separable convolution (DWConv) for extracting local textures, 7×7 DWConv for expanding receptive fields, and 1×7 and 7×1 asymmetric convolutional structures to enhance linear defect perception. These multi-scale features are then aggregated to generate intermediate features $\mathbf{U}$:

$$\mathbf{U} = \text{Concat}\left(\mathcal{F}_1(\mathbf{X}), \mathcal{F}_2(\mathbf{X}), \mathcal{F}_3(\mathbf{X}), \mathcal{F}_4(\mathbf{X})\right). \quad (1)$$

In order to achieve dynamic selection of different features, MKAM introduces a dynamic fusion mechanism. This mechanism aims to adaptively recalibrate the aggregated features $\mathbf{U}$, according to the importance of the channels. First, channel statistics are extracted by global average pooling. Subsequently, a multi-layer perceptron containing dimensionality reduction and dimensionality boosting operations is used to capture the dependencies between channels, and then, the Sigmoid is used to generate the attention weight vector $\boldsymbol{\alpha}$:

$$\boldsymbol{\alpha} = \delta\left(W_2 \cdot \sigma\left(W_1 \cdot \text{GAP}(\mathbf{U})\right)\right), \quad (2)$$

where W_1 and W_2 represent 1×1 convolutional weights, and σ and δ represent SiLU and Sigmoid.

Finally, multi-scale features are weighted using the generated attention weights to enhance feature responses related to defects and suppress background noise.

B. SPATIAL-SEMANTIC ALIGNMENT MODULE

In the multi-level feature fusion, the shallow features focus on spatial details, and the deep features contain higher semantic information. There is inconsistency between the two at the scale and semantic level, and the direct addition or splicing is easy to introduce noise and lead to feature degradation. To this end, we propose the Spatial-Semantic Alignment Module (SSAM) to achieve effective cross-scale feature fusion, as shown in Fig. 3. First, a 1×1 convolution is used to unify the channel dimensions of multi-scale features, and the Global-Local Attention Module (GLAM) is introduced for parallel feature enhancement. Among them, the local branch uses depth-wise separable convolution and gradient prior to strengthen defect edge response, and the global branch models long-range semantic dependence through lightweight Transformer. Then, the enhanced features are aligned to the same resolution and fused, and the channel and spatial attention are further calibrated by the Convolutional Block Attention Module (CBAM) [35]. Channel attention adjusts the importance distribution of feature subspaces, while spatial attention emphasizes the response of potential defect areas to alleviate the semantic mismatch problem. All GLAM enhanced feature maps are bilinear interpolated to the same spatial resolution, concatenated in the channel dimension, and deep fusion of information is completed by a final 1×1 convolution.

C. WAVELET ENHANCEMENT MODULE

Many PCB defects exhibit weak high-frequency responses that are easily confused with dense traces, pads, and other background textures. Pure spatial-domain enhancement may therefore be insufficient to separate defect-related details from background noise. To address this issue, we design the Wavelet Enhancement Module (WEM), which enhances feature representations through wavelet decomposition, frequency-domain collaborative enhancement, and inverse reconstruction, as shown in Fig. 4.

Given an input feature map $F \in \mathbb{R}^{C\times H\times W}$, WEM first applies the two-dimensional discrete wavelet transform (DWT) to decompose it into a low-frequency approximation subband and three high-frequency detail subbands:

$$\{F_l, F_h\} = \text{DWT}(F), \quad F_h = \{F_{LH}, F_{HL}, F_{HH}\}. \quad (3)$$

where F_l denotes the low-frequency approximation subband, while F_{LH}, F_{HL}, and F_{HH} denote the three high-frequency detail subbands.

For the low-frequency component, channel attention is used to enhance global semantic and structural information:

$$\tilde{F}_l = \mathcal{A}_c(F_l) \odot F_l, \quad (4)$$

where $\mathcal{A}_c(\cdot)$ denotes the channel attention mapping function, and $\odot$ represents element-wise weighting.

Since defect-related textures are mainly reflected in high-frequency components, WEM further uses the enhanced low-frequency feature $\tilde{F}_l$ to guide high-frequency enhancement. Specifically, a spatial guidance map M_{guide} and a background gating map G_{bg} are generated as:

$$M_{\text{guide}} = \mathcal{G}_s(\tilde{F}_l), \quad G_{\text{bg}} = \mathcal{G}_b(\tilde{F}_l). \quad (5)$$

Here, $\mathcal{G}_s(\cdot)$ highlights high-frequency responses aligned with potential defect regions, while $\mathcal{G}_b(\cdot)$ suppresses irrelevant

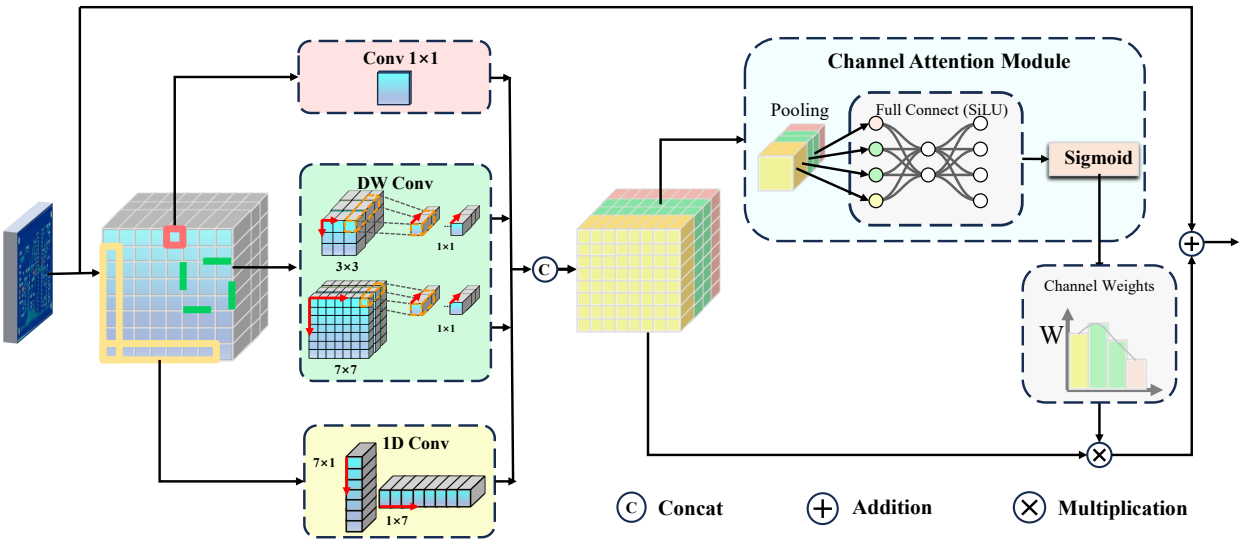


FIGURE 2. The detailed structure of the Multi-kernel Adaptive Module.

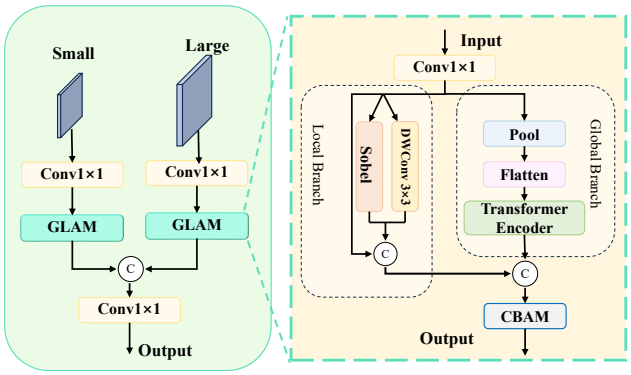


FIGURE 3. Illustration of the Spatial-Semantic Alignment Module and its core component, the Global-Local Attention Module.

background textures. The high-frequency features are then refined by guided modulation and channel attention:

$$F'_h = F_h \odot M_{\text{guide}} \odot G_{\text{bg}}, \tag{6}$$

$$\tilde{F}_h = \mathcal{A}_c(F'_h) \odot F'_h, \tag{7}$$

$$F_{\text{out}} = \text{IDWT}(\tilde{F}_l, \tilde{F}_h). \tag{8}$$

where $\text{IDWT}(\cdot)$ denotes the inverse discrete wavelet transform.

Through this process, WEM effectively integrates global structural cues and local defect details while maintaining the original spatial resolution, providing more discriminative features for the subsequent detection head.

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

Datasets.

TABLE 1. Experimental parameter settings.

Parameters	PKU-Market-PCB	DeepPCB
Epochs	200	200
Batch size	8	8
Learning rate	0.01	0.01
Momentum	0.937	0.937
Image size	640	640
Optimizer	SGD	SGD
Weight decay	0.0005	0.0005

We evaluate SHNet mainly on the PKU-Market-PCB dataset [36], which contains 693 high-resolution PCB images covering six common defect categories. Each image contains only one defect category but may include multiple instances of that defect. Representative examples of the six defect categories are shown in Fig. 5. As summarized in Table 2, the dataset is divided into 483 training images, 66 validation images, and 144 test images, containing 2,062, 279, and 612 defect instances, respectively. In total, PKU-Market-PCB contains 2,953 annotated defect instances.

We further evaluate the generalization ability of SHNet on the DeepPCB dataset [37], a publicly available black-and-white PCB defect detection dataset released by Shanghai Jiao Tong University. DeepPCB contains 1,500 defect images. Unlike PKU-Market-PCB, a single image in DeepPCB may contain multiple defect categories, with multiple defect instances co-occurring within the same image, as illustrated in Fig. 6. As reported in Table 3, the dataset consists of 1,050 training images, 150 validation images, and 300 test images, containing 7,009, 993, and 2,011 defect instances, respectively, for a total of 10,013 annotated defect instances.

For both datasets, the images are randomly divided into

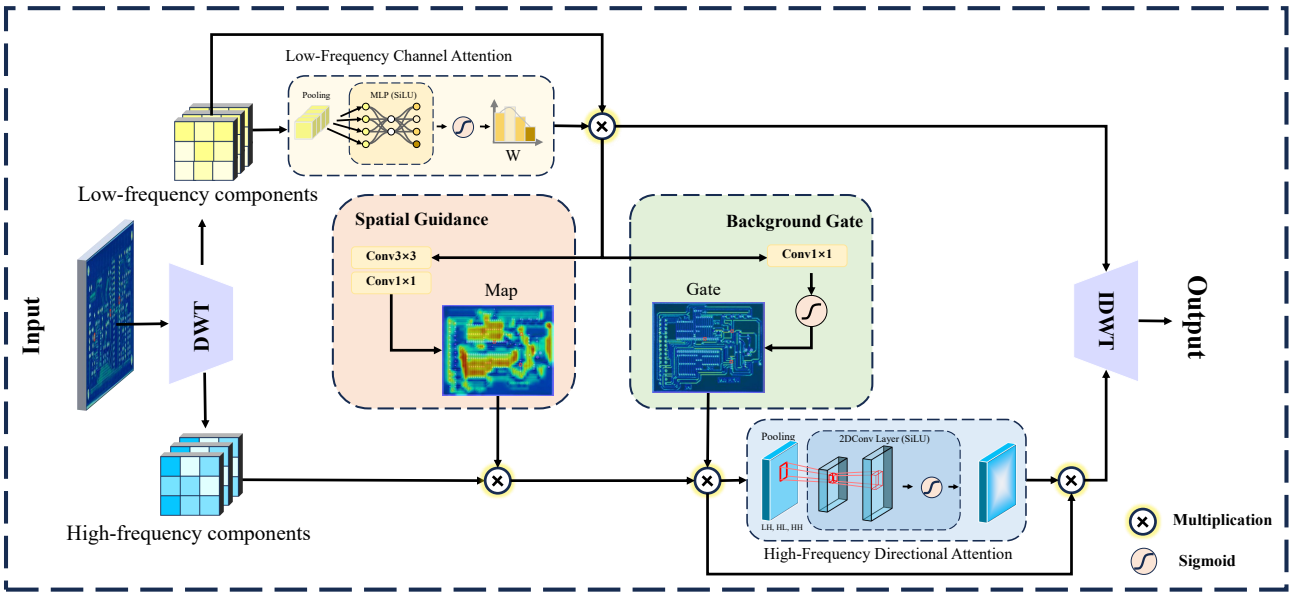


FIGURE 4. The workflow of the Wavelet Enhancement Module, including decomposition, synergistic enhancement, and reconstruction.

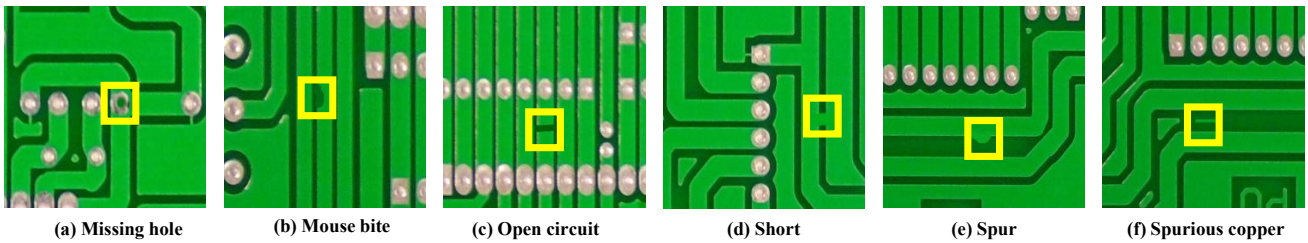


FIGURE 5. Examples of different defect maps in the PKU-Market-PCB dataset.

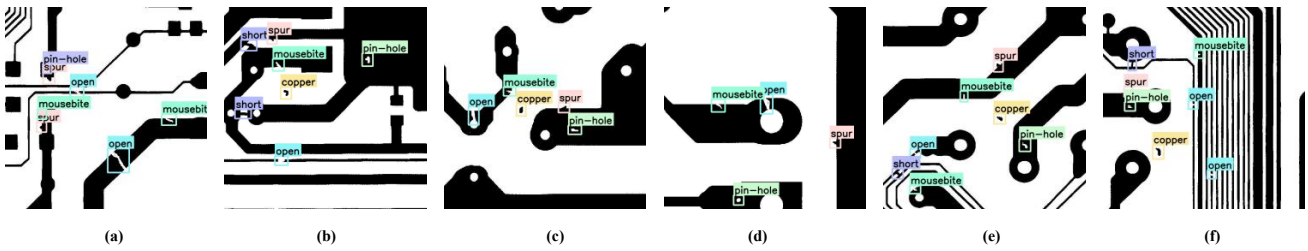


FIGURE 6. Representative defect images from the DeepPCB dataset. From (a) to (f), each image may contain multiple defect categories and multiple defect instances. The colored bounding boxes indicate annotated defect locations, and the accompanying text denotes the corresponding defect category.

training, validation, and test sets with a ratio of 7:1:2.

Experimental configuration. The experimental environment of this paper was based on Linux, and the hardware setup was composed of NVIDIA GeForce RTX 4090 GPU. The deep learning framework used was PyTorch 2.4.1, CUDA version 12.1, and Python 3.10.18. See Table 1 for the specific training strategies.

Evaluation metrics. For fair comparison with existing methods, we adopt the same evaluation metrics as in [22], including precision (P), recall (R), F1 score, mean average precision (mAP), parameter count, computational complexity, inference speed, and model size. The number of parameters (Params) is reported in millions (M), computational complexity is measured in giga floating-point operations (GFLOPs), inference speed is reported in frames per second (FPS), and model size is reported in megabytes (MB). The metric mAP@0.5 denotes mean average precision at an intersection-over-union (IoU) threshold of 0.5, whereas mAP@0.5:0.95 denotes the mean AP averaged over IoU thresholds from 0.50 to 0.95 with a step size of 0.05.

TABLE 2. Distribution of images and defect instances across different defect categories in the PKU-Market-PCB dataset.

Defect Type	Train		Validation		Test		Total	
	Images	Defects	Images	Defects	Images	Defects	Images	Defects
Missing Hole	80	345	11	41	24	111	115	497
Mouse Bite	80	344	11	48	24	100	115	492
Open Circuit	81	340	11	50	24	92	116	482
Short	81	348	11	46	24	97	116	491
Spur	80	328	11	53	24	107	115	488
Spurious Copper	81	357	11	41	24	105	116	503
Total	**483**	**2062**	**66**	**279**	**144**	**612**	**693**	**2953**

TABLE 3. Statistics of the DeepPCB dataset.

Split	Images	Defects
Train	1,050	7,009
Validation	150	993
Test	300	2,011
Total	**1,500**	**10,013**

B. RESULTS AND DISCUSSION

1) Main Experiments and Comparison

In order to evaluate the comprehensive performance of SHNet in industrial defect detection scenarios, experiments are carried out on two benchmark datasets. The comparison includes general object detection algorithms, such as YOLO series, Faster R-CNN, and RT-DETR, as well as lightweight detection methods, including Hyper-YOLO, MHAF-YOLO, and Mamba-YOLO. The quantitative results are summarized in Table 4.

On the PKU-Market-PCB dataset, SHNet obtains 0.9554 precision, 0.9115 recall, 0.9320 F1 score, 0.9398 mAP@0.5, and 0.5091 mAP@0.5:0.95. Among the compared methods, these values are the highest across the main accuracy metrics. Compared with YOLOv8s, SHNet improves mAP@0.5 from 0.9053 to 0.9398 and mAP@0.5:0.95 from 0.4666 to 0.5091. At the same time, the number of parameters is reduced from 7.60M to 0.97M, and the model size is reduced from 14.8 MB to 2.3 MB. This result suggests that the proposed design can improve detection accuracy without increasing the model scale.

Compared with other YOLO-series baselines, SHNet also shows a favorable balance between detection performance and model complexity. YOLOv11s obtains the second-best recall, F1 score, and mAP@0.5:0.95 on PKU-Market-PCB, but it requires 9.42M parameters and has a model size of 18.3 MB. YOLOv13s achieves relatively high precision, but its recall and mAP@0.5:0.95 are lower than those of SHNet. This indicates that, for PCB defect detection, a high precision value alone may not be sufficient, because missed detections are also important in practical inspection scenarios. Faster R-CNN and RT-DETR show lower recall and mAP@0.5:0.95 on this dataset, which may be related to the small size and low contrast of PCB defects.

Among the lightweight methods, Hyper-YOLO provides a competitive result on PKU-Market-PCB, with 0.9247 mAP@0.5, 0.4731 mAP@0.5:0.95, 3.62M parameters, and a model size of 7.3 MB. SHNet further improves mAP@0.5:0.95 to 0.5091 while using fewer parameters. MHAF-YOLO has the lowest GFLOPs among the compared models, but its precision, recall, F1 score, and mAP values are lower than those of SHNet. Mamba-YOLO also has moderate model complexity, but its accuracy and FPS are less favorable under this experimental setting. These comparisons indicate that reducing computational cost alone is not sufficient; for PCB defects, preserving fine spatial details is also important.

On the DeepPCB dataset, SHNet achieves 0.9779 precision, 0.9725 recall, 0.9752 F1 score, 0.9867 mAP@0.5, and 0.8096 mAP@0.5:0.95. SHNet is not the best method in every individual metric: YOLOv9s obtains a slightly higher precision, and YOLOv5s and YOLOv10s obtain marginally higher mAP@0.5 values. However, the difference in mAP@0.5 between SHNet and the best result is only 0.0011, whereas SHNet improves mAP@0.5:0.95 by 0.0283 over YOLOv5s. This distinction is important because mAP@0.5 is a relatively loose localization criterion and is nearly saturated on this dataset, while mAP@0.5:0.95 evaluates localization quality across stricter IoU thresholds. Therefore, the DeepPCB results should be interpreted as evidence of stronger high-threshold localization and efficiency, rather than uniform dominance across all metrics.

Compared with YOLOv8s on DeepPCB, SHNet improves mAP@0.5:0.95 from 0.6701 to 0.8096 and increases FPS from 152.92 to 240.11. The parameter count is also reduced from 7.60M to 0.97M. Compared with YOLOv5s, which performs strongly on this dataset, SHNet improves mAP@0.5:0.95 from 0.7813 to 0.8096 while using fewer parameters and achieving higher FPS. These results suggest that the proposed model is not only effective on PKU-Market-PCB, but also maintains reasonable generalization on the grayscale DeepPCB dataset.

The comparison with lightweight detectors on DeepPCB shows a similar trend. Hyper-YOLO achieves 0.7531 mAP@0.5:0.95 with a compact model size of 7.3 MB, while

TABLE 4. Performance comparison of different models on PKU-Market-PCB and DeepPCB. The best results within each dataset are shown in bold, and the second-best results are underlined.

Model	Precision	Recall	F1 Score	mAP@0.5	mAP@0.5:0.95	GFLOPs	Params(M)	FPS	Size(MB)
PKU-Market-PCB dataset									
YOLOv5s [13]	0.9268	0.8815	0.9032	0.9253	0.4691	18.7	7.82	108.31	15.2
YOLOv8s [14]	0.9075	0.8698	0.8876	0.9053	0.4666	21.6	7.60	126.03	14.8
YOLOv9s [38]	0.9254	0.8680	0.8949	0.8970	0.4684	22.1	6.20	106.93	12.7
YOLOv10s [39]	0.9177	0.8624	0.8870	0.9141	0.4847	21.4	7.22	110.44	15.8
YOLOv11s [40]	0.9363	0.8972	0.9157	0.9161	0.4884	21.3	9.42	72.10	18.3
YOLOv12s [41]	0.9062	0.8807	0.8929	0.9056	0.4660	19.3	9.08	97.74	17.8
YOLOv13s [42]	0.9400	0.8329	0.8818	0.8815	0.4177	20.0	9.00	112.44	17.8
Faster R-CNN [11]	0.7821	0.4693	0.5863	0.7822	0.3653	181.76	41.10	20.10	158.1
RT-DETR [43]	0.8883	0.8564	0.8717	0.8850	0.4115	103.5	31.99	92.14	63.1
Hyper-YOLO [44]	0.9410	0.8819	0.9092	0.9247	0.4731	9.5	3.62	142.15	7.3
MHAF-YOLO [45]	0.8750	0.7963	0.8297	0.8597	0.4391	**7.7**	2.28	148.53	30.6
Mamba-YOLO [21]	0.9294	0.8369	0.8784	0.8814	0.4383	12.3	5.67	60.13	11.1
Ours	**0.9554**	**0.9115**	**0.9320**	**0.9398**	**0.5091**	9.5	**0.97**	**168.00**	**2.3**
DeepPCB dataset									
YOLOv5s [13]	0.9689	0.9706	0.9696	**0.9878**	0.7813	18.7	7.82	197.63	15.2
YOLOv8s [14]	0.9464	0.9069	0.9254	0.9688	0.6701	21.6	7.60	152.92	14.8
YOLOv9s [38]	**0.9812**	0.9673	0.9739	0.9865	0.7523	22.1	6.20	135.60	12.7
YOLOv10s [39]	0.9646	0.9670	0.9658	0.9873	0.7643	21.4	7.22	172.66	15.8
YOLOv11s [40]	0.9748	0.9396	0.9567	0.9815	0.7326	21.3	9.42	191.63	18.3
YOLOv12s [41]	0.9743	0.9497	0.9614	0.9849	0.7328	19.3	9.08	130.48	17.8
YOLOv13s [42]	0.9669	0.9364	0.9510	0.9796	0.7469	20.0	9.00	72.79	17.8
Faster R-CNN [11]	0.9630	0.7311	0.8315	0.9633	0.6572	181.7	41.10	50.67	158.1
RT-DETR [43]	0.9570	0.9532	0.9545	0.9816	0.6825	103.5	31.99	92.14	63.1
Hyper-YOLO [44]	0.9502	0.9573	0.9525	0.9678	0.7531	9.5	3.62	177.53	7.3
MHAF-YOLO [45]	0.9319	0.9070	0.9190	0.9629	0.5406	**7.7**	2.28	222.15	30.6
Mamba-YOLO [21]	0.9531	0.8967	0.9224	0.8974	0.7236	12.3	5.67	60.13	11.1
Ours	0.9779	**0.9725**	**0.9752**	0.9867	**0.8096**	9.5	**0.97**	**240.11**	**2.3**

SHNet reaches 0.8096 mAP@0.5:0.95 with a smaller model size. MHAF-YOLO has relatively low GFLOPs and few parameters, but its mAP@0.5:0.95 is 0.5406, indicating that its localization performance is limited in this setting. Mamba-YOLO obtains 0.7236 mAP@0.5:0.95, but its FPS is lower than that of SHNet. These results suggest that SHNet achieves a relatively balanced trade-off among accuracy, parameter efficiency, and inference speed.

Overall, the results in Table 4 show that SHNet provides a favorable accuracy–efficiency trade-off under the present experimental protocol. On PKU-Market-PCB, the gains are reflected across the main accuracy metrics, suggesting that preserving shallow high-resolution features is beneficial for small and low-contrast defects. On DeepPCB, SHNet does not dominate every loose-threshold metric, but it achieves the highest mAP@0.5:0.95 while using the fewest parameters and the smallest model size. These results support the effectiveness of the proposed design for lightweight PCB defect detection, within the scope of the two evaluated datasets.

See Fig. 7 for a qualitative comparison of the detection results produced by different models on the PKU-Market-PCB dataset. In the ''True defect'' column, the red bounding boxes indicate the ground-truth defect locations. For the detection results of each model, each colored bounding box represents a predicted defect region; the text attached to the box denotes the predicted defect category, while the following numerical value denotes the corresponding detection confidence score, ranging from 0 to 1. A higher score indicates that the model is more confident in the predicted category and localization.

The visualization reveals noticeable differences among the compared methods, particularly for small defects located in complex PCB patterns. Successful detections are characterized by predicted boxes that closely overlap with the ground-truth regions and have correct category labels. Missed detections occur when one or more ground-truth defects are not covered by any predicted box, whereas false positives correspond to additional predicted boxes in regions where no defect is annotated. Category-confusion errors occur when a defect is localized but assigned an incorrect class label.

For example, in the short-circuit sample, YOLOv5s fails to detect all defect instances, while YOLOv9s introduces an incorrect prediction of spurious copper. Faster R-CNN also shows category confusion by classifying a short-circuit defect as an open-circuit defect, and Hyper-YOLO produces an erroneous spur prediction in the same sample. In the spur case, Hyper-YOLO, MHAF-YOLO, and Mamba-YOLO generate additional spurious-copper predictions that are absent from the ground truth, indicating susceptibility to false alarms under dense and repetitive PCB textures. In contrast, SHNet produces predictions that are more consistent with the ground-

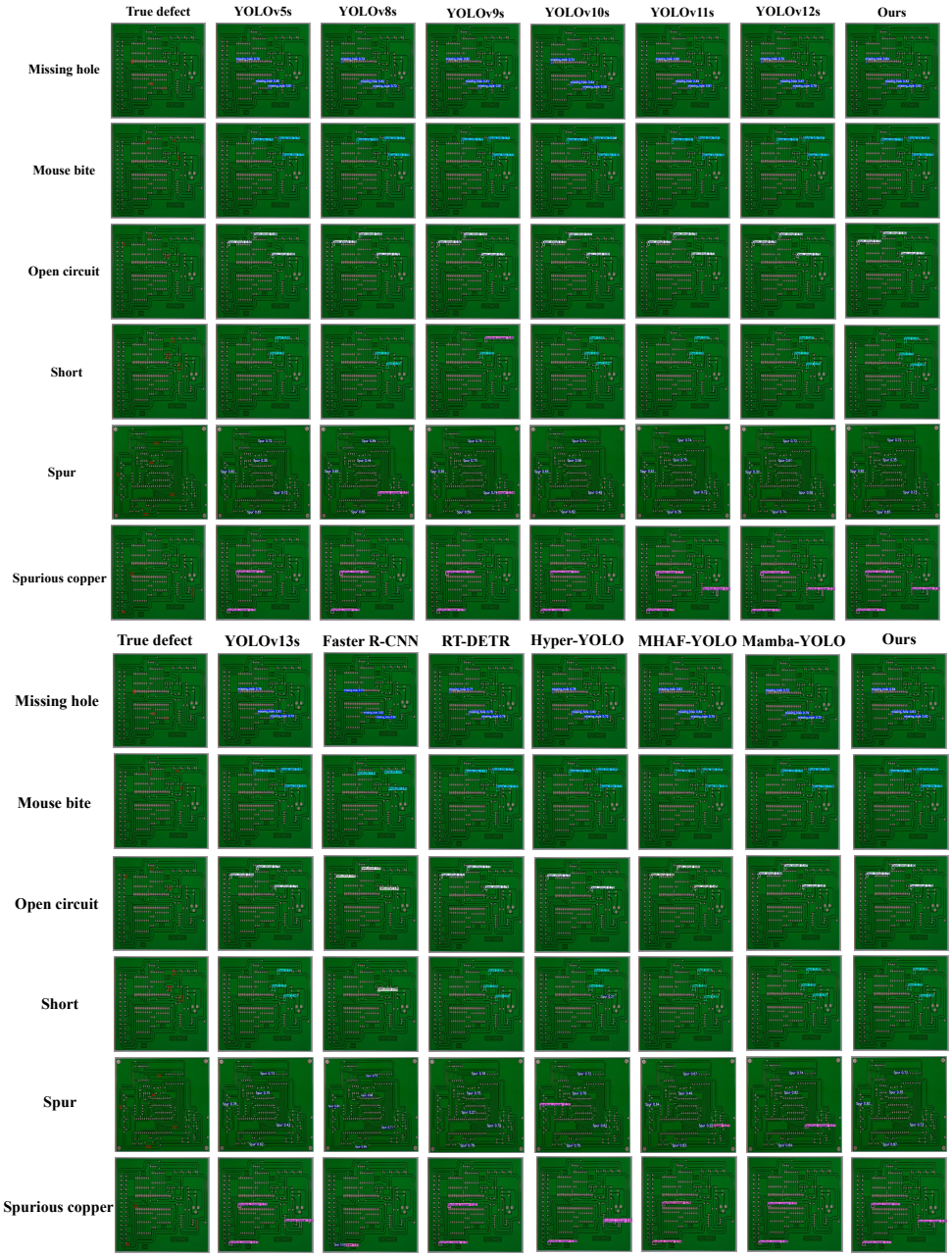


FIGURE 7. **Qualitative comparison of different detection models on the PKU-Market-PCB dataset. Red bounding boxes in the "Ground truth" column indicate the annotated defect locations. For each detection model, colored bounding boxes denote predicted defect regions, the accompanying text denotes the predicted defect category, and the numerical value following the category label represents the corresponding detection confidence score in the range [0, 1].**

truth annotations across the six defect categories, with fewer missed detections, false positives, and category-confusion errors. These qualitative results indicate that the proposed shallow high-resolution architecture preserves fine-grained defect cues more effectively and improves the localization and recognition of small and low-contrast PCB defects.

On PKU-Market-PCB, SHNet achieves the best overall detection performance with 0.9554 precision, 0.9115 recall, 0.9398 mAP@0.5, and 0.5091 mAP@0.5:0.95. Meanwhile, it contains only 0.97M parameters and reaches 168 FPS, showing a favorable balance between accuracy and efficiency. On DeepPCB, SHNet further obtains the best mAP@0.5:0.95 of 0.8096 and the highest inference speed of 240.11 FPS, which verifies its robustness and generalization ability across different PCB defect datasets. To further show the accuracy-efficiency trade-off, Fig. 8 visualizes mAP@0.5:0.95, parameter size, and GFLOPs on DeepPCB. SHNet lies in the high accuracy and low parameter region, indicating a strong Pareto advantage for lightweight industrial deploy- ment.

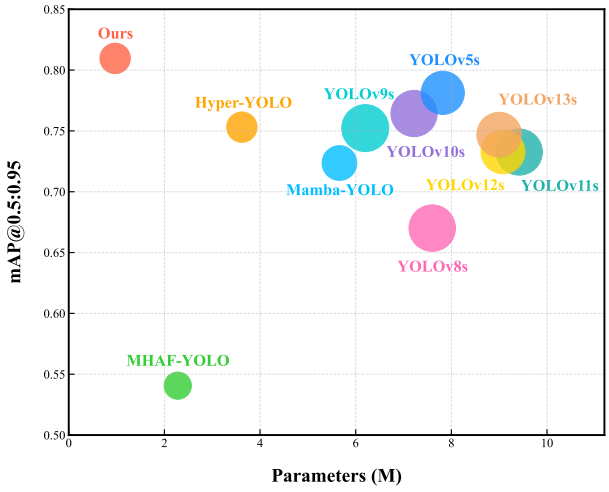


FIGURE 8. Accuracy–efficiency comparison of different models on the DeepPCB dataset. The horizontal axis represents the number of parameters in millions (M), the vertical axis represents mAP@0.5:0.95, and the bubble size encodes the computational cost in giga floating-point operations (GFLOPs). Larger bubbles indicate higher computational cost.

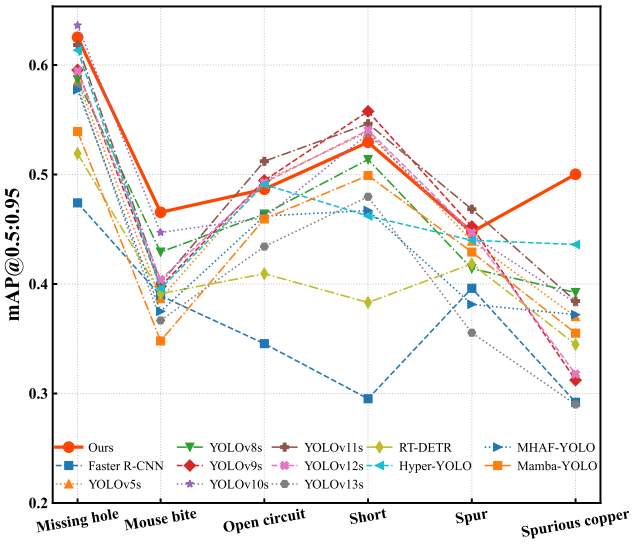


FIGURE 9. Performance comparison across individual defect categories on the PKU-Market-PCB dataset.

2) Performance on Individual Defect Categories

To further analyze the behavior of SHNet on specific defect types, we evaluate the category-level mAP@0.5:0.95 on the PKU-Market-PCB and DeepPCB datasets, as shown in Fig. 9 and Fig. 10. This comparison provides a more fine-grained view of the detection performance beyond the dataset-level metrics reported in Table 4.

On the PKU-Market-PCB dataset, Fig. 9 shows that SHNet achieves competitive performance across most defect categories. For Spurious copper, SHNet obtains a higher mAP@0.5:0.95 than the compared models, while Hyper-YOLO reaches 0.4361 on this category. SHNet also performs well on Missing hole and Mouse bite, suggesting that the proposed shallow high-resolution design is helpful for detecting defects with small size or weak visual contrast. However, for Open circuit and Short, some compared models achieve slightly better results than SHNet. This indicates that although SHNet is effective for several texture-confused or small-scale defects, there is still room for improvement when handling fine, narrow, or disconnected defect morphologies.

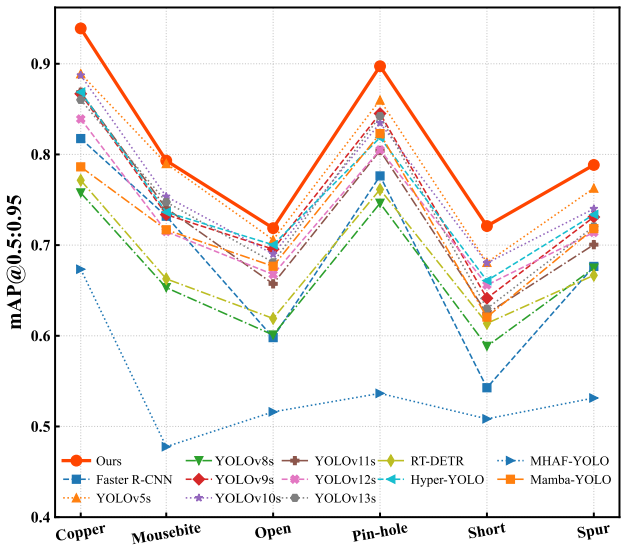


FIGURE 10. Performance comparison across individual defect categories on the DeepPCB dataset.

On the DeepPCB dataset, Fig. 10 presents the category-level comparison across the six defect types. Under the current experimental setting, SHNet achieves the highest mAP@0.5:0.95 among the compared methods for all six categories. The improvement is relatively clear for Copper and Pin-hole, where SHNet obtains mAP@0.5:0.95 values of 0.9390 and 0.8973, respectively. These results suggest that SHNet can maintain stable category-level detection performance on the grayscale DeepPCB dataset. Nevertheless, the results should be interpreted within the scope of the two evaluated datasets, and further validation on more diverse industrial PCB images would be needed to assess its broader generalization ability.

Overall, the category-level results indicate that preserving shallow high-resolution features is beneficial for detecting several small and low-contrast PCB defects that may be weakened by repeated downsampling. At the same time, the remaining gaps on categories such as Open circuit and Short suggest that future work may need to further improve the representation of slender or structurally discontinuous defects.

3) Heatmap Visualization and Analysis

We further employ heatmap visualization to qualitatively analyze the spatial response characteristics of different models. Brighter regions indicate stronger model responses and are expected to overlap with potential defect regions.

Fig. 11 presents representative examples from the DeepPCB dataset, where the original defect images, corresponding heatmaps, and final detection results are shown together.

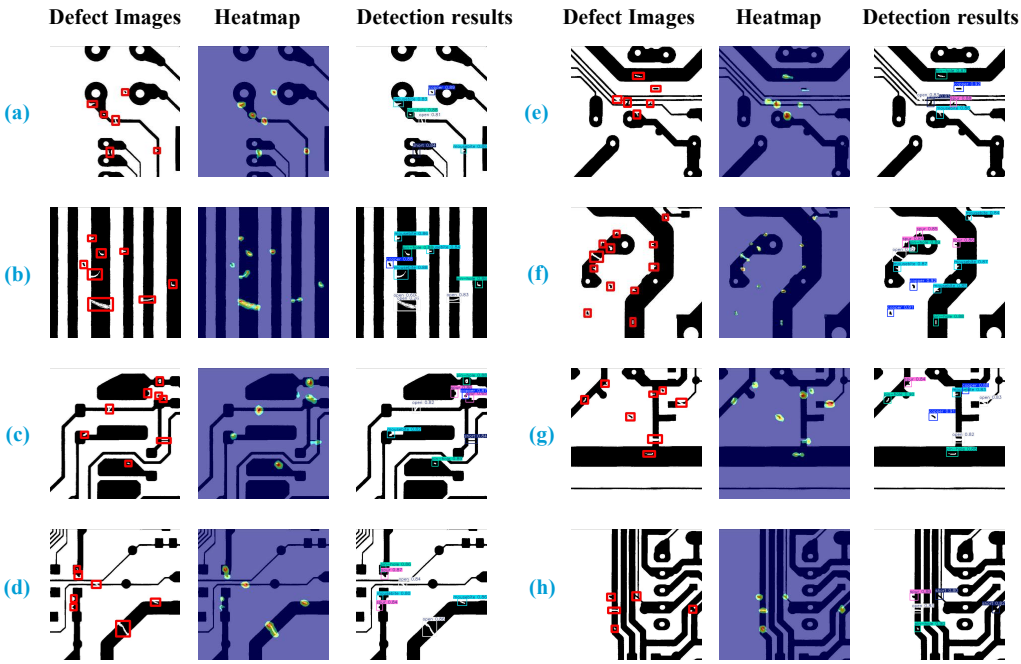


FIGURE 11. Representative detection results on the DeepPCB dataset, including input defect images, heatmaps, and predicted detection boxes. The visual results show that SHNet can produce concentrated responses around defect regions and maintain reliable localization under complex PCB textures.

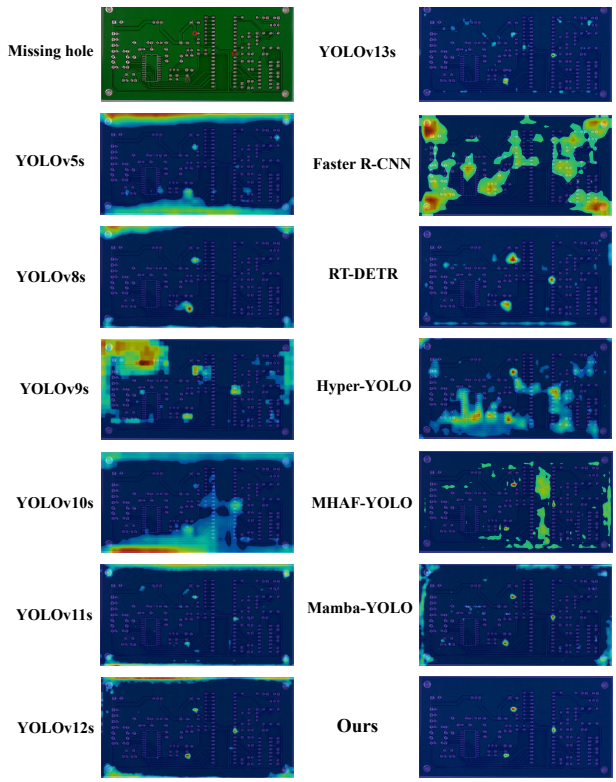


FIGURE 12. Heatmap comparison of different detection models for the same Missing Hole sample from the PKU-Market-PCB dataset. Brighter regions indicate stronger relative model responses at the corresponding spatial locations.

Across the selected cases, SHNet generally produces localized high-response regions around the annotated defects rather than activating broadly over the PCB background. This behavior is particularly evident in samples containing multiple defects distributed over different circuit structures, where several compact response peaks appear at spatially separated defect locations. Even in challenging cases with dense parallel traces, curved conductors, and repetitive geometric patterns, the dominant responses remain concentrated near the actual defect regions, while most irrelevant background structures receive relatively weak activation. Moreover, the locations of the major heatmap responses are largely consistent with the predicted bounding boxes, indicating that the features used for prediction are spatially correlated with the detected defects rather than being dominated by surrounding circuit patterns.

Fig. 12 further compares the response patterns of different models on a representative missing-hole sample from PKU-Market-PCB. Noticeable differences can be observed among the compared methods. YOLOv5s exhibits strong responses along the image boundaries and several non-defect regions, indicating considerable sensitivity to background structures. YOLOv9s and YOLOv10s produce relatively broad and spatially dispersed activation areas, with substantial responses extending beyond the defect locations. Faster R-CNN shows even more widespread high-response regions over large portions of the PCB, suggesting that its feature responses are strongly affected by complex circuit textures. Hyper-YOLO and MHAF-YOLO also generate fragmented or elongated activations along conductive traces and surrounding components, which may increase the risk of background interfer-

ence. Although RT-DETR and Mamba-YOLO exhibit more localized responses than several of the above models, noticeable activations can still be observed around peripheral or non-defect structures. In comparison, SHNet produces more compact and clearly separated response peaks that are concentrated around the actual missing-hole regions, while suppressing unnecessary activation over the surrounding PCB background. These qualitative observations are consistent with the quantitative improvements in mAP@0.5:0.95.

4) Ablation Study

To evaluate the contribution of the proposed architecture and individual modules, we conduct ablation experiments on PKU-Market-PCB and DeepPCB, as summarized in Table 5. SHNet-Base follows the same overall architecture as SHNet but uses the standard C2f and Concat operations of YOLOv8 without MKAM, SSAM, and WEM. The three modules are then ablated individually from the complete model while the remaining components are kept unchanged. In addition, an "Ours + 32×" variant is introduced by restoring the conventional P5/32× detection branch to directly examine the effect of removing this low-resolution branch.

Compared with YOLOv8s, SHNet-Base maintains comparable performance on PKU-Market-PCB, with mAP@0.5:0.95 changing only from 0.4666 to 0.4605, while substantially reducing model complexity. On DeepPCB, SHNet-Base further improves mAP@0.5:0.95 from 0.6701 to 0.7123. The different gains on the two datasets may be associated with differences in defect-scale distribution and image characteristics. Overall, these results indicate that emphasizing shallower high-resolution representations can provide a compact foundation without substantially compromising detection capability.

The influence of the 32× detection branch is more directly demonstrated by comparing "Ours + 32×" with SHNet. Restoring this branch results in mAP@0.5:0.95 values of 0.5076 and 0.8082 on PKU-Market-PCB and DeepPCB, respectively, compared with 0.5091 and 0.8096 for SHNet. The accuracy differences are minor, whereas the additional branch increases the computational cost from 9.5 to 11.4 GFLOPs, the parameter count from 0.97M to 1.35M, and the model size from 2.3 MB to 3.0 MB. This comparison shows that the 32× branch provides limited additional benefit for the evaluated PCB defects while introducing noticeable model overhead, supporting its removal from SHNet.

The module-wise ablations further verify the effectiveness of MKAM, SSAM, and WEM. On PKU-Market-PCB, removing MKAM, SSAM, and WEM reduces mAP@0.5:0.95 from 0.5091 to 0.4712, 0.4690, and 0.4306, respectively. WEM causes the largest degradation, accompanied by a decrease in recall from 0.9115 to 0.8341, indicating the importance of frequency-domain enhancement for weak defect responses. The performance reductions caused by removing MKAM and SSAM also confirm the benefits of scale-adaptive representation and spatial-semantic feature alignment.

A consistent trend is observed on DeepPCB. Removing MKAM, SSAM, and WEM decreases mAP@0.5:0.95 from 0.8096 to 0.7851, 0.7854, and 0.7483, respectively, with WEM again producing the largest degradation. Although the w/o SSAM variant obtains a slightly higher mAP@0.5 than SHNet (0.9888 versus 0.9867), SHNet achieves higher recall, F1 score, and mAP@0.5:0.95. This indicates that SSAM contributes more clearly to detection completeness and localization performance across stricter IoU thresholds.

Overall, the results demonstrate that the shallow high-resolution architecture and the three proposed modules play complementary roles. Removing the 32× branch improves the accuracy–efficiency trade-off, while MKAM, SSAM, and WEM enhance scale-aware representation, cross-level feature fusion, and weak defect responses, respectively. Their combination yields the best overall performance on both datasets.

V. CONCLUSION

This work presents SHNet, a lightweight PCB defect detector designed around shallow high-resolution representations. Beyond improving detection performance, the main scientific contribution of this study is to reconsider the conventional dependence on progressively deeper and lower-resolution features for PCB defect detection. For tiny and low-contrast defects, excessive downsampling can irreversibly weaken fine structural cues. SHNet therefore preserves shallow high-resolution features as the primary representations for detection while using the 16× feature mainly to provide semantic guidance. On this basis, MKAM, SSAM, and WEM address three complementary challenges: scale variation, cross-level spatial-semantic misalignment, and weak defect responses under complex background textures. The ablation results further indicate that these components contribute in complementary rather than interchangeable ways, providing empirical support for the proposed feature-preservation and enhancement strategy.

From a practical perspective, SHNet achieves this representation capability with only 0.97M parameters, a 2.3 MB model size, and 9.5 GFLOPs, while reaching 168.00 FPS on PKU-Market-PCB and 240.11 FPS on DeepPCB under the current experimental platform. The compact computational footprint makes SHNet a promising candidate for resource-constrained PCB inspection systems, where memory consumption, inference latency, and detection reliability must be considered simultaneously. More importantly, the results suggest that lightweight industrial inspection does not necessarily require sacrificing fine-grained defect representation when high-resolution information is explicitly preserved and selectively enhanced.

REFERENCES

[1] Y. Zhou, M. Yuan, J. Zhang, G. Ding, and S. Qin, "Review of vision-based defect detection research and its perspectives for printed circuit board," *Journal of Manufacturing Systems*, vol. 70, pp. 557–578, 2023, doi: 10.1016/j.jmsy.2023.08.019.

TABLE 5. Ablation study on PKU-Market-PCB and DeepPCB. SHNet-Base follows the same overall architecture as SHNet but uses the standard C2f and Concat operations of YOLOv8 without MKAM, SSAM, and WEM. The "w/o" variants remove one proposed module at a time while keeping the remaining components unchanged. "Ours + 32×" restores the conventional P5/32× detection branch to the complete SHNet. The best accuracy result for each dataset is shown in bold.

Model	32×	MKAM	SSAM	WEM	Precision	Recall	F1	mAP@0.5	mAP@0.5:0.95	GFLOPs	Params(M)	Size(MB)
PKU-Market-PCB dataset												
YOLOv8s [14]	✓				0.9075	0.8698	0.8876	0.9053	0.4666	21.6	7.60	14.8
SHNet-Base					0.9189	0.8094	0.8575	0.8763	0.4605	8.3	0.72	1.8
w/o MKAM			✓	✓	0.9277	0.8824	0.9040	0.9169	0.4712	9.3	0.98	2.3
w/o SSAM		✓		✓	0.8973	0.8561	0.8750	0.8941	0.4690	8.8	0.83	2.0
w/o WEM		✓	✓		0.9028	0.8341	0.8664	0.8646	0.4306	8.6	0.84	2.0
Ours + 32×	✓	✓	✓	✓	0.9538	0.9099	0.9313	0.9389	0.5076	11.4	1.35	3.0
Ours		✓	✓	✓	**0.9554**	**0.9115**	**0.9320**	**0.9398**	**0.5091**	9.5	0.97	2.3
DeepPCB dataset												
YOLOv8s [14]	✓				0.9464	0.9069	0.9254	0.9688	0.6701	21.6	7.60	14.8
SHNet-Base					0.9372	0.9428	0.9397	0.9762	0.7123	8.3	0.72	1.8
w/o MKAM			✓	✓	0.9752	0.9650	0.9697	0.9875	0.7851	9.3	0.98	2.3
w/o SSAM		✓		✓	0.9759	0.9602	0.9677	**0.9888**	0.7854	8.8	0.83	2.0
w/o WEM		✓	✓		0.9667	0.9107	0.9370	0.9798	0.7483	8.6	0.84	2.0
Ours + 32×	✓	✓	✓	✓	0.9771	0.9710	0.9740	0.9871	0.8082	11.4	1.35	3.0
Ours		✓	✓	✓	**0.9779**	**0.9725**	**0.9752**	0.9867	**0.8096**	9.5	0.97	2.3

[2] W. Jiang, T. Li, S. Zhang, W. Chen, and J. Yang, "PCB defects target detection combining multi-scale and attention mechanism," *Engineering Applications of Artificial Intelligence*, vol. 123, pp. 106359, 2023, doi: 10.1016/j.engappai.2023.106359.

[3] J. Tang, Z. Wang, H. Zhang, H. Li, P. Wu, and N. Zeng, "A lightweight surface defect detection framework combined with dual-domain attention mechanism," *Expert Systems with Applications*, vol. 238, pp. 121726, 2024, doi: 10.1016/j.eswa.2023.121726.

[4] T. Luo, Y. Zhou, D. Shi, Q. Yun, S. Wang, J. Zhang, and G. Ding, "A lightweight defect detection transformer for printed circuit boards combining image feature augmentation and refined cross-scale feature fusion," *Engineering Applications of Artificial Intelligence*, vol. 155, pp. 111128, 2025, doi: 10.1016/j.engappai.2025.111128.

[5] V. H. Gaidhane, Y. V. Hote, and V. Singh, "An efficient similarity measure approach for PCB surface defect detection," *Pattern Analysis and Applications*, vol. 21, no. 1, pp. 277–289, 2018, doi: 10.1007/s10044-017-0640-9.

[6] A. Malhi and R. X. Gao, "PCA-based feature selection scheme for machine defect classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 6, pp. 1517–1525, 2004, doi: 10.1109/TIM.2004.834070.

[7] G. Biau and E. Scornet, "A random forest guided tour," *TEST*, vol. 25, no. 2, pp. 197–227, 2016, doi: 10.1007/s11749-016-0481-7.

[8] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Networks*, vol. 107, pp. 3–11, 2018, doi: 10.1016/j.neunet.2017.12.012.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014, doi: 10.1109/CVPR.2014.81.

[10] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, 2015, doi: 10.1109/ICCV.2015.169.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.

[13] G. Jocher, *et al.*, "YOLOv5 by Ultralytics," 2020, URL: https://github.com/ultralytics/yolov5.

[14] G. Jocher, *et al.*, "YOLOv8 by Ultralytics," 2023, URL: https://github.com/ultralytics/ultralytics.

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *European Conference on Computer Vision*, pp. 21–37, 2016, doi: 10.1007/978-3-319-46448-0_2.

[16] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10781–10790, 2020, doi: 10.1109/CVPR42600.2020.01079.

[17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021, URL: https://arxiv.org/abs/2010.11929.

[18] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *First Conference on Language Modeling*, 2024, doi: 10.48550/arXiv.2312.00752.

[19] X. Liu, "An Adaptive Defect-Aware Attention Network for Accurate PCB-Defect Detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 5040811, 2024, doi: 10.1109/TIM.2024.3488158.

[20] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, Q. Ye, J. Jiao, and Y. Liu, "VMamba: Visual state space model," in *Advances in Neural Information Processing Systems*, vol. 37, pp. 103031–103063, 2024, doi: 10.48550/arXiv.2401.10166.

[21] Z. Wang, C. Li, H. Xu, X. Zhu, and H. Li, "Mamba YOLO: A simple baseline for object detection with state space model," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 8, pp. 8205–8213, 2025, doi: 10.1609/aaai.v39i8.32885.

[22] K. Li, X. Zhong, and Y. Han, "A high-performance small target defect detection method for PCB boards based on a novel YOLO-DFA algorithm," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 2008312, 2025, doi: 10.1109/TIM.2025.3551584.

[23] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8759–8768, 2018, doi: 10.48550/arXiv.1803.01534.

[24] D. Fan and H. Wu, "Improving PCB defect detection with KAN-YOLO: A knowledge-enhanced feature fusion approach," in *2025 IEEE 5th International Conference on Electronic Technology, Communication and Information*, pp. 1348–1353, 2025, doi: 10.1109/ICETCI64844.2025.11084156.

[25] S. Wu, X. Lu, C. Guo, and H. Guo, "MV-YOLO: An efficient small object detection framework based on Mamba," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 63, pp. 5632814, 2025, doi: 10.1109/TGRS.2025.3584955.

[26] Y. Li, T. Yao, Y. Pan, and T. Mei, "Contextual Transformer Networks for Visual Recognition," *arXiv preprint arXiv:2107.12292*, 2021, doi: 10.48550/arXiv.2107.12292.

[27] T. Liu, G.-Z. Cao, Z. He, and S. Xie, "Refined defect detector with deformable transformer and pyramid feature fusion for PCB detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–11, 2023, doi: 10.1109/TIM.2023.3326460.
[28] J. Yang, Z. Liu, W. Du, and S. Zhang, "A PCB defect detector based on coordinate feature refinement," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–10, 2023, doi: 10.1109/TIM.2023.3322483.
[29] W. Chen, S. Meng, and X. Wang, "Local and global context-enhanced lightweight CenterNet for PCB surface defect detection," *Sensors*, vol. 24, no. 14, pp. 4729, 2024, doi: 10.3390/s24144729.
[30] X. Li, H. Yan, K. Cui, Z. Li, R. Liu, G. Lu, K. C. Hsieh, X. Liu, and C. Hon, "A novel hybrid YOLO approach for precise paper defect detection with a dual-layer template and an attention mechanism," *IEEE Sensors Journal*, vol. 24, no. 7, pp. 11651–11669, 2024, doi: 10.1109/JSEN.2024.3356356.
[31] C. Chen, Q. Wu, J. Zhang, H. Xia, P. Lin, Y. Wang, M. Tian, and R. Song, "U2D2PCB: Uncertainty-aware unsupervised defect detection on PCB images using reconstructive and discriminative models," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–10, 2024, doi: 10.1109/TIM.2024.3386210.
[32] C. Mo, Z. Hu, J. Wang, and X. Xiao, "SGT-YOLO: A lightweight method for PCB defect detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 3543911, 2025, doi: 10.1109/TIM.2025.3563011.
[33] W. Yan, H. Yan, N. Xu, X. Su, and A. Jiang, "Real-time detection algorithm for industrial PCB defects based on HS-LSKA and global-EMA," *Journal of Electronic Imaging*, vol. 34, no. 3, pp. 033005, 2025, doi: 10.1117/1.jei.34.3.033005.
[34] L. Dong, H. Zhu, H. Ren, T.-Y. Lin, and K.-P. Lin, "A novel lightweight MT-YOLO detection model for identifying defects in permanent magnet tiles of electric vehicle motors," *Expert Systems with Applications*, vol. 288, pp. 128247, 2025, doi: 10.1016/j.eswa.2025.128247.
[35] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision*, pp. 3–19, 2018, doi: 10.1007/978-3-030-01234-2_1.
[36] Peking University, Open Lab for Intelligent Robots, "PKU-Market-PCB," 2019, URL: https://robotics.pkusz.edu.cn/resources/dataset, [Dataset].
[37] Charmve, "Surface defect detection: Dataset and papers," 2020, URL: https://github.com/Charmve/Surface-Defect-Detection/tree/master/DeepPCB, [Dataset].
[38] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," in *European Conference on Computer Vision*, pp. 1–21, 2024, doi: 10.48550/arXiv.2402.13616.
[39] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, *et al.*, "YOLOv10: Real-time end-to-end object detection," in *Advances in Neural Information Processing Systems*, vol. 37, pp. 107984–108011, 2024, doi: 10.48550/arXiv.2405.14458.
[40] R. Khanam and M. Hussain, "YOLOv11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024, URL: https://arxiv.org/abs/2410.17725.
[41] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025, URL: https://arxiv.org/abs/2502.12524.
[42] M. Lei, S. Li, Y. Wu, H. Hu, Y. Zhou, X. Zheng, G. Ding, S. Du, Z. Wu, and Y. Gao, "YOLOv13: Real-time object detection with hypergraph-enhanced adaptive visual perception," *arXiv preprint arXiv:2506.17733*, 2025, URL: https://arxiv.org/abs/2506.17733.
[43] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "DETRs beat YOLOs on real-time object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16965–16974, 2024, doi: 10.1109/CVPR52733.2024.01605.
[44] Y. Feng, J. Huang, S. Du, S. Ying, J.-H. Yong, Y. Li, G. Ding, R. Ji, and Y. Gao, "Hyper-YOLO: When visual object detection meets hypergraph computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 4, pp. 2388–2401, 2025, doi: 10.48550/arXiv.2408.04804.
[45] Z. Yang, Q. Guan, Z. Yu, X. Xu, H. Long, S. Lian, H. Hu, and Y. Tang, "MHAF-YOLO: Multi-branch heterogeneous auxiliary fusion YOLO for accurate object detection," *arXiv preprint arXiv:2502.04656*, 2025, URL: https://arxiv.org/abs/2502.04656.

YU LIU received the B.S. degree from Southwest Petroleum University. She is currently pursuing the M.S. degree in Information and Communication Engineering with the Beijing Institute of Graphic Communication, Beijing, China. Her main research interests include artificial intelligence and computer vision.

CHENGJIE FANG is currently a graduate student majoring in artificial intelligence at the University of Chinese Academy of Sciences, Beijing, China. His primary research interests include embodied intelligence and marine science.

LIKUN LU is an Associate Professor with the School of Information Engineering, Beijing Institute of Graphic Communication. He serves as the Director of the Laboratory for Digital Asset Evaluation of Publications under the National Technical Committee for Standardization of Press Distribution, and the Deputy Director of the Key Technology Application Research and Service Laboratory for the Press and Publication Industry of the National Press and Publication Administration. He is also the leader of the Beijing Academic Innovation Team on Digital Printing and Inspection. His research interests include artificial intelligence, computer vision, and inkjet digital printing technology.