2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE 2021)

2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE) | 978-1-6654-1540-8/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICBAIE52039.2021.9390006

A Method for Surface Defect Detection of Printed Circuit Board Based on Improved YOLOv4

Haofei Xie[1st]
School of Automation
Chongqing University of Posts and Telecommunications
Chongqing, China.
xiehf@cqupt.edu.cn

Yuan Li[2nd*]
School of Automation
Chongqing University of Posts and Telecommunications
Chongqing, China.
1144524549@qq.com

Xingchen Li[3rd]
School of Automation
Chongqing University of Posts and Telecommunications
Chongqing, China.
673911440@qq.com

Li He[4th]
School of Automation
Chongqing University of Posts and Telecommunications
Chongqing, China.
593247526@qq.com

***Abstract*—The quality detection of printed circuit board (PCB, Printed Circuit Board) is a very important task. However, due to the diversity, complexity and smallness of PCB defects, traditional surface defect detection methods are still difficult to deal with. Therefore, this paper proposed a method for detecting surface defects of printed circuit board based on improved YOLOv4. Firstly, we used the multistage residual hybrid attention module (MRHAM) to improve feature learning to enhance the feature expression ability of the shallow network, so that the receptive field pays more attention to the object feature and ignores irrelevant features; Then, we used the K-means++ clustering algorithm perform cluster analysis on the experimental dataset, determined the anchor boxes value of the PCB defects dataset, and improved the accuracy of the model for small object defect positioning; Finally, we used online and offline data augmentation, transfer learning and multi-scale training methods to enhance the adaptability of the model to images of different input scales and improved the generalization ability of the model. Experimental results showed that compared with other traditional and deep learning object detection models, the improved model has improved detection accuracy and detection rate. The mAP value reaches 99.71% and the detection rate reaches 68 f.s^{-1}. It showed that the improved YOLOv4 has high application value in PCB surface defect detection.**

Keywords-component; PCB defect detection; attention mechanism; K-means++ clustering; data augmentation; transfer learning; Multi-scale training

I. INTRODUCTION

Printed circuit board (PCB, Printed Circuit Board) is an important part of electronic products, and its quality detection has become the key to meeting the long-term normal operation of electronic products [1]. At present, PCB surface defect detection mainly has the following 3 methods [2]: (1) Manual visual detection, using a magnifying glass or a calibrated microscope, using the operator's visual detection to determine whether the PCB board is qualified, there are large subjective errors and detection efficiency low, high labor costs, difficult data collection and other shortcomings; (2) electrical testing methods, using electrical characteristics, contact testing of the PCB board through the needle bed, there are problems such as contact testing that easily damage the PCB board, and the detection defect type is single. (3) The traditional machine vision detection method using image processing collects image data information through a camera, and uses image processing algorithms to compare the detected substrate image with the standard image, and use the set parameters as the basis to determine the position and type, there are problems such as weak generalization ability of the detection algorithm itself, poor robustness, and insufficient detected information.

In recent years, with the rapid development of deep learning, its application fields have become more and more extensive, and it has gradually shown obvious advantages in industrial defect detection. Literature et al. [3] analyzed the current situation and requirements of the detection system, and proposed the introduction of computer micro-vision into the detection system, and used the object detection algorithm based on convolutional neural network to realize the defect detection of small parts. Literature et al. [4] proposed two improved object detection algorithms, which effectively solved the problem of small samples and difficult samples, and improved the accuracy of defect detection. Literature et al. [5] proposed a PCB defect recognition algorithm based on deep learning, which can detect and recognize a variety of PCB defects, and the recognition accuracy is much higher than that of traditional defect recognition methods. Literature et al. [6] proposed a PCB bare board defect recognition algorithm based on multi-scale lightweight convolutional neural network, which is significantly better than classic convolutional network method and traditional defect detection algorithm. Literature et al. [7] proposed a PCB defect detection method based on deep learning feature pyramid network, which can effectively detect small objects in PCB defects and is compatible with multiple defect detection. In short, most of the current methods for detecting defects on printed circuit boards have problems such as low detection efficiency, difficulty in adapting to the diversity of defects, and difficulty in detecting small object defects.

The main contributions of this paper are as follows:

978-1-6654-1540-8/21/$31.00 ©2021 IEEE

1) We use multistage residual hybrid attention module (MRHAM) to improve feature learning to enhance the feature expression ability of shallow networks, so that the receptive field pays more attention to object features and ignores irrelevant features;

2) We use the K-means++ clustering algorithm to perform cluster analysis on the experimental dataset, determine the anchor boxes size of the PCB defect dataset, and improve the accuracy of the model for small object defect positioning;

3) We use online and offline data augmentation, transfer learning, and multi-scale training methods to enhance the adaptability of the model to images of different input scales and improve the generalization ability of the model.

II. Related Work

A. Introduction to YOLOv4 model

YOLOv4 [8] adds a lot of practical skills on the basis of traditional YOLOv3 [9] to achieve the best trade-off between detection speed and accuracy. Experiments show that on the Tesla V100 GPU, the real-time detection speed of the MS COCO dataset reaches 65 FPS and the accuracy reaches 43.5% AP. The overall structure of YOLOv4 is shown in Figure 1.

YOLOv4 has a great improvement in the overall structure, mainly reflected in:

a) Input: Mainly to improve the input during training, including Mosaic data augmentation, Cross mini-Batch Normalization (CmBN), Self-Adversarial Training (SAT).

b) Backbone: Combine various new methods, including CSPDarknet53, Mish activation function, and DropBlock.

c) Neck: The object detection network often inserts some layers between Backbone and the final output layer, such as the SPP module and FPN+PAN structure in YOLOv4.

d) Head: The anchor box mechanism of the output layer is the same as YOLOv3. The main improvement is the loss function GIoU loss during training, and the NMS filtered by the predicts bounding boxes becomes DIOU NMS.

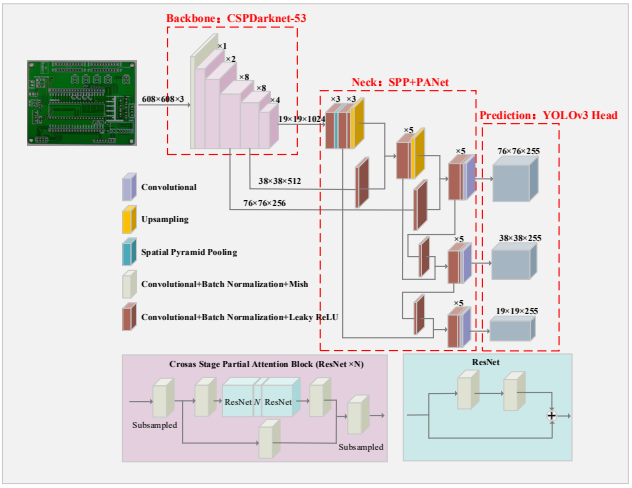


Figure 1. YOLOv4 network model

III. Model Improvement and Optimization

A. Attention-based YOLOv4 Framework

Inspired by the idea of attention mechanism [10], this paper introduces a multistage residual hybrid attention module (MRHAM) in the YOLOv4 network framework to improve feature learning. The attention module guides the detection system to learn from the area with specific information in the PCB defect image, extract the purer features of the six types of defects, and pay more attention to the space and content information of the six types of defect structures in the PCB defect image, thereby improving Feature learning improves the accuracy of the positioning of small and medium object defects in PCB.

The YOLOv4 network framework uses CSPDarknet53 [8] as the backbone feature extraction network. The purpose of using more residual identity mapping modules in CSPDarknet53 is to extract features that are very distinctive, without the gradient disappearance and disappearance in large datasets. Therefore, in order to better fit the PCB defect dataset in this article, this article has made some modifications to the CSP architecture in CSPDarknet53. Specifically, this article first reduces the number of residual identity mappings in CSP. Among them, the number of residual identity mappings in the 3rd, 4th, and 5th CSP modules in CSPDarknet53 are reduced from 8, 8 and 4 respectively to 4, 4 and 2. In order to further improve the feature extraction performance of the backbone network, this paper uses the MRHAM attention module to replace the residual identity mapping module in the CSP module structure to enhance the feature expression ability of the shallow network, so that the receptive field pays more attention to the object feature and ignores irrelevant feature. The improved network structure is shown in Figure 2. Similar to YOLOv4, several other special network structures are also used, such as spatial pyramid pool (SPP) [14], feature pyramid network (FPN) [15] and path aggregation Network (PAN) [16].

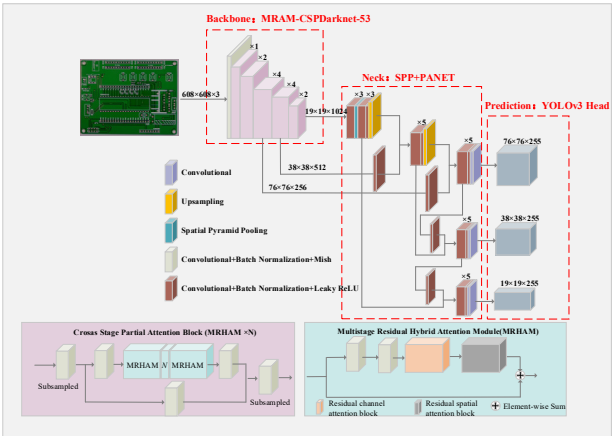


Figure 2. Improved YOLOv4 network structure

In the above network structure, the MRHAM attention module is mainly composed of residual channel attention module (RCA) and residual spatial attention module (RSA) modules. As shown in Figure 3, this module constructs the MRHAM network structure by simply stacking the RCA module

and the RSA module in a sequential manner, while considering the spatial context and channel interaction.

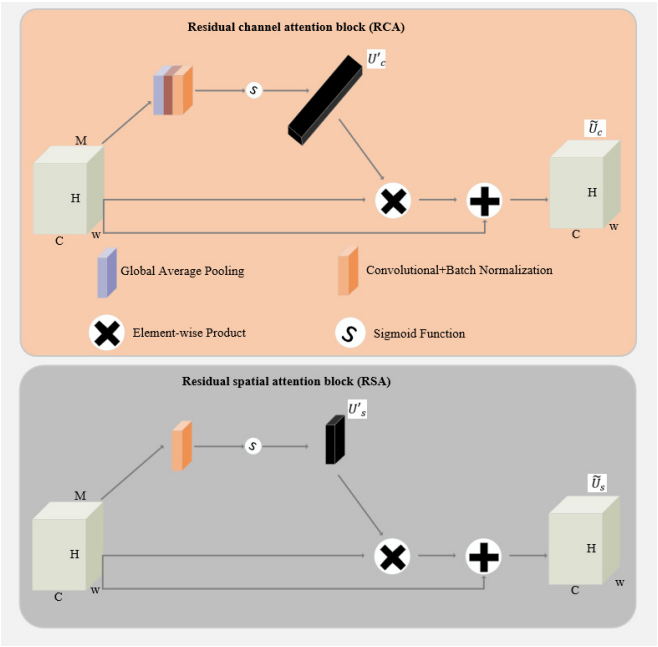


Figure 3. Multistage residual hybrid attention module

1) Residual channel attention block (RCA) module:

The RCA module is used to focus on the valuable information in the feature map and extract the unique features of the object. This article first uses the global average pooling operation to explore the effective context information of the complete feature map in the RCA module. This information will aggregate the input feature map $M \in R^{H\times W\times C}$ to generate the channel descriptor $I_c \in R^{1\times 1\times C}$ .In terms of implementation, the channel descriptor $I_c \in R^{1\times 1\times C}$ is generated by compressing the spatial size of $M \in R^{H\times W\times C}$. For example, the c-th element of I_c can be calculated by the following formula, Where $I_c = [i_1, i_2, \dots i_c]$, $M = [m_1, m_2, \dots m_c]$:

$$i_c = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} m_c(i,j) \quad (1)$$

Then, in order to further explore the information of I_c, this paper uses a two-layer CNN to capture the nonlinear relationship between channels. In this paper, the first layer of the two-layer CNN is set as the zoom layer, and the zoom rate is set to 16 to reduce parameter overhead. In addition, this paper uses a simple gating mechanism to process the channel descriptors after CNN, thereby generating a logical channel information structureU'_c. In addition, this paper uses the weighted sum operation to merge the logical spatial structure U'_c with the input feature map M to generate a temporary channel attention map. The calculation formula for the above attention calculation process as follows, where $\emptyset(.)$ is the Leaky ReLU activation function; $W_1 \in R^{\frac{C}{r}\times C}$, $W_2 \in R^{C\times\frac{C}{r}}$.

$$U'_c = \sigma\big(W_2\emptyset(W_1 I)\big) \quad (2)$$

$$U_c = U'_c \times M \quad (3)$$

Finally, in order to reduce the loss of input feature map information, this paper uses residual connection to sum the input feature map and the temporary channel attention map element by element to obtain the final channel attention feature map $\widetilde{U}$. The calculation formula is:

$$\widetilde{U} = U_c + M \quad (4)$$

2) Residual spatial attention block (RSA) module:

This paper first uses a 7×7 convolutional layer to process the input feature maps, and merges the spatial relationships of the feature maps to generate a spatial descriptor $I_s = R^{H\times W\times 1}$. Then, this article still uses the sigmoid activation function to gate the spatial descriptors to generate logical spatial information structure U'_s. In addition, similar to RCA, the input feature map is used to perform a weighted sum operation on the logical spatial information structure to generate a temporary spatial attention feature map $U_s \in R^{H\times W\times C}$. The formula for the above attention calculation process as follows, where $\sigma(.)$ is the sigmoid activation function; $Conv^{7\times 7}(.)$ represents the 7×7 convolutional layer.

$$U'_s(M) = \sigma\big(Conv^{7\times 7}(M)\big) = \sigma(I_s) \quad (5)$$

$$U_s = U'_s(M) \quad (6)$$

Similar to RCA, this paper finally connects two key feature maps (input feature map and temporary spatial attention feature map) through residuals to perform the summation operation of the elements to obtain the final spatial attention feature map $\widetilde{U} \in R^{H\times w\times C}$, the position where the code is highlighted or hidden. The calculation process is as follows:

$$\widetilde{U} = U_c + M \quad (7)$$

B. Anchor boxes cluster analysis and modification

YOLOv4 introduces the idea of RPN used in Faster R-CNN [14]. In the three feature maps of different scales output by the YOLOv4 model, each set of feature maps corresponds to three sets of anchor boxes of different scales obtained by the K-means clustering [15] algorithm.

A proper anchor boxes can effectively accelerate the convergence of the network and reduce errors. However, the anchor boxes value of the original YOLOv4 is obtained by clustering the COCO dataset, which cannot meet the application scenarios of PCB defect detection. Therefore, in order to effectively improve the positioning accuracy of small object defects in PCB defect detection application scenarios, the K-means++ clustering [16] algorithm is used to perform clustering analysis on the PCB defect dataset, and the values of 9 groups of anchor boxes can be obtained as shown in Table 1. Shown. Among them, a feature map with a larger scale uses a smaller anchor boxes, and finally the parameter size of the anchor boxes value in the configuration file is modified to retrain and test.

TABLE 1. COMPARISON OF ANCHORS VALUES UNDER DIFFERENT SCALES OF YOLOV4

Feature layer	Original cluster anchors value	Re-cluster anchors values
Feature layer 1	(142,110)(192,243)(459,401)	(26,24)(21,31)(34,36)
Feature layer 2	(36,75)(76,55)(72,146)	(23,15)(18,21)(14,28)
Feature layer 3	(12,16)(19,36)(40,28)	(9,13)(14,15)(11,20)

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Experimental dataset

The experimental dataset comes from the public dataset provided by Peking University Intelligent Robot Open Laboratory. The dataset has a total of 693 images, including six common types of defects: Missing Hole, Mouse Bite, Open Circuit, Short, Spur, Spurious Copper. Examples of defects are shown in Figure 4.

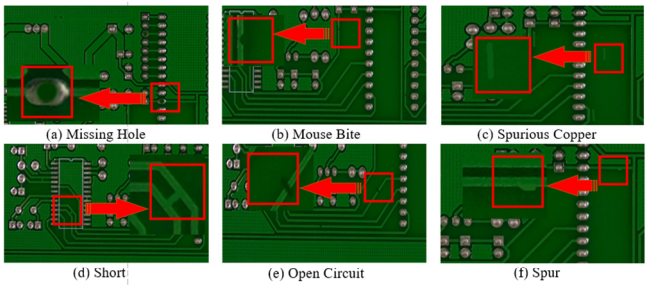


Figure 4. Examples of defects

The dataset is made into the standard PASCAL VOC2007 dataset [17] format, but the amount of data in the original dataset is relatively small, which easily leads to the occurrence of overfitting during the training process. Therefore, this article adopts two data augmentation methods to increase the richness and diversity of data. First, take offline data augmentation methods, including cropping the size, rotating the picture, adjusting the contrast, and adding noise. Second, online data augmentation is used to randomly adjust the rotation angle, saturation, exposure, hue, and Mosaic data augmentation of training images [8] to generate more training samples.

B. Experimental environment construction

In this experiment, an experimental environment for training is built on a cloud server, files and data are configured locally, and then uploaded to the cloud server to train the model. The software and hardware configuration of the cloud server is shown in Table 2.

TABLE 2. CLOUD SERVER SOFTWARE AND HARDWARE CONFIGURATION

	Configuration Environment	Description
Hardware Environment	Ubuntu 18.04.5 LTS	operating system
	Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50GHz	CPU
	GPU GeForce GTX 1080 Ti 11G	Video memory
	DDR4 16G	Memory
Software Environment	Python3.6.9	Python Version
	keras2.2.4, tensorflow1.12.0, darknet	Python Library
	scikit-learn0.19.2	Python Library
	pandas0.23.4, numpy1.15.3	Python Library
	matplotlib3.0.2, seaborn0.9.0	Python Library
	opencv-python3.4.4.19 CUDA10.1 cuDNN7.4	Python Library

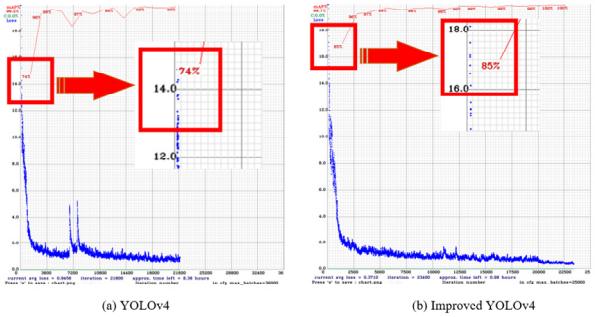
C. Training the Networks

The experimental training model uses transfer learning [18] and multi-scale training [19] methods. First of all, based on the open source deep learning framework Darknet provided by the official website, the network model based on YOLOv4 is combined with the idea of migration learning to train the entire network model. Secondly, it adopts dynamic monitoring during the training process and continuously saves training weight files. Finally, according to the loss function and mAP information, the entire network parameters are fine-tuned and optimized, and combined with the early termination strategy, the optimal weight file is saved as a predictive model for PCB defect detection. The parameter configuration for training initialization is shown in Table 3.

TABLE 3. INITIALIZATION PARAMETERS FOR NETWORK TRAINING

Parameter Name	Parameter Value
batch	16
subdivisions	8
momentum	0.949
decay	0.0005
learning_rate	0.001
NMS	0.5
saturation	1.5
exposure	1.5
hue	0.1
max_batches	25000
policy	steps
mosaic	1
random	1

The change trend of the loss function and mAP value during the training process is shown in Figure 5.



(a) YOLOv4 (b) Improved YOLOv4

Figure 5. Comparison of the change trend of loss function and mAP value

As shown in Figure 5(a), at the beginning of training, the loss function value is about 14.0. As the number of training increases, the loss value begins to gradually decrease, and the trend gradually stabilizes. When the iteration reaches about 20,000 times, the loss if the value fluctuates around 1.0, the model can be saved. At the same time, at the beginning of training, the mAP value is about 74%. As the number of training iterations increases, the mAP value gradually increases, which can indicate that the detection accuracy of the model begins to gradually

increase. When the iteration reaches about 13000 times, the mAP value it can reach about 99%, and the weight file generated by training is called for the first test on the test picture.

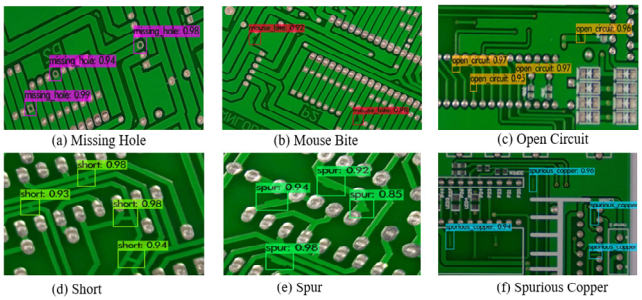


Figure 6. YOLOv4 network first test results

The test results are shown in Figure 6. The first test result shows that the model still has room for improvement in the detection of small object defects, and the test results are prone to problems such as low accuracy and missed detection.

D. Evaluation metric

In this experiment, the selected model evaluation metric [20] is:

(a) Precision and Recall:

$$P_{\text{Precision}} = \frac{TP}{TP+FP} \times 100\% \tag{8}$$

$$R_{Recall} = \frac{TP}{TP+FN} \times 100\% \tag{9}$$

Among them, TP represents the number of samples where the detected object category is consistent with the real object category, FP represents the number of samples where the detected object category is inconsistent with the real object category, and FN represents the number of samples where the real object exists but has not been detected.

(b) Average Precision and mean Average Precision:

$$\text{AP} = \int_0^1 P(R)\, dR \tag{10}$$

$$\text{mAP} = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{11}$$

Among them, N represents the number of all object categories. In general, the increase in recall rate is often accompanied by a decrease in recall rate. In order to better balance the two, this paper introduces the P-R curve, and the area under the P-R curve is the AP value of a certain category.

(c) Detection rate:

The detection rate refers to the number of pictures (frames) that the object detection network can detect per second, using FPS (Frames Per Second).

E. Result analysis

The comparative test results of the improved YOLOv4 and YOLOv4 proposed in this paper are shown in Figure 7. Experiments show that the improved YOLOv4 network model has improved detection accuracy compared with the original YOLOv4 model, and the mAP has increased by 0.674%. Secondly, the improved YOLOv4 network model improves the detection accuracy of small object defects and enhances the reliability of the model.

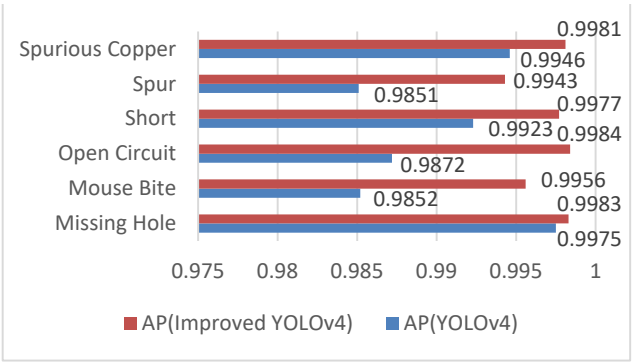


Figure 7. Comparison of detection before and after improvement

This paper proposes the design of multi-scale training. In this experiment, 4 kinds of scales (320×320, 416×416, 512×512, 608×608) images are selected for performance analysis on the test dataset. Figure 8 shows the comparison of mAP performance indicators between YOLOv4 and YOLOv4 to improve the test results of different image sizes. This shows that the improved YOLOv4 model using a multi-scale training strategy has improved detection accuracy compared to the original YOLOv4 model. The mAP is increased by 3.497% on average. It also enhances the model's adaptability to images of different input scales, which can effectively improve the detection effect of different resolution images.

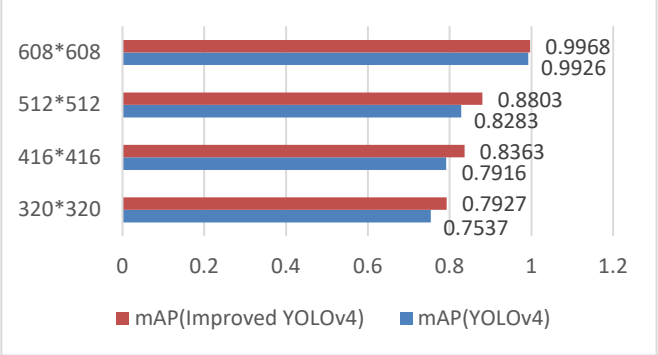


Figure 8. Comparison of two algorithms at different scales

In order to effectively verify the detection effect on the same dataset, this paper uses the SSD in literature [21], the Faster R-CNN in literature [14], and the YOLOv3 in literature [9] to compare models such as mAP and FPS. The number of recognition frames is used as the evaluation metric of the detection effect. The experimental results are shown in Table 4. The PR curves corresponding to the same detection model under different training times are shown in Fig. 9, and the PR curves of different detection models are shown in Fig. 10.

TABLE 4. COMPARISON OF DETECTION PERFORMANCE OF DIFFERENT MODELS

Model	mAP/%	FPS/（f.s^{-1}）
Faster R-CNN	95.6	0.4

SSD	84.67	41
RetinaNet	89.7	48
YOLOv3	92.98	56
YOLOv4	99.03	62
Improved YOLOv4	99.71	68

As shown in Table 4, YOLOv4 has the highest recognition accuracy among the five models compared above. Among them, the improved and optimized YOLOv4 detection accuracy can reach 99.71%, and the detection speed is significantly higher than other detection models. Compared with SSD model and YOLOv3 model, the improved YOLOv4 model has a certain improvement in mAP and FPS. It can be seen that the improved YOLOv4 model takes into account both the detection accuracy and the detection rate, and can effectively complete the PCB surface defect detection task.

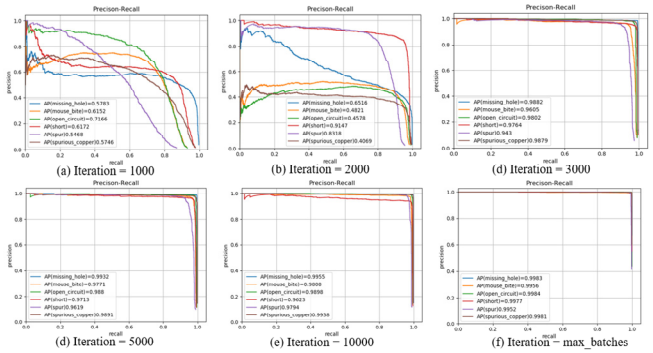


Figure 9. Improve the PR curve of YOLOv4 under different training times

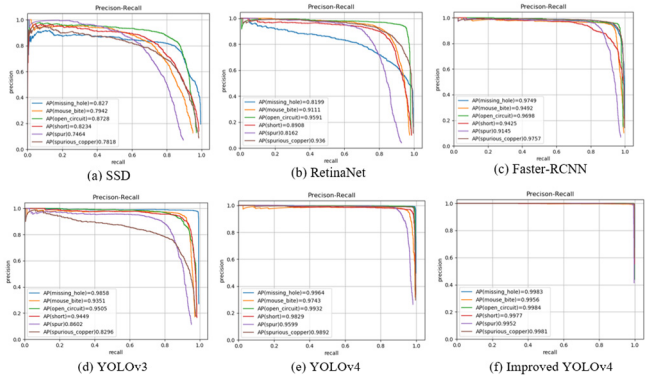


Figure 10. Improved YOLOv4 compared to other detection models

As shown in Figure 9, the PR curve of YOLOv4 under different training times is improved. It can be seen from the figure that when the number of training times is 1000 times, the AP value of the six types of defects is obviously low, and the detection effect at this time is also poor. However, as the number of training increases, the AP value of the six types of defects increases significantly, and the detection effect is getting better and better. At the same time, in the comparison of the change trend of the loss function and the mAP value in Figure 5, the improved YOLOv4 performs well in the convergence of the overall loss function and the mAP value. Moreover, as shown in Figure 10, the improved YOLOv4 is compared with other detection models. The improved YOLOv4 model proposed in this paper also has an excellent performance in PCB defect detection. The final detection effect is shown in Figure 11.

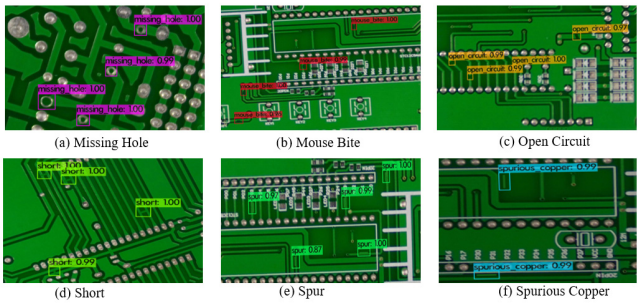


Figure 11. The final detection effect of the model

V. CONCLUSION

Based on the YOLOv4 network model, this paper uses the ideas of open datasets and transfer learning, machine learning and data visualization to closely integrate academic research and engineering applications. And from the specific project process of PCB surface defect detection, a series of processes in object detection are systematically explained, including project pre-research, algorithm selection, dataset download and packaging, environment construction, model training, model testing and model optimization etc. and combined with specific object detection application PCB surface defect detection scenarios, proposed model improvement and optimization methods, including: (1) We used the multistage residual hybrid attention module (MRHAM) to improve feature learning to enhance the feature expression ability of the shallow network, so that the receptive field pays more attention to the object feature and ignores irrelevant features; (2) We used the K-means++ clustering algorithm perform cluster analysis on the experimental dataset, determined the anchor boxes value of the PCB defects dataset, and improved the accuracy of the model for small object defect positioning; (3) We used online and offline data augmentation, transfer learning and multi-scale training methods to enhance the adaptability of the model to images of different input scales and improved the generalization ability of the model.

ACKNOWLEDGMENT

This work was financially supported by 2019 Industrial Technology Foundation Public Service Platform Project "Public Service Platform Construction for Standard Verification and Testing in the Field of Internet of Things"(No.2019-00894-1-1).

REFERENCES

[1] Song Y. Printed Circuit Board inspection System Based on Vision Processing[D]. East China University of Science and Technology, 2011.

[2] Liu C. A CNN Based Reference Comparison Method for Classifying Bare PCB Defects[J]. The Journal of Engineering, 2018, 2018(16): 1528-1533.

[3] Liu C. Research on Surface Defect Detection Technology of Tiny Parts Based on Convolutional Neural Network[D]. Harbin University of Science and Technology, 2019.

[4] Wu D. Research on Defect Detection of High Density Flexible Circuit Substrates Based on Deep Learning[D]. South China University of Technology, 2019.

[5] Yongli W., Jiangtao C., Xiaofei J. PCB Defect Detection and Recognition Algorithm Based on Convolutional Neural Network[J]. Journal of Electronic Measurement and Instrument, 2019, 33(08): 78-84.

[6] Weisen L., Yijian F. PCB Bare Board Defect Recognition Algorithm Based on Multi Scale Lightweight Convolutional Network[J]. Automation and Information Engineering, 2020, 41(05): 20-25.

[7] Hu B., Wang J. Detection of PCB Surface Defects with Improved Faster-RCNN and Feature Pyramid Network[J]. IEEE Access, 2020, 2020(99): 108335-108345.

[8] Bochkovskiy A., Wang C Y., Liao H. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. 2020.

[9] Redmon J , Farhadi A. YOLOv3: An Incremental Improvement[J]. arXiv e-prints, 2018.

[10] Itti L., Koch C., Niebur E. A model of saliency-based visual attention for rapid scene analysis[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2002, 20(11): 1254-1259.

[11] He K., Zhang X., Ren S., et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 2015(9): 1904-1916.

[12] Lin T Y., Dollár., Piotr., et al. Feature Pyramid Networks for Object Detection[J]. 2016. arXiv preprint arXiv: 1612.03144, 2016.

[13] Liu S., Qi L., Qin H., et al. Path Aggregation Network for Instance Segmentation[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City: IEEE Access, 2018: 8759-8768.

[14] Ren S., He K., Girshick R., et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(6): 1137-1149.

[15] Kwedlo W. A clustering method combining differential evolution with the K-means algorithm[J]. Pattern Recognition Letters, 2011, 32(12): 1613-1621.

[16] Arthur D., Vassilvitskii S. K-Means++: The Advantages of Careful Seeding[C]// Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans: USA, ACM, 2007: 1027–1035.

[17] Everingham M., Eslami S M A., Van Gool L., et al. The Pascal Visual Object Classes Challenge: A Retrospective[J]. International Journal of Computer Vision, 2015, 111(1): 98-136.

[18] Yosinski J., Clune J., Bengio Y., et al. How transferable are features in deep neural networks?[C]// International Conference on Neural Information Processing Systems. MIT Press, 2014: 3320–3328.

[19] Redmon J., Farhadi A. YOLO9000: Better, Faster, Stronger[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, IEEE Access, 2017: 6517-6525.

[20] Yihao W., Hongwei D., Bo L., et al. Mask wearing detection algorithm based on improved YOLOv3 in complex scenes[J]. Computer Engineering, 2020, 520(11): 18-28.

[21] Liu W., Anguelov D., Erhan D., et al. SSD: Single Shot MultiBox Detector[J]. 2016, (2016): 21-37.