

A fast defect detection method for PCBA based on YOLOv7

Shugang Liu^{1,3}, Jialong Chen^{1*}, Qiangguo Yu², Jie Zhan^{1,3}, and Linan Duan¹

¹ School of Physics and Electronic Science, Hunan University of Science and Technology, Xiangtan 411201, China

² School of Electronic Information, Huzhou College, Huzhou 313000, China

³ Key Laboratory of Intelligent Sensors and Advanced Sensing Materials of Hunan Province, Hunan University of Science and Technology, Xiangtan 411201, China

*Corresponding author: Jialong Chen

E-mail: 2395904554@qq.com

*Received March 14, 2024; revised June 11, 2024; accepted July 15, 2024;
published August 31, 2024*

Abstract

To enhance the quality of defect detection for Printed Circuit Board Assembly (PCBA) during electronic product manufacturing, this study primarily focuses on optimizing the YOLOv7-based method for PCBA defect detection. In this method, the Mish, a smoother function, replaces the Leaky ReLU activation function of YOLOv7, effectively expanding the network's information processing capabilities. Concurrently, a Squeeze-and-Excitation attention mechanism (SEAM) has been integrated into the head of the model, significantly augmenting the precision of small target defect detection. Additionally, considering angular loss, compared to the CIoU loss function in YOLOv7, the SIoU loss function in the paper enhances robustness and training speed and optimizes inference accuracy. In terms of data preprocessing, this study has devised a brightness adjustment data enhancement technique based on split-filtering to enrich the dataset while minimizing the impact of noise and lighting on images. The experimental results under identical training conditions demonstrate that our model exhibits a 9.9% increase in mAP value and an FPS increase to 164 compared to the YOLOv7. These indicate that the method proposed has a superior performance in PCBA defect detection and has a specific application value.

Keywords: Object detection, Deep learning, YOLOv7, Defect Detection

1. Introduction

Common issues such as missed soldering and component misplacement are widespread during the printed circuit board assembly. Currently, the detection of PCBA defects primarily relies on manual visual inspection and electrical testing. Although widely employed, manual inspection is inefficient and often experiences a decline in accuracy over time. Electrical testing, while capable of quickly and accurately assessing the conductivity and insulation of printed circuit boards through bed-of-nails testers, is constrained by probe spacing and the risk of potential damage or contamination. Moreover, for manufacturers producing a variety of circuit boards, the frequent need to change test fixtures can significantly escalate production costs. Therefore, traditional manual and electrical testing methods are suboptimal for PCBA defect detection. In recent years, many companies have adopted automated defect detection techniques based on deep learning to achieve flawless PCBA production. In particular, advanced target detection algorithms centered around Convolutional Neural Networks (CNN) have shown exceptional performance in defect detection tasks [1] [2]. For instance, two-stage detectors such as the R-CNN series [3], including Fast R-CNN [4], Faster R-CNN [5], and Mask R-CNN [6], as well as single-stage detectors like SSD [7][8], RetinaNet [9], EfficientDet [10], and the YOLO series [11][12][13][14][15][16][17], have struck a good balance between processing speed and accuracy. However, these algorithms have not effectively addressed the dual requirements of high precision and speed essential for PCBA quality inspection.

This study addresses the issues of missing and reversed polarity in PCBA components, concentrating on five common defect types as illustrated in Fig. 1: missed capacitors, resistors, triodes, electrolytic capacitors, and reverse polarity of electrolytic capacitors. Leveraging the high precision and rapid detection advantages of the single-stage detector YOLOv7, we have enhanced the foundational YOLOv7 network model to achieve swift and efficient detection of PCBA defects. The primary contributions of this research include:

- (1) The Mish activation function enhances the model's generalization ability, thereby improving the quality of training.
- (2) The SEAM strengthens the recognition ability for tiny targets and increases the detection precision.
- (3) The substitute SIOU for CIOU of YOLOv7 can efficiently measure model performance and deal with model uncertainty, enhancing the robustness of the model.
- (4) The design of a brightness adjustment data enhancement algorithm based on split filtering enriches the training dataset and reduces the impact of uneven lighting during the shooting process, enhancing the model's resistance to interference in practical applications.

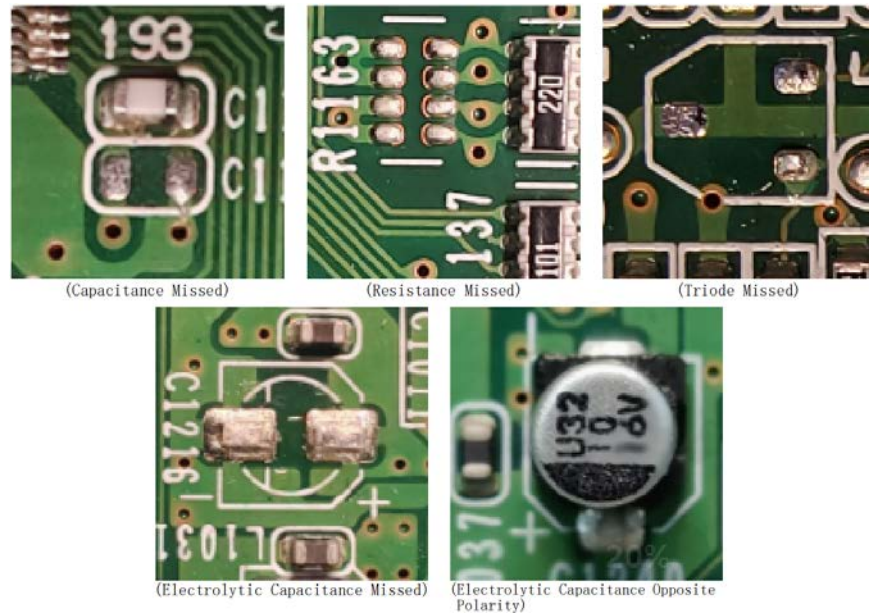


Fig. 1. Five common types of PCBA defects

2. Related work

Defect detection in PCBA is an indispensable part of industrial production and has garnered widespread attention domestically and internationally. Wang and others devised and implemented an automated optical inspection system leveraging machine vision to inspect holes in printed circuit boards (PCBs) [18]. However, the performance of optical inspection systems is significantly influenced by variations in lighting conditions. Reflective surfaces and shadows on the PCBs can interfere with the machine vision system, potentially leading to false alarms or missed defect detections. Nabil El Belghiti Alaoui and colleagues proposed using electromagnetic near-field sensors for PCBA detection [19]. By measuring the magnetic field above the PCBA, this method provides non-contact information about the current distribution and is not limited by surface contact with the PCBA. However, this method is vulnerable to electromagnetic interference in the environment, which can adversely affect the accuracy and stability of the sensor. Jeon Mingu and others proposed a method for detecting PCBA defects using thermal image comparison and deep learning [20]. They utilized thermal image analysis employing a structural similarity index map as a rule-based object detection method, alongside using CNN, CNN feature regions, and autoencoders for analysis. The primary limitation of thermal imaging is its insufficient resolution for detecting minute defects. Furthermore, the implementation of deep learning and thermal image analysis techniques generally necessitates substantial computational resources, which can pose challenges for real-time detection.

In research utilizing the YOLO algorithm, Xu Siang et al. applied the YOLOv5 algorithm to defect detection in bare PCBs. They enhanced the feature fusion pathways to minimize information loss, effectively identifying defects on bare boards [21]. However, this study did not detect defects in the assembled devices on the PCB. Yih-Lon Lin and colleagues utilized the YOLO algorithm to detect capacitors mounted on PCBs, efficiently identifying nine capacitors assembled on the boards [22]. There is a broader range of assembly devices on PCBs beyond just capacitors, underscoring the necessity for further research to detect various defects across different components. They designed a coordinated attention detection head and

introduced feedback connections that significantly enhanced feature recognition, achieving excellent performance in identifying surface-mounted device components. It is important to note that the YOLOv7 model may face challenges such as false or missed detections, mainly when dealing with densely distributed surface mount devices of varying sizes, complex backgrounds, or blurred device edges. So, data enhancement algorithms are needed to mitigate the effect of image quality on detection.

In pursuit of efficient PCBA defect identification and simplified network deployment, this paper initially focuses on optimizing network structure and data enhancement algorithms. Specifically tailored for PCBA defect detection, the YOLOv7 model undergoes optimization in this study.

3. Methodology

3.1 YOLOv7 overview

The architecture of the YOLOv7 model is delineated into three key segments: Input, Backbone, and Head. Initially, images undergo data augmentation and a sequence of preprocessing operations during the initial processing stage before entering the backbone network. The backbone network performs feature extraction on the input images. Subsequently, the extracted features are merged in the head module, resulting in three feature maps of different sizes for prediction and output [23].

Through algorithm enhancement and composite model scaling strategy, YOLOv7 has been meticulously optimized for edge devices, resulting in multi-scale creation. This strategic approach is tailored to accommodate various inference speeds and accuracy requirements. These models are YOLOv7, YOLOv7-e6, YOLOv7-tiny, and others, offering adaptable deployment options and efficient operation across various devices. Specifically, YOLOv7-tiny, the most compact iteration within the series, is finely tuned for edge GPUs and adopts Leaky ReLU as its activation function. Its modest memory footprint (12.00M) and swift processing capabilities make it well-suited for PCBA defect detection in industrial manufacturing settings. However, while the original model's streamlined complexity is advantageous, it may not fully meet the rigorous efficiency and accuracy demands inherent in defect detection tasks, particularly those involving multiple targets. Hence, targeted enhancements to YOLOv7-tiny are imperative to ensure both the effectiveness and precision of detection outcomes.

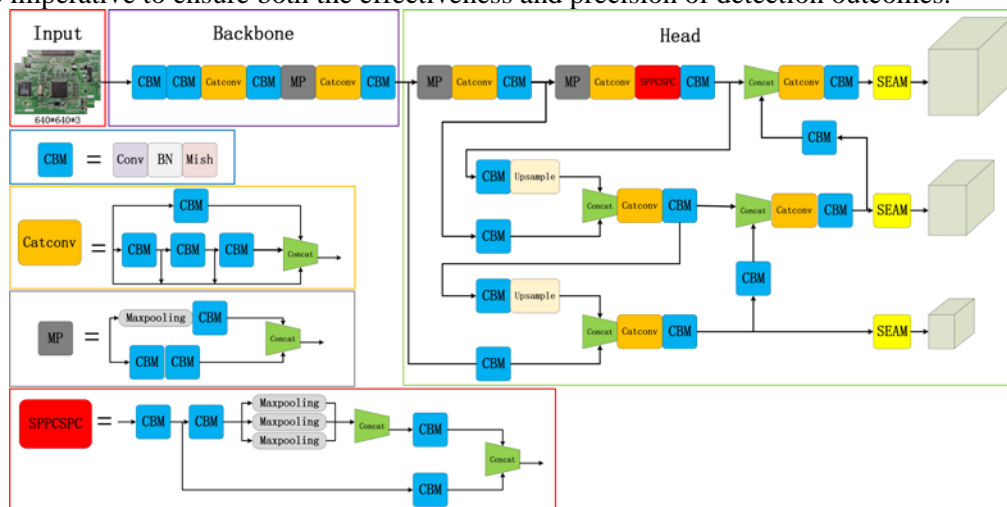


Fig. 2. Improved YOLOv7 network structure

3.2 Network architecture

Fig. 2 illustrates the structure of the improved YOLOv7 proposed in this paper. We designed these modifications to optimize the original YOLOv7 architecture for the unique requirements of PCBA defect detection. The enhanced YOLOv7 network structure mainly comprises the following modules: CBM, MP, SPPCSPC, and SEAM.

CBM Module: This foundational convolution module consists of a convolution layer, a batch normalization layer, and the Mish activation function. In this enhancement, the Leaky ReLU has been replaced with the Mish activation function to improve the model's capacity to handle nonlinear problems. **MP Module:** Serving as a downsampling module, the MP module considers not only the maximum value information of local small areas but also the comprehensive information of the whole, aiming to capture features more effectively. **SPPCSPC Module:** This improved spatial pyramid pooling framework fuses SPP with the CSP framework. It's designed to bolster the capability to detect densely packed targets in PCBA, particularly for recognizing small objects. **SEAM Module:** The SEAM module was introduced in the head part to enhance target detection, especially for recognizing dense targets on the PCBA board.

Furthermore, we substitute the initial model's CIoU with SIoU to increase training speed and accuracy during inference. These comprehensive enhancements improve the model's performance and provide a more precise and efficient solution for PCBA defect detection.

3.3 Mish activation function

PCBA defect detection requires the identification of a diverse array of intricate defect patterns. The Mish activation function, known for capturing complex nonlinear relationships, enhances the model's expressive capability by adaptively adjusting the activation values. Using the Mish activation function can significantly enhance the overall performance of PCBA defect detection models, thereby better aligning these models with the stringent demands of industrial applications.

The Mish activation function is distinguished by its unboundedness, smoothness, and non-monotonic nature, which effectively circumvent the saturation issues inherent in traditional capped activation functions, unlike the rigid zero boundaries characteristic of Leaky ReLU activation functions [24]. Mish's smooth activation facilitates deeper information propagation within neural networks, enhancing accuracy and generalization capabilities. Equation (1) presents the mathematical definition of the Mish function, and **Fig. 3** provides its graphical representation.

$$\text{Mish}=x * \tanh(\ln(1+\exp(x))) \quad (1)$$

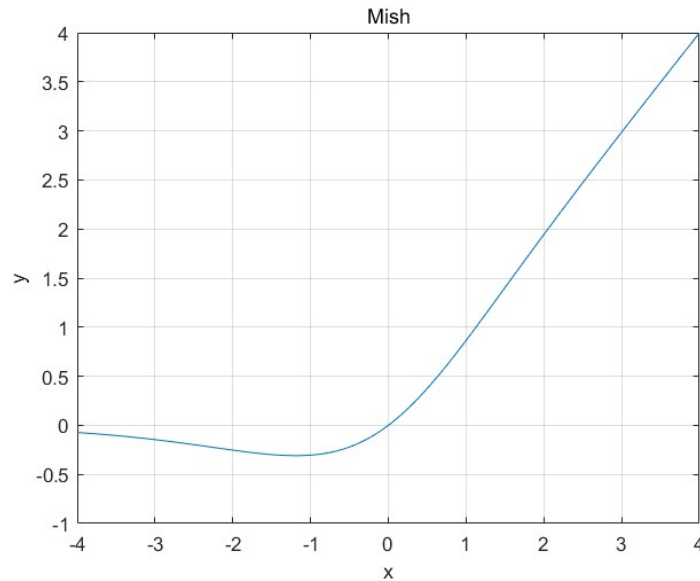


Fig. 3. Mish function graph

3.4 Squeeze-and-Excitation attention mechanism

We have many minor PCBA defects to detect, resulting in low detection accuracy. To improve the detection accuracy without increasing the computational overhead, we use SEAM to improve the model detection performance. The SEAM enhances the feature representation capabilities of the model, aiming to improve the efficiency of architectures such as CNN in learning valuable features. The core idea of SEAM is to enable the model to autonomously determine each channel's significance and then reweight these channels, thereby allocating different attention weights among various channels. This mechanism facilitates a focus on feature channels containing crucial information, enhancing the model's performance.

Fig. 4 illustrates the schematic of the SEAM. In this diagram, the incoming feature map X_1 first undergoes a transformation operation F_{tr} to generate the feature map U . Subsequently, the map U undergoes global average pooling F_{sq} and is then processed by F_{ex} through two fully connected layers, imparting distinct weight information to different channels. Finally, in the F_{scale} step, the weight information obtained from the previous step is used to reweight the feature map U , resulting in the desired feature map X_2 , which retains the exact dimensions of the feature map U .

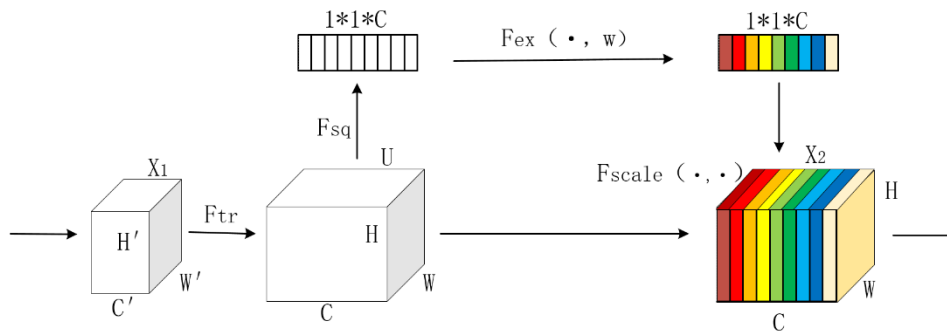


Fig. 4. Squeeze and Excitation Attention Mechanism

3.5 SIOU loss function

The efficacy of target detection algorithms is intricately tied to the formulation of the loss function. Conventional loss functions for target detection primarily integrate bounding box regression metrics, encompassing distance, aspect ratio, and overlap ratio (e.g., CIOU, IOU). However, these methods often overlook the disparity in direction between the forecasted and actual boxes, a flaw that can lead to slower model convergence and reduced efficiency. This issue is particularly pronounced in the PCBA domain, where the multitude of components often concentrated in a small area can significantly degrade model performance during training due to the imprecision of predicted box locations. To address this challenge, our study employs the SIOU instead of the traditional CIOU. The SIOU integrates four cost components: Angle cost, Distance cost, Shape cost, and IoU cost. The function is defined as follows equation:

$$L_{\text{SIOU}} = 1 - I_{\text{IOU}} + \frac{\Delta + \Omega}{2} \quad (2)$$

I_{IOU} denotes IoU cost, Δ denotes distance cost, and Ω denotes shape cost. SIOU incorporates an angle factor in calculating distance loss and shape loss. This incorporation enriches the expression of the loss function but also reduces its degrees of freedom. These enhancements contribute to a more stable convergence process and improve regression accuracy [25].

3.6 Brightness adjustment data enhancement algorithm based on split filtering

Data augmentation is a widely utilized technique in deep learning to generate new training samples via various transformations applied to the original data. Its primary purpose is to mitigate overfitting during training and enhance the model's adaptability and robustness to diverse scenarios. By introducing randomness and variability, data augmentation allows models to generalize more effectively to unseen data, improving their performance in practical applications.

In the context of PCBA defect detection, where the target distribution is dense and numerous tiny components are often concentrated in the same area, detection becomes particularly complex. Additionally, since cameras capture all data, the reflective nature of PCB exacerbates issues such as uneven brightness and noise in the images. This study proposes a Brightness Adjustment based on Split Filtering (BASF-DAA) data augmentation algorithm to address these specific challenges.

The Split Filtering algorithm primarily comprises the following steps: Image Splitting: Initially, the original image is segmented into four equal-sized parts, aiming to reduce the number of targets on a single image and lower the processing complexity. Bilateral Filtering for Noise Reduction: The bilateral filtering algorithm is then applied to remove image noise [26]. Bilateral filtering smooths the image while preserving essential edge information, eliminating noise, and maintaining image details by considering spatial proximity and pixel value similarity. Brightness Adjustment Processing: The four split image sections undergo brightness adjustments to address issues caused by uneven lighting and reflections. The implementation principle of bilateral filtering can be summarized in the following steps:

- (1) Parameter Definition: Define the spatial and intensity domain kernels. The spatial domain kernel determines the weight of the spatial distance between pixels, while the intensity domain kernel determines the weight of the similarity between pixel values.
- (2) Weight Calculation: For each pixel, bilateral filtering calculates weights in both the spatial and intensity domains, reflecting the spatial distance and value similarity between pixels.

- (3) Filtering Process: Using the calculated weights, a weighted average is applied to the neighboring pixels of each target pixel to obtain the filtered pixel value.

The formula for bilateral filtering can be expressed as (3):

$$I_{\text{filtered}}(x, y) = \frac{1}{W_p(x, y)} \sum_{(i, j) \in \Omega} I(x+i, y+j) * F_{\text{spatial}}(i, j) * F_{\text{range}}(I(x, y), I(x+i, y+j)) \quad (3)$$

$I_{\text{filtered}}(x, y)$ represents the pixel value after filtering, and $I(x, y)$ represents the pixel value in the original image. $W_p(x, y)$ is the normalized weight used for filtering result normalization. Ω denotes the neighborhood window of the filter, typically a fixed-size window. $F_{\text{spatial}}(i, j)$ is the spatial domain kernel function used to calculate the weights of spatial distances between the pixels, while $F_{\text{range}}(I(x, y), I(x+i, y+j))$ is the pixel-value domain kernel function used to calculate the weights of similarity between the pixel values. The detailed process is illustrated in Fig. 5:

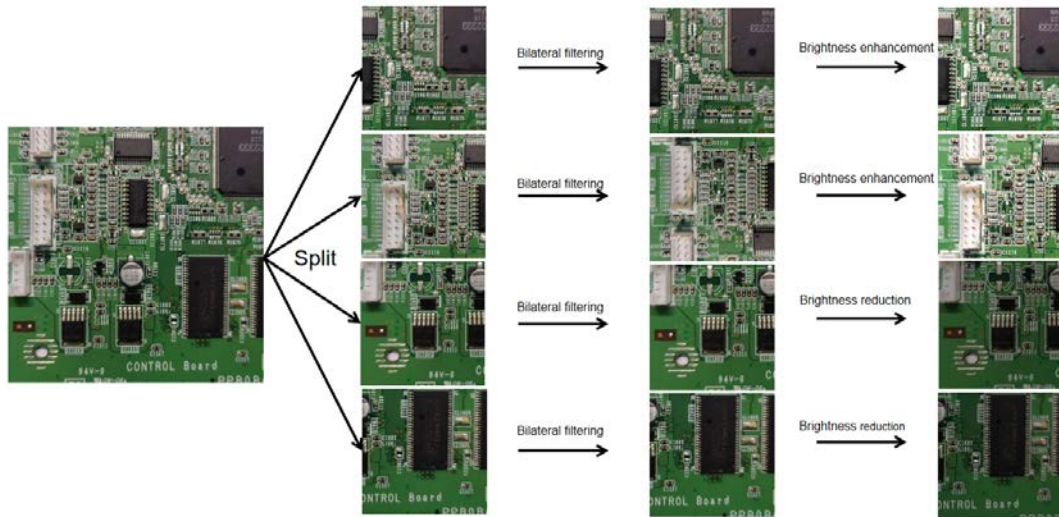


Fig. 5. Flowchart of brightness adjustment data enhancement algorithm based on split filtering

By implementing the data augmentation algorithm, this study not only achieved diversification and enrichment of the dataset but also significantly reduced the negative impact of noise and uneven illumination on model training. This strategy effectively enhanced the model's resistance to interference and improved the network's accuracy and reliability in PCBA defect detection.

4. Experiment and Analysis

4.1 Training conditions

To validate the effectiveness and advancement of this paper's improvement for PCBA defect detection, the experimental setup includes an Intel(R) Core(TM) i7-12700 CPU and a GeForce RTX 3070 GPU. Python is the programming language utilized, and the deep learning framework used is Pytorch 1.10.2. The datasets utilized in this experiment are all independently produced and processed using a data enhancement algorithm, totaling 5852 sheets. Among these, 4780 sheets comprise the training set, 560 sheets are allocated for the validation set, and the test set consists of 512 sheets. The dataset encompasses five categories

of defects, with each image containing zero to multiple defects. The input size of the images is set to 640*640 pixels, and the batch size is configured as 8. Finally, the number of epochs is set to 300.

4.2 Assessment of Indicators

To assess the performance of the trained model from a professional and objective perspective, this paper utilizes commonly employed evaluation metrics, which include precision (P), recall (R), average precision (AP), mean average precision (mAP), floating point operations (FLOPs), frames per second (FPS), and model size.

When the detection frame's intersection and concurrency ratio IoU to the actual frame is more significant than 0.5, the sample is recorded as a true positive (TP); otherwise, it is recorded as a false positive (FP). A false negative (FN) denotes a positive sample misclassified as unfavorable. Precision (P) is the proportion of actual positive samples to all detected targets of that type, and recall (R) is the proportion of accurate positive samples to all actual positive samples. P and R are shown in (4) and (5):

$$\text{precision}(P) = \frac{TP}{TP + FP} \quad (4)$$

$$\text{recall}(R) = \frac{TP}{TP + FN} \quad (5)$$

A two-dimensional P-R curve is drawn with precision P represented on the vertical axis and recall R on the horizontal axis. The area of the region enclosed by the P-R curve and the coordinate axis is the value of AP. Its formula is as follows:

$$AP_i = \int_0^1 P_i(R_i) dR_i \quad (6)$$

After figuring out the AP values for each category, the mAP value is obtained by averaging all AP values. The mAP formula is calculated as follows:

$$\text{mAP} = \frac{1}{n} \sum_{i=1}^n AP_i \quad (7)$$

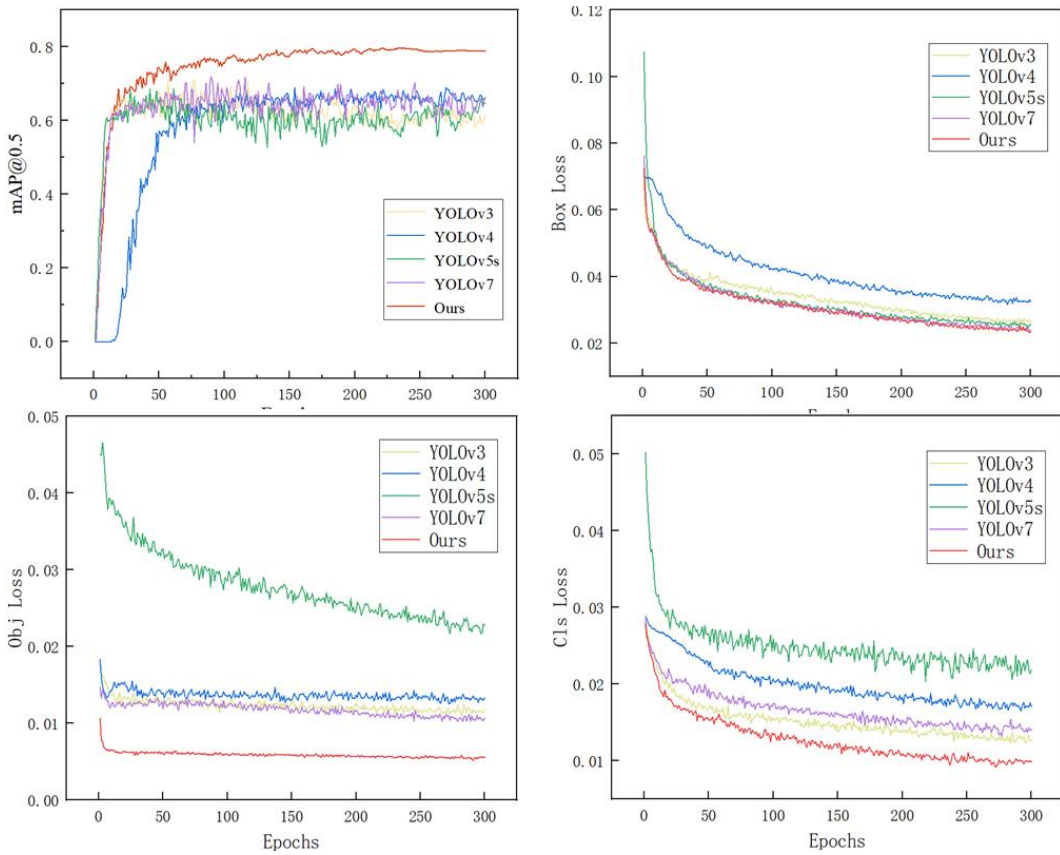
The AP value can be utilized to comprehensively assess both P and R, measuring detection performance. The mAP, the average of all AP values, evaluates the algorithm's overall detection performance. In the context of the YOLO algorithm, the values of these two metrics reflect the detection accuracy of the model, with higher values indicating better performance.

4.3 Performance analysis

While ensuring that the computer configuration and initial training conditions are the same, this paper conducts experiments comparing the enhanced model with other classical YOLO models. The efficacy of the enhanced network model is demonstrated through the results presented in [Table 1](#). It is evident that the enhanced YOLOv7 model not only outperforms other classic network models in terms of mAP and FPS but also has a smaller model size. Additionally, when fed images of the same size, the number of floating-point operations and the size of parameters are smaller than those of other models, demonstrating the superiority of our algorithm. [Fig. 6](#) illustrates the variation of parameter values during the training process. The mAP values gradually increase while the loss function values steadily decrease. It can be observed that the model, enhanced in this study, not only achieves an optimal loss function curve but also attains the highest mAP values.

Table 1. The performance between the method proposed and the previous YOLO model

Method	Model	mAP	FPS	FLOPs	Params
YOLOv3	123.5M	0.701	36.8	154.6	61.5M
YOLOv4	100M	0.685	42	119	52.5M
YOLOv5s	14.4M	0.672	102	16.3	7.06M
YOLOv7	74.9M	0.709	47	103.2	36.5M
Ours	12.4M	0.808	164	13.1	6.06M

**Fig. 6.** Comparison of training parameters

We use the improved model to detect defects on PCBA, as illustrated in **Fig. 7**. The results demonstrate that Our method identifies all the defects accurately and operates without any identification errors. These findings further validate the feasibility of the proposed method.

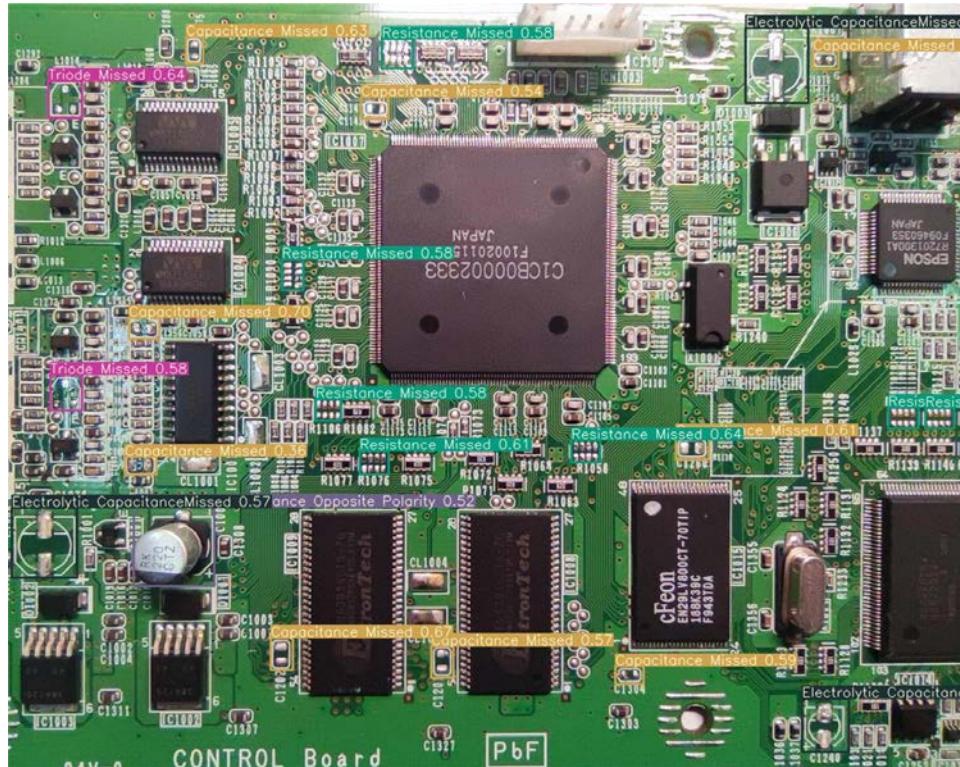


Fig. 7. PCBA test results

4.4 Ablation experiments

In the present manuscript, we propose several improvement methods. To demonstrate the advantages and effects of each method on the model, we conducted the following experiments, with the results presented in [Table 2](#).

Table 2. Role of different approaches

Method	Model	mAP	FPS	FLOPs	Params
YOLOv7	74.9M	0.709	47	103.2	36.5M
YOLOv7+BASF-DAA	74.9M	0.771	53	103.2	36.5M
YOLOv7+SEAM	12.4M	0.746	154	13.1	6.06M
Ours	12.4M	0.808	164	13.1	6.06M

After processing with the data enhancement algorithm, the number of photographs in the dataset increased. At the same time, the interfering information decreased, resulting in an increase in the mean mAP to 0.771, which is 8.75% higher than without the data enhancement algorithm. Adding the data enhancement algorithm improves recognition accuracy without increasing computational demands or model size.

After integrating the SEAM into the model head and applying layer pruning for simplification, experimental results indicate a substantial reduction in model size and

parameters. This optimization yields a nearly 2.5-fold increase in the model's FPS and reduces the FLOPs to as low as 12.7. These findings demonstrate that incorporating the SEAM and implementing layer pruning can significantly enhance FPS without compromising performance.

Fig. 8 illustrates the performance improvement across each category following model integration. The modified model demonstrates a notable enhancement in mAP values. The analysis and comparison of each category reveal that the impact of algorithm optimization varies among different classes. Notably, there has been a significant improvement in the categories of Capacitance Missed, Resistance Missed, Electrolytic Capacitance Opposite Polarity, and Triode Missed. Additionally, the recognition capability for Electrolytic Capacitance Missed remains excellent.

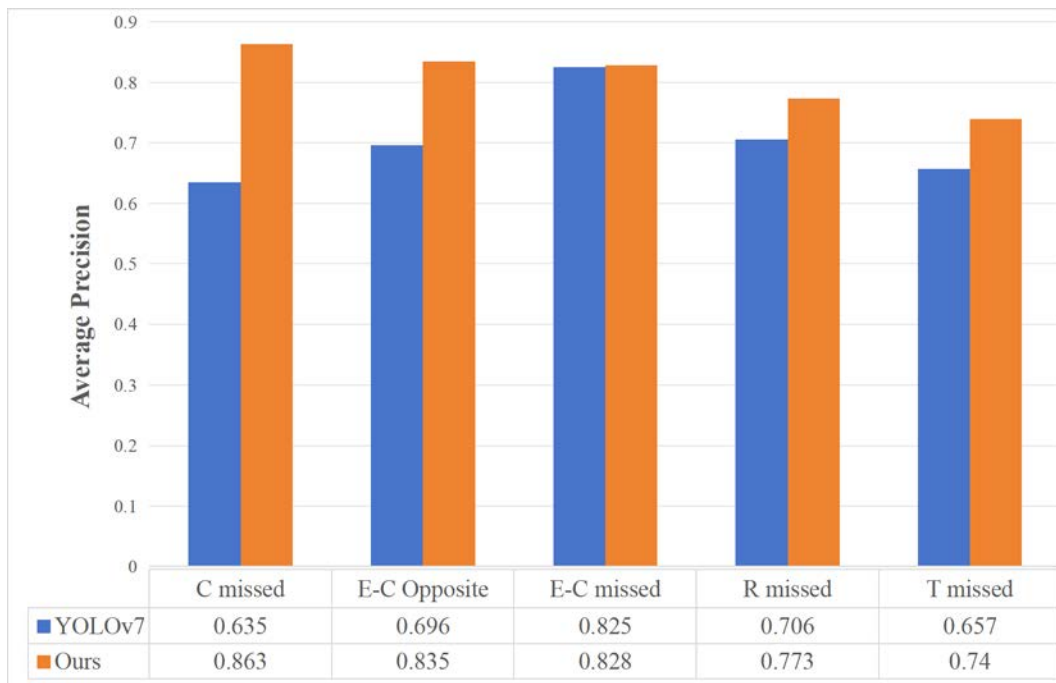


Fig. 8. Before and after improvement of AP values for each category

5. Conclusion

To address detection challenges in PCBA defects, including missed and false detections, we propose a novel defect detection algorithm for PCBA based on YOLOv7. We introduce the Mish activation function, which improves information propagation through the network and enhances detection performance. Additionally, the SEAM has been integrated into the model's head, enabling a greater focus on small targets and improving recognition. The SIOU is employed to replace the CIOU in YOLOv7, thereby improving the efficiency of performance measurement, addressing uncertainty, and enhancing the model's robustness. A brightness adjustment data enhancement algorithm based on split filtering is proposed for data processing to optimize the dataset and improve the quality of training. Experimental results demonstrate that this method achieves both higher FPS and mAP. These findings indicate that our model can maintain high recognition accuracy despite low computational requirements. Compared to other models discussed in this paper, various evaluation metrics show that our method

performs better and holds significant reference value for PCBA defect detection in the industry. In the future, we will continue refining our algorithm to detect a broader range of defects applicable in a wider variety of scenarios.

Acknowledgement

This work was supported in part by the National Engineering Research Center for Mobile Private Networks, Beijing Jiaotong University (No. BJTU20221102), and in part by the Xiangtan Key Science and Technology Achievement Transformation Project (No. GX-ZD202210012).

References

- [1] Ling Q, Isa N. A. M., "Printed Circuit Board Defect Detection Methods Based on Image Processing, Machine Learning and Deep Learning: A Survey," *IEEE Access*, vol.11, pp.15921-15944, 2023. [Article \(CrossRef Link\)](#)
- [2] W. Fang, W. Yi, L. Pang, S. Hou, "A Method of License Plate Location and Character Recognition based on CNN," *KSII Transactions on Internet and Information Systems*, vol.14, no.8, pp.3488-3500, 2020. [Article \(CrossRef Link\)](#)
- [3] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the Objectness of Image Windows," *IEEE transactions on pattern analysis and machine intelligence*, vol.34, no.11, pp.2189-2202, 2012. [Article \(CrossRef Link\)](#)
- [4] R. Girshick, "Fast R-CNN," in *Proc. of the 2015 IEEE international conference on computer vision*, pp.1440-1448, Dec. 2015. [Article \(CrossRef Link\)](#)
- [5] 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. [Article \(CrossRef Link\)](#)
- [6] He K., Gkioxari G., Dollár P., Girshick R., "Mask R-CNN," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pp.2961-2969, 2017. [Article \(CrossRef Link\)](#)
- [7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *Proc. of Computer Vision – ECCV 2016: 14th European Conference*, pp.21-37, 2016. [Article \(CrossRef Link\)](#)
- [8] H. Sun, W. Geng, J. Shen, N. Liu, D. Liang, H. Zhou, "Deeper SSD: Simultaneous Up-sampling and Down-sampling for Drone Detection," *KSII Transactions on Internet and Information Systems*, vol.14, no.12, pp.4795-4815, 2020. [Article \(CrossRef Link\)](#)
- [9] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," in *Proc. of IEEE international conference on computer vision (ICCV)*, pp.2980-2988, 2017. [Article \(CrossRef Link\)](#)
- [10] M. Tan, R. Pang, Q. V. Le, "EfficientDet: Scalable and Efficient Object Detection," in *Proc. of IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp.10781-10790, 2020. [Article \(CrossRef Link\)](#)
- [11] P. Jiang et al., "A Review of Yolo Algorithm Developments," *Procedia Computer Science*, vol.199, pp.1066-1073, 2022. [Article \(CrossRef Link\)](#)
- [12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.779-788, 2016. [Article \(CrossRef Link\)](#)
- [13] J. Redmon, A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv:1804.02767*, 2018. [Article \(CrossRef Link\)](#)
- [14] S. Agarwal, J. O. D. Terrail, F. Jurie, "Recent Advances in Object Detection in the Age of Deep Convolutional Neural Networks," *arXiv:1809.03193*, 2018. [Article \(CrossRef Link\)](#)

- [15] E. Kim, J. Lee, H. Jo, K. Na, E. Moon, G. Gweon, B. Yoo, Y. Kyung, "SHOMY: Detection of Small Hazardous Objects using the You Only Look Once Algorithm," *KSII Transactions on Internet and Information Systems*, vol.16, no.8, pp.2688-2703, 2022. [Article \(CrossRef Link\)](#)
- [16] S. Liu et al., "CEAM-YOLOv7: Improved YOLOv7 Based on Channel Expansion and Attention Mechanism for Driver Distraction Behavior Detection," *IEEE Access*, vol.10, pp.129116-129124, Dec. 2022. [Article \(CrossRef Link\)](#)
- [17] F. J. P. Montalbo, "A Computer-Aided Diagnosis of Brain Tumors Using a Fine-Tuned YOLO-based Model with Transfer Learning," *KSII Transactions on Internet and Information Systems*, vol.14, no.12, pp.4816-4834, 2020. [Article \(CrossRef Link\)](#)
- [18] W.-C. Wang et al., "A Machine Vision Based Automatic Optical Inspection System for Measuring Drilling Quality of Printed Circuit Boards," *IEEE Access*, vol.5, pp.10817-10833, Nov. 2016. [Article \(CrossRef Link\)](#)
- [19] N. E. B. Alaoui et al., "New defect detection approach using near electromagnetic field probing of high density PCBAs," *Microelectronics Reliability*, vol.88-90, pp.288-293, Sep. 2018. [Article \(CrossRef Link\)](#)
- [20] M. Jeon, S. Yoo, and S.-W. Kim, "A Contactless PCBA Defect Detection Method: Convolutional Neural Networks With Thermographic Images," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol.12, no.3, pp.489-501, 2022. [Article \(CrossRef Link\)](#)
- [21] S. Xu, Y. Li et al., "Bare PCB Defect Detection Based on Improved YOLOv5 Algorithm," *Packaging Engineering*, vol.43, no.15, pp.33-41, 2022. [Article \(CrossRef Link\)](#)
- [22] Y.-L. Lin, Y.-M. Chiang, and H.-C. Hsu, "Capacitor Detection in PCB Using YOLO Algorithm," in *Proc. of 2018 International Conference on System Science and Engineering*, pp.1-4, 2018. [Article \(CrossRef Link\)](#)
- [23] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," in *Proc. of 2023 IEEE/CVF conference on computer vision and pattern recognition*, pp.7464-7475, 2023. [Article \(CrossRef Link\)](#)
- [24] D. Mishra, "Mish: A Self Regularized Non-Monotonic Activation Function," in *Proc. of the 31st British Machine Vision Conference (BMVC 2020)*, 2020. [Article \(CrossRef Link\)](#)
- [25] Z. Gevorgyan, "SIOU Loss: More Powerful Learning for Bounding Box Regression," *arXiv:2205.12740*, May. 2022. [Article \(CrossRef Link\)](#)
- [26] C. Tomasi, R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. of Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pp.839-846, 1998. [Article \(CrossRef Link\)](#)



Shugang Liu received a B.Sc degree in Electronic information engineering from Hunan University, Changsha, China, in 2000, and a Ph.D. degree in Electronic Science and Technology from Hunan University, Changsha, China, in 2011. He is currently a senior lecturer at Hunan University of Science and Technology. His research interests include deep learning, semantic communication, and embedded systems.



Jialong Chen received his Bachelor of Engineering degree in Optoelectronic Information Science and Engineering from Hunan University of Science and Technology, Xiangtan, China in 2021. He is pursuing his master's degree in the School of Physics and Electronic Science, at Hunan University of Science and Technology, China. His research interests are deep learning and embedded systems.



Qiangguo Yu received a B.Sc degree in Electronic information engineering from Hunan University, Changsha, China, in 2000, and an M.Sc degree in Software Engineering from Central South University, Changsha, China, in 2016. Since 2020, he has been with Huzhou College, where he is currently a senior engineer. His current research interests include intelligent control and pattern recognition.



Jie Zhan received a B.Sc. Degree in physics from Hunan University of Science and Technology, Xiangtan, China, in 1995, and a Ph.D. degree in Electronic Science and Technology from Hunan University, Changsha, China, in 2011. He is currently a professor/doctoral supervisor at the School of Physics and Electronic Science, Hunan University of Science and Technology. His research interests include deep learning, short-range wireless communication embedded systems, etc.



Linan Duan received a B.Eng degree in Electronic information engineering from Pingdingshan University Pingdingshan, China, in 2022. He is studying for a master's degree in the School of Physics and Electronic Science of Hunan University of Science and Technology. His research interests are deep learning and embedded systems.