Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 4, August 2026, pp. 3070~3079
ISSN: 2302-9285, DOI: 10.11591/eei.v15i4.10873

Decoder-only transformer with multi-scale attention for efficient printed circuit board defect inspection

Chi Kien Ha, Hoanh Nguyen
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

Article Info

Article history:

Received Jun 16, 2025
Revised May 22, 2026
Accepted Jun 19, 2026

Keywords:

Defect detection
Deformable attention
Encoder-decoder
Multi-scale features
Transformer structure

ABSTRACT

Printed circuit boards (PCBs) are critical components in modern electronics, detecting defects rapidly, and accurately is crucial in manufacturing. We propose a novel real-time PCB defect detection model based on a Transformer decoder-only architecture. The framework first extracts multi-scale features via a backbone with a feature pyramid network (FPN) and generates candidate regions using a region proposal network (RPN). These region proposals are encoded as query embeddings that drive an adaptive multi-scale deformable attention (AMDA) module in the Transformer decoder, replacing the standard encoder-decoder attention. By dynamically weighting multi-scale feature maps for each query, AMDA emphasizes the feature scale most relevant to each defect's size and texture, and yielding enhanced discriminative representations for subtle defect detection. The decoder-only design drastically reduces computational overhead compared to full encoder-decoder Transformers, enabling real-time inference, and faster training convergence. Experiments on benchmark PCB defect datasets demonstrate that our approach outperforms state-of-the-art methods in both accuracy and speed. The proposed model's efficiency and high precision make it well suited for deployment in fast-paced PCB manufacturing lines that demand stringent real-time performance and reliable defect detection.

Corresponding Author:

Hoanh Nguyen
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City
12 Nguyen Van Bao, Ho Chi Minh City, Vietnam
Email: nguyenhoanh@iuh.edu.vn

1. INTRODUCTION

Printed circuit boards (PCBs) form the backbone of virtually all electronic systems, and ensuring their structural and functional integrity is critical. Even small defects such as misaligned components, solder bridging, or micro-cracks can lead to device malfunction, safety hazards, or costly product recalls. With ever-increasing production volumes and stringent reliability requirements, automated optical inspection (AOI) systems have become indispensable for quality assurance in PCB manufacturing. However, conventional AOI methods based on template matching or classical machine learning often struggle to generalize to new defect types and are sensitive to variations in illumination or background conditions. This limitation highlights the urgent need for robust and adaptive inspection solutions that can meet industrial demands at scale.

In recent years, deep learning–based methods, particularly convolutional neural networks (CNNs), have achieved notable progress in industrial defect detection by learning rich feature representations directly from large amounts of data. Numerous CNN-based techniques have been applied to PCB inspection [1]–[4] and have demonstrated improved localization and classification accuracy in real-world settings [5]–[12]. Some studies leverage multi-scale feature fusion or attention mechanisms to capture subtle anomalies, while others propose specialized architectures tailored to the diverse and complex textures of PCB surfaces. Despite these advances, CNN-based detectors often involve multiple processing stages and require carefully tuned anchors or prior boxes. Such designs become inefficient when defect sizes vary significantly, and they frequently impose trade-offs between accuracy and throughput.

Transformer-based detectors have recently emerged as a promising alternative by reframing object detection as a direct set prediction task. The pioneering detection Transformer (DETR) model [13] introduced a Transformer encoder-decoder architecture that performs global context reasoning without many handcrafted components. While effective in principle, DETR suffers from high computational cost and slow convergence, especially on the high-resolution images typical in PCB inspection. To address these shortcomings, several extensions have been developed: deformable DETR [14] accelerates training with sparse multi-scale attention; Sparse DETR [15] filters encoder tokens; Conditional DETR [16] incorporates conditional queries; Anchor DETR [17]; DESTR [18] add anchor priors; real-time-DETR (RT-DETR) [19] eliminates the encoder for real-time detection; and DETR with improved denoising (DINO) [20] introduces improved training strategies. These methods enhance efficiency and accuracy, yet most retain the full encoder-decoder pipeline, resulting in large model sizes and inference latency that remain unsuitable for real-time industrial inspection.

Despite the progress of CNN-and Transformer-based approaches, two key challenges remain unresolved: i) achieving real-time throughput without compromising accuracy in detecting tiny and subtle PCB defects and ii) effectively leveraging multi-scale features in an adaptive and query-specific manner. Existing CNN methods often lack flexibility in handling variable defect scales, while Transformer-based models, though powerful, and introduce excessive computational overhead. These gaps underscore the need for a new framework that balances accuracy, efficiency, and scalability in high-resolution PCB inspection scenarios.

Motivated by these challenges, we propose a novel Transformer decoder-only framework with adaptive multi-scale deformable attention (AMDA). Our contributions are fourfold: i) we eliminate the encoder stage and introduce a lightweight region-proposal module, drastically reducing computational complexity while preserving contextual reasoning; ii) we develop the AMDA mechanism, which employs query-dependent gating to adaptively fuse multi-scale feature maps for optimal representation of both small and large defects; iii) through comprehensive experiments on benchmark datasets, we demonstrate that the proposed model achieves state-of-the-art accuracy while maintaining real-time inference speeds; and iv) we show the practical relevance of this approach for modern AOI systems, enabling robust deployment in high-throughput manufacturing environments. The remainder of this paper is organized as follows: section 2 details the proposed methodology, section 3 presents experimental results and ablation studies, and section 4 concludes with discussions on industrial applicability and future research directions.

2. METHOD

2.1. Overall structure

Transformer-based detectors like DETR [13] reformulate object detection as direct set prediction via bipartite matching, removing the need for hand-crafted anchors, and post-processing. However, DETR's dense global attention and reliance on a heavy Transformer encoder result in slow convergence and high computational cost on high-resolution inputs. Deformable DETR [14] alleviates some inefficiency by attending only to sparse reference points across multiple scales, but it still requires both encoder and decoder modules, leading to large model sizes, and latency that are unsuitable for real-time PCB inspection. To improve efficiency, recent studies have explored selective Transformer usage: thin plate spline-fully convolutional one-stage object detector (TPS-FCOS) [21] integrates a Transformer encoder into a single-stage CNN detector (FCOS [22]) for global context, while RT-DETR [19] eliminates the encoder and uses only a lightweight decoder for high-speed detection. These studies confirm that removing redundant modules can enhance throughput, yet they do not fully resolve the challenge of adaptively leveraging multi-scale features, a key requirement for detecting both subtle and large PCB defects.

To address these limitations, we propose a decoder-only Transformer architecture that combines a conventional region proposal pipeline with an AMDA mechanism. Figure 1 illustrates the overall structure of the proposed model for real-time PCB defect detection. The model consists of a backbone network, FPN, RPN, and a Transformer decoder-only pipeline with AMDA. Our model begins with a standard backbone network that extracts intermediate feature maps at multiple levels, denoted as C_3, C_4, C_5. These features capture increasingly rich semantic information as we progress through the backbone layers. To further enhance multi-scale representation, a feature pyramid network (FPN) [23] is applied to the output of the backbone. The FPN produces a set of feature maps $\{P_3, P_4, P_5, P_6\}$, each corresponding to a different spatial resolution. This design ensures that both high-resolution features at lower feature levels and rich-semantic features at higher feature levels are available. A region proposal network (RPN) then operates on these feature maps to generate a set of candidates bounding boxes region of interests (RoIs) that potentially contain defects. By efficiently scanning across the multi-level feature pyramid, the RPN can handle small and large objects alike. From each detected RoI, we perform RoI align [24] to extract the aligned region features from the appropriate P_i. RoI align preserves spatial alignment by avoiding harsh quantization, ensuring that subsequent layers receive high-quality, position-sensitive features. The region-aligned features are subsequently passed through a fully connected (FC) layer. This FC layer compresses and transforms the RoI features into a more compact embedding representation, which serves as the query input to the Transformer decoder.

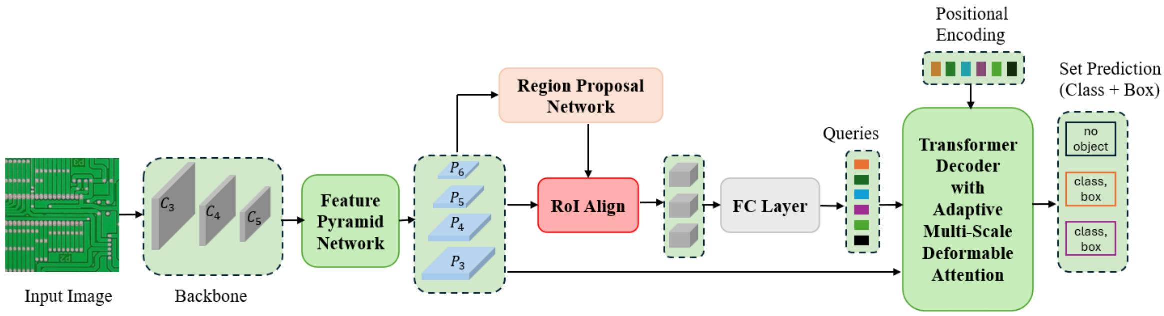

Figure 1. Overall architecture of the proposed PCB defect detection model

In the Transformer decoder, each query vector is enhanced by attending to the feature pyramid outputs. Unlike standard DETR-like models that perform cross-attention only with the final encoder output, we allow the decoder to attend to all levels of the FPN output. We replace the usual cross-attention with our proposed AMDA module (detailed in subsection 2.3). AMDA leverages query-dependent gating to fuse multi-scale feature representations. Specifically, AMDA learns a set of scale-specific weights for each query based on the query's current feature. These weights determine the relative importance of different feature pyramid levels. Hence, objects that are more reliably detected at a particular scale receive a higher weight for that scale's feature map. Concretely, the gating mechanism takes as input the query embedding and predicts per-scale gating coefficients via a sigmoid-activated linear layer. These coefficients modulate the outputs of deformable cross-attention, which selectively samples a sparse set of locations from each scale. By aggregating the attention outputs with learned weights, AMDA effectively captures the most discriminative features for each RoI. This stands in contrast to standard Transformer cross-attention, where only the final encoder layer is used. While the final layer may capture stronger semantics overall, it may not be the optimal source for every object size or shape. Due to this adaptive scheme, each query can focus on the scale level that provides the highest utility for classification and localization. Small defects can leverage the higher-resolution lower pyramid levels, while larger defects or more context-heavy regions may rely on the deeper, semantically richer levels.

Once the decoder's outputs have been refined by AMDA, they undergo a set prediction stage to produce final bounding boxes and defect classifications. Each query either predicts a valid object with class and box or a "no object" label under a bipartite matching strategy similar to DETR. Through the integration of FPN-based proposals and AMDA's gated multi-scale fusion, our method maintains rapid inference while offering strong accuracy across different defect sizes. The outcome of this design is a more robust detection framework that capitalizes on multi-scale features without sacrificing real-time performance. By confining Transformer computation to the decoder and using AMDA to adaptively weigh feature maps, we avoid the overhead of a large Transformer encoder.

2.2. Adaptive multi-scale deformable attention

To effectively utilize multi-scale features for object queries, we propose AMDA mechanism (illustrated in Figure 2). In the standard deformable DETR formulation [14], each query attends to a small set of reference points in feature maps drawn from multiple scales, thereby reducing computational overhead and accelerating convergence. However, merely sampling from multiple scales without adaptive weighting can be suboptimal for objects of different sizes or shapes. AMDA addresses this by learning query-dependent weighting to selectively aggregate the most relevant scale information.

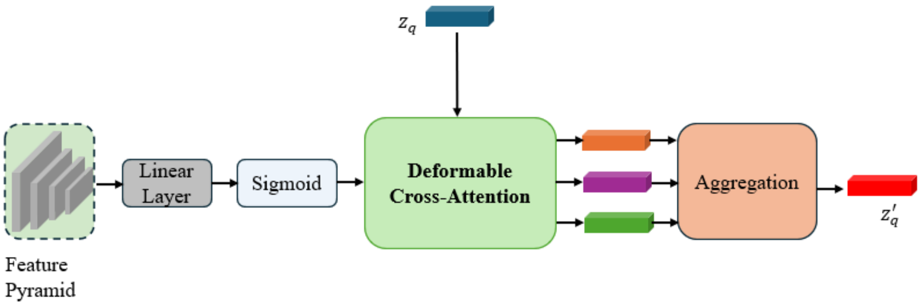

Figure 2. Illustration of AMDA mechanism

Formally, let $Q \in \mathbb{R}^{d \times N}$ be the set of N query embeddings, each of dimension d. Suppose we also have multi-level FPN features $\mathcal{E} = \{P_3, \dots, P_L\}$, where each P_l corresponds to one scale level. Following the deformable DETR approach, each query q (a column in Q) attends to a sparse set of reference points on each scale. For scale l, we define:

$$A_l = W_a q \quad (1)$$

$$\Delta r = W_p q \quad (2)$$

$$V_l = sampling(W_v^l P_l, r + \Delta r) \quad (3)$$

Here, r denotes the reference point in normalized 2D coordinates, Δr is the predicted offset for that query, and V_l is the sampled feature from P_l. The standard deformable cross-attention for scale l is then computed by:

$$cross\_att = Softmax(A_l)V_l \quad (4)$$

To adaptively combine information from multiple scales, AMDA learns a fusion weight for each scale. Specifically, we compute:

$$w_e = \sigma(W_e q) \in \mathbb{R}^L \quad (5)$$

where $\sigma(.)$ is the sigmoid function that ensures each component $w_e^l \in (0,1)$. These scale-wise weights reflect the relevance of each feature level for the current query. Finally, we obtain the output of AMDA by weighting each scale's deformable attention result:

$$\text{AMDA}(\text{q}, \mathcal{E}) = \sum_{l=1}^{L} w_e^l . Softmax(A_l)V_l \quad (6)$$

By introducing query-dependent gating, AMDA effectively emphasizes scales that offer more discriminative cues for each object query, while de-emphasizing less informative ones. This mechanism is particularly valuable for multi-scale object detection in complex scenarios, such as PCB defect detection, where small defects demand high-resolution features and larger or context-rich defects may benefit more from semantically deeper layers.

Our design combines standard modules with the novel AMDA mechanism, chosen deliberately for industrial AOI constraints. The use of FPN+RPN reduces the number of Transformer tokens by focusing only on likely defect regions, justifying why the encoder can be safely removed without losing contextual reasoning. The AMDA module ensures scale adaptivity without excessive computational cost, aligning with the dual requirements: i) speed and ii) fine-grained multi-scale detection. We provide full implementation details to enable replication. The backbone is ResNet-34 pre-trained on ImageNet [25], the optimizer is AdamW with specific learning rate schedules, and loss functions include focal loss for classification and generalized intersection over union (L1+GIoU) for bounding box regression (see subsection 3.2). All training and inference settings are reported, ensuring that our method is transparent and reproducible.

3. RESULTS AND DISCUSSION

3.1. Dataset and evaluation metrics

We evaluate the proposed approach on two publicly available PCB defect datasets: DeepPCB [6] and thorough defect detection-PCB (TDD-PCB) [9]. These datasets collectively cover a broad spectrum of common defect types and capture realistic variations encountered in PCB inspection. DeepPCB dataset comprises 1,500 image pairs, each containing a defect-free template image and a corresponding tested image of the same PCB board, aligned at the pixel level. The annotations focus on six common defect types: open, short, mouse bite, spur, pin hole, and spurious copper. DeepPCB thus provides a controlled scenario where defects are relatively localized and can be compared against a nearly identical reference board image. TDD-PCB is a larger dataset with 10,668 PCB images, which are typically divided into smaller 600×600 patches for detailed inspection. TDD-PCB encompasses six primary defect classes similar to DeepPCB: missing hole, mouse bite, open, short, spur, and spurious copper. The larger size and broader variety of samples in TDD-PCB make it particularly suitable for evaluating scalability and robustness under more diverse inspection conditions.

For evaluation, we adopt standard object detection metrics. We report average precision (AP) at intersection over union (IoU) thresholds of 0.50 (AP50) and 0.75 (AP75), following common practice. AP50 is a looser criterion that primarily evaluates detection/classification success, while AP75 is stricter on localization precision. We also summarize the mean average precision (mAP) across multiple IoU thresholds where relevant. In addition, to assess efficiency, we report the number of model parameters and the computational cost measured in floating point operations (FLOPs) (floating-point operations). We also measure the inference speed in frames per second (FPS) on a single graphics processing unit (GPU). These metrics together provide a comprehensive view of the accuracy-speed trade-off, which is critical for practical deployment in manufacturing environments.

3.2. Experimental setups

Our model is implemented in PyTorch and we use a ResNet-34 backbone pre-trained on ImageNet [25] for a good balance of speed and accuracy. The FPN generates pyramid features at four levels. The RPN uses default anchor sizes tuned for PCB defect dimensions, and we select the top 100 proposals per image to feed into the Transformer decoder as queries. Minimal data augmentation is applied: we use random horizontal flips and minor rescaling, avoiding excessive color jittering to preserve the natural appearance of PCB patterns.

We train the model for 100 epochs on each dataset. The optimizer is AdamW with weight decay 1×10^{-4}. The learning rate is set to 1×10^{-4} for the Transformer parameters and 1×10^{-5} for the CNN backbone; we decay the learning rates by a factor of 10 after 50 epochs. We employ the DETR-style bipartite matching loss: specifically, a focal loss for classification and a combination of L1 loss and generalized IoU loss for bounding box regression. The loss weights are tuned on a validation split to balance classification and localization performance. We train with a batch size of 4 on an NVIDIA RTX 4080 GPU. To prevent overfitting, an early stopping criterion is used: if the validation mAP does not improve for 5 consecutive epochs, training is halted. During inference, the highest-confidence predictions are retained after bipartite matching, and non-max suppression is avoided to preserve the direct set-prediction design. For each experiment, we compare both quantitative results and practical runtime speed to ensure that our proposed method meets real-time requirements.

3.3. Experimental results

We compare our model against several classical and state-of-the-art detectors. Table 1 reports results on the DeepPCB dataset. Our approach outperforms both traditional CNN-based detectors and recent Transformer-based methods in both accuracy and efficiency. In terms of AP50/AP75, our model achieves the highest scores while also being the fastest. Faster region-based convolutional neural network (Faster R-CNN) achieves a solid 95.2% AP50 and 88.9% AP75 but requires 42.5 M parameters and delivers only 30 FPS, underscoring its relatively high computational demands. Single shot multibox detector (SSD) offers a higher speed of 40 FPS but at the cost of reduced accuracy (AP50=92.5% and AP75=85.3%), indicating its limitations on the DeepPCB dataset. RetinaNet balances accuracy and speed more effectively (95.8% AP50, 89.6% AP75, and 34 FPS), yet still trails behind newer architectures. CornerNet and YOLOv8 both exhibit decent detection capabilities, with YOLOv8 notably hitting 50 FPS, though neither achieves the top-tier accuracy levels. Moving to Transformer-based approaches, DETR and RT-DETR achieve high AP50 scores (97.1% and 98.2%, respectively), though both have relatively large FLOPs, highlighting the trade-off between attention-based architectures and efficiency. Deformable DETR further improves detection performance (98.6% AP50 and 93.8% AP75) but at 40.6 M parameters and 150.2 G FLOPs, making it more resource-intensive. In contrast, our model not only attains the highest AP50 (98.9%) but also secures a competitive AP75 of 94.0%. Notably, we achieve this with only 17.9 M parameters (less than half of deformable DETR's parameters) and 81.6 G FLOPs, which is nearly 2× more efficient than deformable DETR in computation. Our model also runs at 58 FPS, significantly faster than all other detectors, including YOLOv8. This superior balance of top-tier accuracy and efficiency underscores the effectiveness of our design. The decoder-only architecture with targeted multi-scale attention allows us to surpass prior state-of-the-art results on DeepPCB while operating in real-time. The contribution of the AMDA mechanism is evident here: by selectively concentrating on the pertinent scale for each defect, the model robustly detects both tiny and large defects without wasting computation on irrelevant features. Overall, our approach is exceptionally well-suited for real-time, high-precision PCB inspection.

Table 1. Comparison of different methods on the DeepPCB dataset

Models	AP50 (%)	AP75 (%)	Params (M)	FLOPs (G)	FPS
Faster R-CNN [26]	95.2	88.9	42.5	180.8	30
SSD [27]	92.5	85.3	34.3	80.7	40
RetinaNet [28]	95.8	89.6	36.9	120.3	34
CornerNet [29]	93.7	86.4	39.1	130.3	31
YOLOv8 [30]	96.5	90.3	18.2	80.9	50
DETR [13]	97.1	91.5	41.1	187.5	29
RT-DETR [19]	98.2	94.1	38.0	140.7	42
Deformable DETR [14]	98.6	93.8	40.6	150.2	39
Our model	98.9	94.0	17.9	81.6	58

Table 2 shows a similar pattern of results on the TDD-PCB dataset. Although Faster R-CNN achieves 94.1% AP50 and 87.2% AP75, it continues to suffer from high FLOPs (180.8 G) and runs at 30 FPS. SSD maintains a moderate 40 FPS but lags in accuracy (91.2% AP50 and 83.6% AP75). RetinaNet and CornerNet both push AP50 above 93%, yet remain outperformed by newer models in accuracy and speed. YOLOv8 remains the fastest among conventional detectors, peaking at 50 FPS, albeit with 95.6% AP50 and 89.2% AP75. Transformer-based solutions again dominate the upper accuracy tiers, with deformable DETR reaching an impressive 98.8% AP50 and 93.5% AP75. However, these improvements often entail heavier model sizes and greater computational demands. Our model stands out by delivering the top AP50 (99.1%) and a similarly high AP75 of 93.2% while keeping parameters at 17.9 M and FLOPs at 81.6 G. Crucially, it operates at 58 FPS, faster than all competing methods, solidifying its status as both highly accurate and computationally efficient. Consequently, the strong performance on TDD-PCB affirms that our proposed architecture generalizes effectively across different PCB defect detection benchmarks.

Table 2. Comparison of different methods on the TDD-PCB dataset

Models	AP50 (%)	AP75 (%)	Params (M)	FLOPs (G)	FPS
Faster R-CNN [26]	94.1	87.2	42.5	180.8	30
SSD [27]	91.2	83.6	34.3	80.7	40
RetinaNet [28]	95.0	88.1	36.9	120.3	34
CornerNet [29]	93.0	85.0	39.1	130.3	31
YOLOv8 [30]	95.6	89.2	18.2	80.9	50
DETR [13]	96.4	90.6	41.1	187.5	29
RT-DETR [19]	97.0	91.5	38.0	140.7	42
Deformable DETR [14]	98.8	93.5	40.6	150.2	39
Our model	99.1	93.2	17.9	81.6	58

In addition to quantitative metrics, we provide qualitative results in Figure 3. From the detection results, it is evident that our model successfully localizes diverse defect types, from minuscule soldering irregularities to larger component issues, with minimal false alarms. This capability underscores the robust multi-scale reasoning that emerges from our AMDA module. The figure also visualizes the attention heatmaps from the decoder for each detected defect. These heatmaps illustrate how the model's attention is distributed across scales for different queries. We observe that for small defects, the attention concentrates on higher-resolution feature maps with finer detail, whereas for larger defects or those requiring more context, the attention shifts toward lower-resolution, deeper features. This behavior confirms that the learned gating in AMDA is effectively routing each query to the appropriate scale. The heatmaps also show the effect of the deformable offsets: the attention is focused on specific locations of interest rather than broadly over the entire image, which helps suppress background interference. Such interpretability is valuable in an industrial setting, as it provides insight into why the model is making a certain prediction, which can help engineers trust and refine the system. Overall, the combination of high detection accuracy and transparent multi-scale attention strategies makes our model attractive for deployment in PCB inspection.

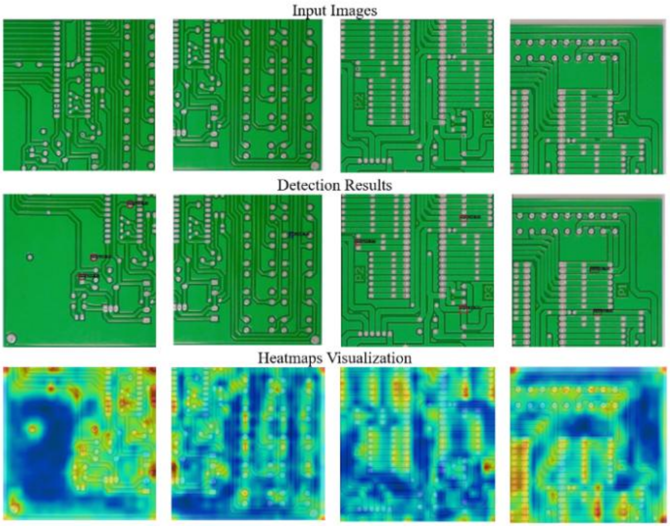

Figure 3. Qualitative results of the proposed model on PCB defect detection

3.4. Ablation experiments

To thoroughly evaluate the contributions of each component in our proposed model, we conduct a series of ablation studies on both DeepPCB and TDD-PCB datasets. We progressively build up from a baseline configuration, introducing the FPN, the deformable attention mechanism, the gating module, and finally our complete AMDA. For consistency, all models are trained under the same settings, with identical backbone, learning schedule, and data augmentations. Tables 3 and 4 summarize the results for DeepPCB and TDD-PCB, respectively.

Table 3. Ablation study on the DeepPCB dataset

Variant	FPN	Deformable	Gating	Params (M)	AP50 (%)	AP75 (%)	FPS
Baseline	✗	✗	✗	12.1	85.2	68.9	101
+ FPN	✓	✗	✗	15.6	87.5	71.0	85
+ Deform	✓	✓	✗	16.2	89.4	72.6	90
+ Gating	✓	✓	✓	16.5	90.8	74.6	78
+ AMDA	✓	✓	✓	17.9	98.9	94.0	58

Table 4. Ablation study on the TDD-PCB dataset

Variant	FPN	Deformable	Gating	Params (M)	AP50 (%)	AP75 (%)	FPS
Baseline	✗	✗	✗	12.1	86.3	67.2	101
+ FPN	✓	✗	✗	15.6	88.4	70.5	85
+ Deform	✓	✓	✗	16.2	89.9	73.1	90
+ Gating	✓	✓	✓	16.5	92.4	81.6	78
+ AMDA	✓	✓	✓	17.9	99.1	93.2	58

In Table 3, the “Baseline” configuration, consisting of a straightforward CNN plus Transformer decoder without FPN or deformable attention, obtains 85.2% AP50 and 68.9% AP75 on the DeepPCB dataset, using 12.1 M parameters and operating at 101 FPS. Introducing FPN (“+ FPN”) raises the parameter count to 15.6 M but improves multi-scale feature representation, resulting in 87.5% AP50 and 71.0% AP75 at 85 FPS. Replacing standard cross-attention with deformable attention (“+ Deform”) provides a further boost to 89.4% AP50 and 72.6% AP75, albeit at 16.2 M parameters and 90 FPS. The subsequent addition of a gating mechanism (“+ Gating”) refines spatial focus, elevating performance to 90.8% AP50 and 74.6% AP75 (16.5 M parameters) but reducing throughput to 78 FPS. Finally, our full AMDA module applies query-dependent gating on top of deformable cross-attention while integrating all multi-scale feature maps. This yields a notably higher 98.9% AP50 and 94.0% AP75 at 58 FPS with 17.9 M parameters. Although the additional modules progressively reduce speed, each component incrementally enhances detection accuracy for PCB defects, culminating in a significant gain when AMDA is fully deployed.

A similar progression emerges in Table 4 on the TDD-PCB dataset. The baseline model again achieves modest accuracy (86.3% AP50 and 67.2% AP75) but maintains the highest speed (101 FPS) with 12.1 M parameters. Augmenting it with FPN to leverage multi-scale features boosts AP50 to 88.4% and AP75 to 70.5%, though the FPS drops to 85 and parameters grow to 15.6 M. Enabling deformable cross-attention further refines the model's spatial sampling to reach 89.9% AP50 and 73.1% AP75 at 90 FPS. Introducing the gating module increases parameter count slightly (16.5 M), yet the model achieves 92.4% AP50 and 81.6% AP75, albeit at 78 FPS. When all components are combined into the full AMDA framework, the model achieves a dramatic leap in accuracy, 99.1% AP50 and 93.2% AP75, at 58 FPS and 17.9 M parameters, underlining the potency of adaptive multi-scale fusion for robust defect detection.

Overall, these ablation experiments confirm that each added module, FPN, deformable attention, and gating, brings tangible benefits. FPN broadens scale coverage for varying defect sizes, deformable attention pinpoints the most salient feature locations with minimal redundancy, and the gating mechanism adaptively emphasizes the scale levels most relevant for each query. Together, these steps compose our AMDA approach, striking a strong balance of accuracy and efficiency for real-world PCB inspection tasks.

3.5. Discussion and implications

Our results show that the proposed decoder-only Transformer with AMDA achieves a superior balance of accuracy and efficiency compared to both CNN-and Transformer-based detectors. This improvement stems from two design choices: removing the encoder stage to reduce redundant computation, and introducing query-dependent multi-scale gating to adaptively emphasize the most relevant feature levels. As a result, subtle micro-defects are captured at high-resolution scales, while larger assembly flaws are robustly detected with semantically richer features. Compared with anchor-based CNNs such as Faster R-CNN or RetinaNet, our method eliminates the need for hand-crafted priors while improving both accuracy and speed. Relative to Transformer baselines, it avoids the heavy cost of deformable DETR while adding the adaptivity that RT-DETR lacks, thus bridging the trade-off between precision and real-time throughput.

These findings carry clear industrial implications. Reliable detection of both small and large PCB defects at 58 FPS demonstrates that our approach can meet the throughput of modern AOI systems without specialized hardware. The interpretability of AMDA attention maps also provides engineers with transparent defect localization, improving trust and usability. Looking forward, the adaptive decoder-only design suggests a direction for future defect detection models across broader manufacturing domains, such as semiconductor or metal surface inspection. Furthermore, integrating attention-based interpretability into AOI interfaces could enable real-time diagnostic feedback, while extending the framework toward predictive maintenance may enhance long-term process optimization.

4. CONCLUSION

In this work, we proposed a decoder-only Transformer architecture with AMDA for efficient and accurate PCB defect inspection. By combining a CNN-FPN backbone, a region proposal stage, and a lightweight Transformer decoder, our model achieves state-of-the-art detection accuracy while maintaining real-time inference speed on high-resolution PCB images. The adaptive multi-scale fusion in AMDA enables the detector to dynamically select the most relevant feature scale, ensuring robust performance across diverse defect sizes and appearances. Beyond benchmarking performance, our findings demonstrate that encoder-free Transformer designs, when combined with adaptive attention, can redefine efficiency standards for industrial inspection. This provides a pathway toward developing lightweight yet highly accurate defect detectors deployable on existing factory hardware without additional accelerators. The interpretability of attention maps further enhances trust and transparency, supporting practical adoption in real manufacturing environments.

The implications of this research extend beyond PCB inspection. The proposed architecture offers a generalizable framework for other high-resolution, multi-scale defect detection tasks, such as semiconductor wafer inspection, solar cell fault analysis, or metal surface quality control. By coupling efficiency with adaptability, our approach contributes to advancing intelligent AOI systems that meet both throughput and reliability demands. For future work, we aim to extend the model's capacity to handle unseen defect types through self-supervised pretraining or domain adaptation. We also envision integrating our method into hierarchical defect inspection pipelines for extremely large PCBs. On the application side, collaboration with industry partners will allow deployment in real AOI workflows, enabling practical evaluation, user-centered interfaces with confidence visualization, and fail-safe mechanisms for critical errors. These extensions will help translate our research into scalable, industry-ready solutions, ultimately benefiting the broader field of automated quality control.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Chi Kien Ha	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Hoanh Nguyen		✓				✓	✓	✓	✓	✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] Z. Huang, J. Duan, Y. Xie, and Y. Liu, "UDNet: Unified Deep Network based on Transformer and Multi-stage Fusion for brain tumor classification from undersampled MRI," *Neurocomputing*, vol. 619, Feb. 2025, doi: 10.1016/j.neucom.2024.129109.

[2] H. Nguyen, T. A. Nguyen, and N. D. Toan, "Optimizing feature extraction and fusion for high-resolution defect detection in solar cells," *Intelligent Systems with Applications*, vol. 24, pp. 1-12, Dec. 2024, doi: 10.1016/j.iswa.2024.200443.

[3] Y. Zheng *et al.*, "Incremental printing product defect detection based on contextual information," *Signal, Image and Video Processing*, vol. 19, no. 4, p. 317, Apr. 2025, doi: 10.1007/s11760-025-03948-5.

[4] Z. Wang, L. Wang, and S. Zheng, "Improved Plate Defect Detection Algorithm Based on YOLOv5," in *International Conference on Internet of Things as a Service*, 2025, pp. 371–384.

[5] D. Li, F. Bai, S. Li, B. Liu, and L. He, "EL-PCBNet: An efficient and lightweight network for PCB defect detection," *Measurement (Lond)*, vol. 253, 2025, doi: 10.1016/j.measurement.2025.117719.

[6] S. Tang, F. He, X. Huang, and J. Yang, "Online PCB Defect Detector On A New PCB Defect Dataset," *arXiv preprint*, 2019, doi: 10.48550/arXiv.1902.06197.

[7] X. Chen, Y. Wu, X. He, and W. Ming, "A Comprehensive Review of Deep Learning-Based PCB Defect Detection," *IEEE Access*, vol. 11, pp. 139017–139038, 2023, doi: 10.1109/ACCESS.2023.3339561.

[8] S. Khalilian, Y. Hallaj, A. Balouchestani, H. Karshenas, and A. Mohammadi, "PCB Defect Detection Using Denoising Convolutional Autoencoders," in *2020 International Conference on Machine Vision and Image Processing (MVIP)*, Feb. 2020, pp. 1–5, doi: 10.1109/MVIP49855.2020.9187485.

[9] R. Ding, L. Dai, G. Li, and H. Liu, "TDD-net: a tiny defect detection network for printed circuit boards," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 110–116, Jun. 2019, doi: 10.1049/trit.2019.0019.

[10] H. Guo, H. Zhao, Y. Zhao, and W. Liu, "PCB defect detection algorithm based on deep learning," *Optik*, vol. 315, p. 172036, Nov. 2024, doi: 10.1016/j.ijleo.2024.172036.

[11] S. Deng *et al.*, "EHIR: Energy-based Hierarchical Iterative Image Registration for Accurate PCB Defect Detection," *Pattern Recognition Letters*, vol. 185, pp. 38–44, Sep. 2024, doi: 10.1016/j.patrec.2024.06.027.

[12] X. Hu, D. Kong, X. Liu, J. Zhang, and D. Zhang, "Printed Circuit Board (PCB) Surface Micro Defect Detection Model Based on Residual Network with Novel Attention Mechanism," *Computers, Materials & Continua*, vol. 78, no. 1, pp. 915–933, 2024, doi: 10.32604/cmc.2023.046376.

[13] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, pp. 213–229, doi: 10.1007/978-3-030-58452-8_13.

[14] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable Transformers for End-to-End Object Detection," *arXiv preprint*, 2021, doi: 10.48550/arXiv.2010.04159.

[15] B. Roh, J. W. Shin, W. Shin, and S. KiM, "Sparse DETR: Efficient end-to-end object detection with learnable sparsity," *arXiv preprint*, 2021, doi: 10.48550/arXiv.2111.14330.

[16] D. Meng *et al.*, "Conditional DETR for Fast Training Convergence," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 3631–3640, doi: 10.1109/ICCV48922.2021.00363.

[17] Y. Wang, X. Zhang, T. Yang, and J. Sun, "Anchor DETR: Query Design for Transformer-Based Object Detection," *arXiv*

preprint, 2022, doi: 10.48550/arXiv.2109.07107.
[18] L. He and S. Todorovic, "DESTR: Object Detection with Split Transformer," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 9367–9376, doi: 10.1109/CVPR52688.2022.00916.
[19] Y. Zhao *et al.*, "DETRs Beat YOLOs on Real-time Object Detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2024, doi: 10.1109/CVPR52733.2024.01605.
[20] H. Zhang *et al.*, "DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection," *arXiv preprint*, 2022, doi: 10.48550/arXiv.2203.03605.
[21] Z. Sun, S. Cao, Y. Yang, and K. Kitani, "Rethinking Transformer-based Set Prediction for Object Detection," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 3591–3600, doi: 10.1109/ICCV48922.2021.00359.
[22] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 9626–9635, doi: 10.1109/ICCV.2019.00972.
[23] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 936–944, doi: 10.1109/CVPR.2017.106.
[24] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386–397, Feb. 2020, doi: 10.1109/TPAMI.2018.2844175.
[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.
[26] 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, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.
[27] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.
[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
[29] H. Law and J. Deng, "CornerNet: Detecting Objects as Paired Keypoints," *International Journal of Computer Vision*, vol. 128, no. 3, pp. 642–656, Mar. 2020, doi: 10.1007/s11263-019-01204-1.
[30] S. Sutikno, A. Sugiharto, R. Kusumaningrum, and H. A. Wibawa, "Improved car detection performance on highways based on YOLOv8," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 5, pp. 3526–3533, Oct. 2024, doi: 10.11591/eei.v13i5.8031.

BIOGRAPHIES OF AUTHORS

Chi Kien Ha received M.Sc. degrees from the Ho Chi Minh City University of Technology, Vietnam, in 2013. Currently, he is a lecturer with the Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Vietnam. His main research interests include automated guided vehicles (AGV) and automatic systems. He can be contacted at email: hachikien@iuh.edu.vn.

Hoanh Nguyen received a Bachelor's and Master's degree in Automation at Ho Chi Minh City University of Technology in 2010 and 2015, respectively. Currently, he is pursuing a doctoral program at Ho Chi Minh City University of Technology. He is also a lecturer at the Department of Electrical Engineering at the Industrial University of Ho Chi Minh City, Vietnam. His research focuses on deep learning, intelligent control, computer vision, and smart transportation. He can be contacted at email: nguyenhoanh@iuh.edu.vn.