Industrial Component Defect Detection Technology Based on Deep Learning

Kailun Bian
School of Software, Nanchang University, Nanchang, Jiangxi, 330031, China
1586985091@qq.com

Guo Chen
School of Software, Nanchang University, Nanchang, Jiangxi, 330031, China
2581066935@qq.com

Guoqing Xie
School of Software, Nanchang University, Nanchang, Jiangxi, 330031, China
1946159632@qq.com

Juntong Li
School of Software, Nanchang University, Nanchang, Jiangxi, 330031, China
2544984909@qq.com

Bocheng Liu *
School of Software, Nanchang University, Nanchang, Jiangxi, 330031, China
bcliu@ncu.edu.cn

ABSTRACT

In recent years, with the advancement of deep learning technology, the task of industrial component defect detection has shifted from manual inspection to deep learning model detection. However, striking a balance between the precision and speed required by industrial production has become a new challenge. This paper categorizes the current mainstream object detection algorithms into three types: one-stage detection algorithms, two-stage detection algorithms, and transformer-based detection algorithms. The structures and characteristics of each type of algorithm are elucidated. Comparative experimental studies are conducted to analyze the advantages and disadvantages of these algorithms. The paper summarizes optimization methods and effects for each type of algorithm and offers a forward-looking perspective on the prospective trends in the evolution of defect detection algorithms.

CCS CONCEPTS

• **Computing methodologies** → Artificial intelligence; Computer vision; Computer vision problems; Object detection.

KEYWORDS

Object detection, Deep learning, Transformer, Yolo, Industrial component defect

ACM Reference Format:
Kailun Bian, Guo Chen, Guoqing Xie, Juntong Li, and Bocheng Liu *. 2024. Industrial Component Defect Detection Technology Based on Deep Learning. In *International Conference on Algorithms, Software Engineering, and Network Security (ASENS 2024), April 26–28, 2024, Nanchang, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3677182.3677297

ASENS 2024, April 26–28, 2024, Nanchang, China

ACM ISBN 979-8-4007-0978-4/24/04
https://doi.org/10.1145/3677182.3677297

1 INTRODUCTION

In 2023, China's industrial economy is overall improving, with the manufacturing industry maintaining its global leadership for 14 consecutive years. The "Digital Transformation Implementation Plan for Manufacturing Industry in Jiangxi Province" [1] proposes accelerating the widespread use of artificial intelligence in the production and manufacturing processes, aiming for the Informationization and Industrialization Fusion Development Index to exceed the national average by 2025. Consequently, many enterprises are adopting intelligent production lines to optimize the means of industrial component defect detection, ensuring that products possess core competitiveness in the market.

Traditional methods for detecting defects in industrial components mainly involve manual inspection and detection methods based on traditional machine vision. Due to the often small and imperceptible nature of component defects, manual inspection exhibits low accuracy, efficiency, high labor intensity, and is significantly influenced by human experience and subjective factors. As a result, there is a possibility of overlooking defects, leading to a relatively low detection rate. Manual report summarization lacks real-time statistical analysis of defect types, resulting in weak real-time feedback. Although detection methods based on traditional machine vision significantly improve detection speed and accuracy, as the number of categories increases, feature extraction becomes increasingly challenging, and the need to handle a growing number of parameters makes it difficult to meet the requirements of industrial production.

In recent years, the rapid development of deep learning technology has brought more efficient solutions to the detection of defects in industrial components. Deep learning technology excels in tasks with strong randomness and complex features. Currently, mainstream object detection algorithms can be broadly categorized into three types. The first type is the two-stage detection algorithm based on region proposal, such as the RCNN series. This type of algorithm involves two steps: generating region proposal and then classifying and locating the candidate targets. The second type is the one-stage detection algorithm based on non-region proposal, such as the YOLO series. This type of algorithm defines the detection task as an end-to-end regression problem, directly

extracting features in the network to determine the probability and position of the target class. The third type is the algorithm based on Transformer, such as DETR and RT-DETR. This type of algorithm introduces attention mechanisms into the field of target detection.

This paper conducts experimental comparisons of the above three types of algorithms, analyzing the network structures and characteristics of each. Based on the actual industrial production environment, models with better accuracy and speed are selected to provide new optimization directions for future industrial component defect detection algorithms.

2 RELATED WORK

The defect detection algorithm based on deep learning primarily utilizes object detection techniques to perform surface defect detection on industrial components, accomplishing the classification and localization of defects in target images. As of now, common deep learning-based defect detection algorithms can be categorized into three types: two-stage detection algorithms [2], one-stage detection algorithms [3], transformer-based end-to-end detection algorithms [4].

As a classic two-stage detection algorithm, RCNN [5], proposed by Ross Girshick in 2014, initially generates numerous region proposals using the Selective Search algorithm, selecting potential bounding boxes that may contain target objects. Subsequently, it adjusts the sizes of these region proposals, extracts features using a pre-trained CNN, and feeds the extracted features into an SVM classifier to predict the presence of the target and its category. Finally, bounding box refinement is performed through bounding box regression and non-maximum suppression. Building upon RCNN, Girshick introduced Fast RCNN [6], which incorporates Spatial Pyramid Pooling and the Region of Interest pooling layer. It shares the entire image's feature extraction process, combining the classification and regression tasks into a single loss function. Fast RCNN increased detection accuracy (mAP) from 58.5% to 70.0% on the VOC-07 dataset, with a 200-fold improvement in detection speed compared to RCNN. However, Fast RCNN still relies on the Selective Search algorithm for region proposal selection, leading to suboptimal processing speed on large datasets. Addressing these limitations, Shaoqing Ren and others proposed the Faster RCNN algorithm [7], introducing the Regional Proposal Network to generate region proposals, reducing the number of region proposals, enhancing the quality of region proposals, and improving object detection accuracy and speed. Faster RCNN outperformed Fast RCNN by 3.2% on the VOC-07 dataset. In 2017, T.-Y. Lin and colleagues further improved upon Faster RCNN by introducing Feature Pyramid Networks (FPN). FPN adopts a top-down and bottom-up information propagation mechanism, better capturing the semantic and detailed information of objects and enhancing detection accuracy.

To address the speed limitations of Faster RCNN for real-time requirements, YOLO (You Only Look Once) [8] was introduced in 2016. YOLO directly regresses object boxes based on feature maps, eliminating region proposal generation and fine-tuning. While YOLO achieves faster detection, it suffers from lower detection accuracy. YOLOv2 [9] and YOLOv3 [10] were subsequently proposed to enhance accuracy and speed, with YOLOv3 introducing Darknet-53 residual network and Feature Pyramid Networks for multi-scale prediction. Another notable one-stage detection algorithm is SSD (Single Shot Multibox Detector) [11], which employs multi-reference and multi-resolution detection techniques. SSD significantly improves multi-scale detection accuracy. However, it faces challenges in detecting small objects due to the limited semantic information in shallowly generated small target features. FSSD (Feature Fusion Separate Shot Multi-box Detector) [12] enhances SSD's performance by incorporating a lightweight and efficient feature fusion module. DSSD (Deconvolutional Single Shot Detector) [13] replaces VGG net with ResNet-101 for improved feature extraction capabilities and introduces deconvolution for context information. These modifications are effective for small object detection, although the increased depth of ResNet-101 makes it slower than SSD.

The Transformer, initially proposed by Vaswani et al. [14], is a novel machine translation building block based on attention mechanisms. DETR (DEtection Transfomer) is a target detection system introduced by the Facebook AI team, leveraging transformers and bipartite matching loss for direct set prediction. Although DETR achieves results comparable to optimized Faster R-CNN on challenging datasets like COCO, it suffers from slow convergence and limited feature space resolution.To mitigate these issues, Deformable DETR [15] was proposed in 2021, fusing deformable convolution's sparse spatial sampling with transformer-related modeling capabilities. Deformable DETR outperforms DETR, particularly in small object detection, with a tenfold reduction in training time. In Transformer-based detectors, object queries are a set of learned embeddings that do not focus on specific regions. In 2022, Anchor DETR was proposed to address this limitation. It introduces an object query based on anchor points and incorporates multiple patterns for improvement. Additionally, Anchor DETR employs a memory-efficient attention variant, Row-Column Decouple Attention (RCDA).

The field of deep learning-based object detection algorithms is rapidly evolving, and not all algorithms are suitable for industrial component defect detection due to limitations imposed by industrial production environments. Therefore, this study will conduct comparative experiments to analyze algorithms capable of addressing industrial component defect detection.

3 METHODOLOGY

3.1 Dataset

This paper uses the dataset from the Baidu Paddle Learning Competition: Steel Defect Detection Challenge. The dataset consists of 1400 grayscale images along with their corresponding XML annotation files. It contains six typical defects of steel components, as shown in Figure 1. These defects include Roll Surface, Patch, Crack, Pitted Surface, Inclusion, and Scratch. The division ratio of the test set is 8:2.

3.2 Data Augmentation

Through the analysis of the dataset, it was found that there is a significant difference in the brightness of the images. Histogram equalization was employed to balance the brightness of the images, and the results are shown in Figure 2. The PPYOLOE_s model was trained for 60 rounds, and comparative experiments were conducted.

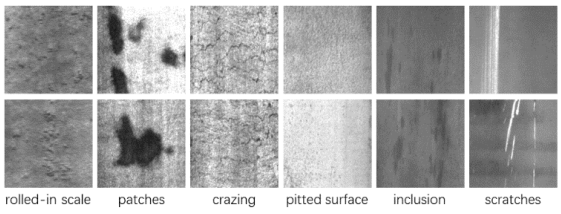

Figure 1: Examples of defect types in steel components.

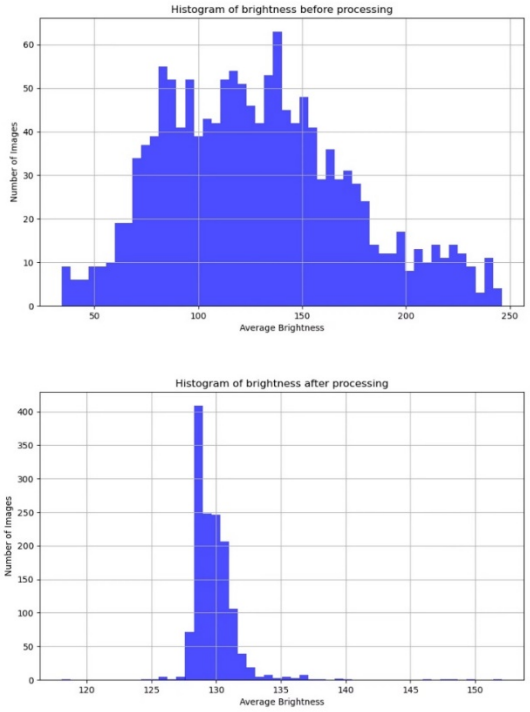

Figure 2: Before and after comparison of histogram equalization.

Observing the training results, the mAP value before equalization was 79.49%, and after equalization, it was 77.51%. In comparison to object detection in natural scenes, industrial component defect detection data primarily involve low-level semantic information with less interference from factors like illumination. Therefore, it is evident that most data augmentation methods are not conducive to improving algorithm accuracy. This paper only adopted image flipping for data augmentation, along with image scaling and normalization to facilitate training.

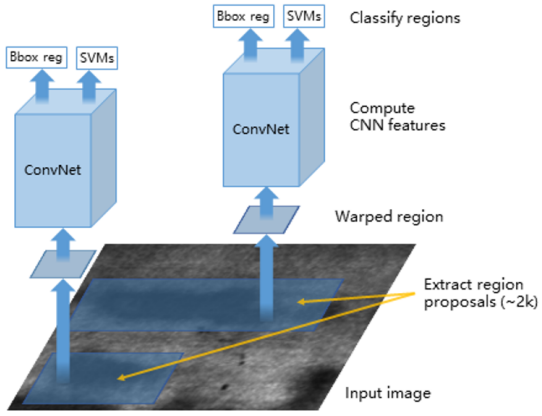

Figure 3: The detection process of two-stage detection algorithm(RCNN).

3.3 Defect Detection Algorithm

3.3.1 Two-Stage Detection Algorithms Based on Region Proposals. The pioneering work of region proposal-based two-stage detection algorithms is RCNN, followed by high-performance models like Faster-RCNN and Swin-Faster-RCNN.

RCNN combines region proposals and CNNs, divided into three modules as shown in Figure 3. The first module generates class-agnostic region proposals, defining a set of candidate detection regions. The second module is a convolutional neural network that extracts a fixed-length feature vector from each region. The third part is a linear SVM specific to the designated category for classifying each region.

(a) Faster-RCNN.

Faster-RCNN is a representative model in the category of two-stage object detection. It integrates feature extraction, proposal extraction, bounding box regression, and classification into a single network. It replaces the use of segmentation algorithms with Region Proposal Network (RPN) for generating candidate boxes, significantly improving overall performance and greatly enhancing box generation speed.

(b) Swin-Faster-RCNN.

To address the insufficient feature extraction of CNN networks, Swin Transformer is employed as the backbone of Faster-RCNN. It adopts a hierarchical design, consisting of four stages, each reducing the resolution of the input feature map, progressively expanding the receptive field like CNN. By replacing the traditional convolutional backbone with Swin Transformer, it becomes adaptable to more complex tasks with varying scales.

3.3.2 One-Stage Detection Algorithms Based on Non-Region Proposals. Two-stage detection algorithms based on region proposals require setting numerous anchor boxes in advance and need a specialized RPN to correct the positions of anchor boxes. This setup makes two-stage object detection algorithms complex and computationally slow. The emergence of YOLO initiated the exploration of one-stage detection algorithms. YOLO, short for You Only Look Once, processes an image through a neural network only once,

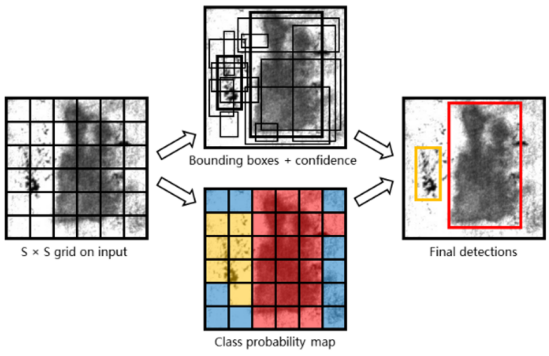

Figure 4: The detection process of one-stage detection algorithm (YOLO).

predicting both the location and category simultaneously. This defines the object detection task as an end-to-end regression problem, enhancing computational speed.

The implementation process of YOLO is shown in Figure 4. Firstly, the input image is resized to a uniform size and divided into an S×S grid on the image. Each grid is responsible for detecting an object with its center falling on it, predicting the confidence, category, and position of the object. Secondly, CNN is used for feature extraction, and a fully connected layer is employed to regressively predict the object's probabilities in various categories. Finally, Non-Maximum Suppression (NMS) is used to process bounding boxes to obtain optimal results.

(a) YOLOv3.

YOLOv3, proposed by Joseph Redmon and Ali Farhadi in 2018, is the third version of the YOLO series. YOLOv3 uses the Darknet-53 network as its backbone, employing strategies such as multiscale feature fusion and hierarchical predictions, enhancing both detection speed and accuracy.

(b) YOLOv8.

YOLOv8, the latest version of the YOLO series developed by Ultralytics, chooses the Decoupled Head and Anchor-Free strategy, eliminating the previous Objectness branch and retaining only decoupled classification and regression branches. This improves the accuracy and efficiency of the detection process. YOLOv8 removes the convolutional structure in the upsampling stage of PAN-FPN and replaces the C3 module with the C2f module.

(c) PPYOLOE+.

PPYOLOE+ continuously improves model performance through a combination of reasonable tricks. It selects ResNet50-vd as the entire architecture, replacing some convolutional layers with deformable convolutions, and appropriately increases network complexity. It includes modules like the backbone network CSPRepResNet, feature fusion CSPPAN, lightweight ET-Head, and an improved dynamic matching algorithm TAL (Task Alignment Learning).

(d) SSD.

The SSD model framework is mainly composed of three parts. Taking SSD300 as an example, it has VGG-Base, Extra-Layers, and Pred-Layers. SSD improves upon YOLO in several aspects: SSD feeds multiple features from different layers of the feature extraction network into the object detection module, enhancing accuracy in detecting small objects. Inspired by the concept of anchor boxes in Faster R-CNN, SSD sets different-sized and aspect ratio anchor boxes for each grid, reducing training difficulty. SSD also replaces fully connected layers with convolutional layers for regression predictions on different feature maps, further reducing model parameters and improving computational speed.

3.3.3 Transformer-Based Detection Algorithms. (a) RT-DETR.

DETR is the first end-to-end algorithm based on transformers. Its algorithm process is shown in Figure 5. Unlike YOLO, DETR does not have anchor preprocessing and NMS post-processing, allowing it to complete the entire object detection process directly in the network. However, DETR suffers from slow convergence, slow training, slow inference, and cannot meet real-time requirements.

RT-DETR improves upon the DETR and DINO detection model, achieving real-time end-to-end detection for the first time. RT-DETR can be divided into three parts: the backbone network, the neck network, and the head network. For the backbone part, two classic networks, ResNet and scalable HGNetv2 are used. For the neck network part, RT-DETR employs a one-layer Transformer Encoder. The detection process of RT-DETR is shown in Figure 6. The high-efficiency mixed encoder uses Scale-Inside Feature Interaction (AIFI) and Cross-Scale Feature Fusion Module (CCFM) to transform multiscale features into an image feature sequence. A fixed number of image features are chosen as initial object queries for the decoder. Finally, auxiliary prediction heads for the decoder iteratively optimize queries to generate boxes and confidence scores.

Table 1: Experimental Environment.

Environment	Version
Operating System	Windows 11
CPU	AMD Ryzen 7 5800H
GPU	NVIDIA GeForce RTX 3050 Ti
Anaconda	2022.10-x86_64
Python	Python 3.7

3.4 Experimental Methods and Evaluation Metrics.

The experimental environment in this study is presented in Table 1. All models are constructed using the PaddlePaddle deep learning framework and scientific computing libraries. The training times Epoch=200, employing a segmented learning rate decay strategy to accelerate network training speed. Upon convergence of the final loss function, the trained network models are obtained for comparison.

The miain evaluation metrics adopted in this study are as follows:

1) mAP: Represents the average value of AP across multiple classes, used to measure the overall accuracy performance of the algorithm across all classes, C is the number of classes.

$$mAP = \sum AP/C \tag{1}$$

2) FPS: Represents the number of images that can be processed per second, used to evaluate the speed performance of the object

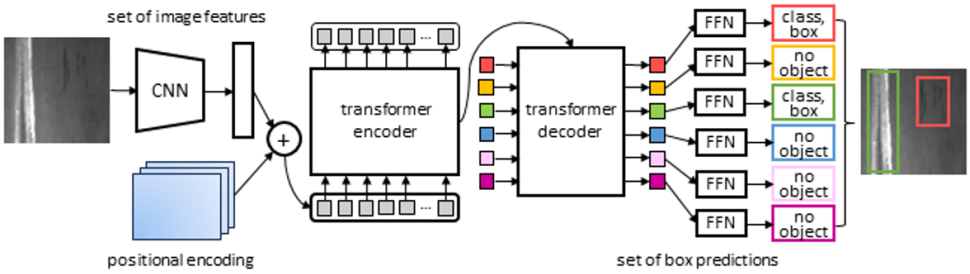

Figure 5: The detection process of DETR.

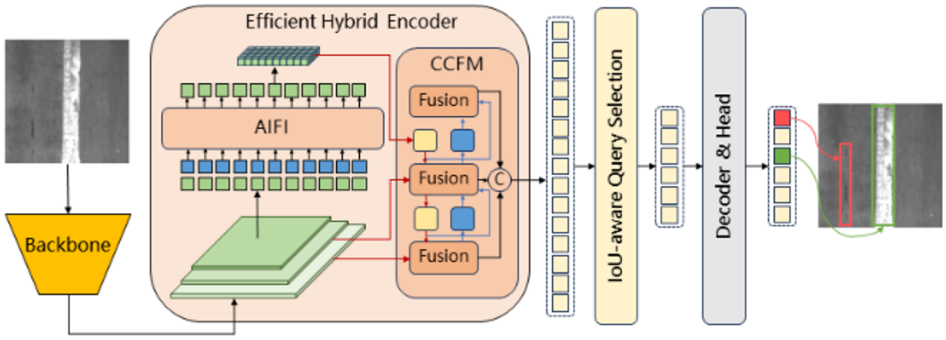

Figure 6: The detection process of RT-DETR.

detection algorithm. Calculated as:

$$FPS = N/T \tag{2}$$

Where N is the total number of frames in the target image or video, and T is the processing time.

4 RESULTS AND DATA ANALYSIS

After the completion of training for each detection algorithm, testing was conducted on the validation set, and the AP values, mAP values, and FPS for various defect classes were recorded. The corresponding precision and recall were calculated, and the results are presented in Table 2 and Table 3, as well as Figure 7.

The experimental results indicate that:

(1) The mAP values of Faster-RCNN and swin-Faster-RCNN are higher than those of the YOLO series, SSD, and RT-DETR, with swin-Faster-RCNN reaching a mAP of 80.07%. However, the FPS of Faster-RCNN and swin-Faster-RCNN are significantly lower than other algorithms. Two-stage detection algorithms based on region proposal exhibit high detection accuracy and precise localization but have slower detection speeds. Additionally, due to different feature extraction methods, two-stage detection algorithms have lower recall than one-stage detection algorithms, making them prone to missed detections. Therefore, they are not suitable for real-time detection in industrial production environments. One-stage detection algorithms based on non-region proposal exhibit slightly lower detection and localization accuracy, but their detection speed is much higher than that of two-stage detection algorithms. The transformer-based detection algorithm RT-DETR achieves a balance between detection accuracy and speed, making it suitable for real-time detection tasks.

(2) For the same model, in the YOLOv8 algorithm, the detection accuracy of YOLOv8-m and YOLOv8-s is lower than that of YOLOv8-n. This suggests that the accuracy of the model does not necessarily increase with the depth of the model; instead, it may decrease. In industrial production environments with a lack of annotated samples, deep models with many redundant parameters can lead to a decrease in various model metrics and a reduction in detection speed.

(3) The detection performance for the "crazy" and "rolled-in_scale" defect classes is not ideal, with low AP values for these defects across various models. This indicates that deep learning detection models exhibit poor performance in detecting dense, subtle, and low-contrast defects of this nature.

5 CONCLUSIONS

The production of industrial components imposes higher demands on the speed and accuracy of deep learning models. Due to constraints on the number of annotated samples, the depth and complexity of algorithm networks also need to be controlled. Therefore, this paper conducted experimental comparisons of three popular types of deep learning detection algorithms. Although two-stage detection algorithms exhibit high accuracy, their lower recall rate and slower speed make them unsuitable for real-time detection tasks. One-stage detection algorithms strike a balance between speed and accuracy, but due to limitations in network structure, achieving

Table 2: Comparison of Detection Capabilities for Different Defectst.

Defect Detection Technology	AP (IoU=0.5)					
	crazing	inclusion	pitted_surface	scratches	patches	rolled-in_scale
FasterRCNN	0.474	0.833	0.849	0.909	0.948	0.674
Swin-FasterRcnn	0.561	0.816	0.857	0.922	0.943	0.706
YOLOv3	0.370	0.749	0.715	0.881	0.961	0.552
YOLOv8-n	0.463	0.815	0.837	0.925	0.931	0.574
YOLOv8-s	0.458	0.809	0.834	0.909	0.942	0.57
YOLOv8-m	0.457	0.808	0.836	0.904	0.935	0.573
PPYOLOE+	0.412	0.814	0.785	0.921	0.921	0.621
SSD	0.505	0.807	0.831	0.722	0.960	0.694
RT-DETR	0.429	0.857	0.889	0.935	0.928	0.599
Average	0.459	0.812	0.826	0.892	0.941	0.618

Table 3: Comparison of Detection Performance.

Name	Precision	Recall	mAP@0.5	FPS
FasterRCNN	0.779	0.555	78.12	9.381
Swin-FasterRcnn	0.798	0.550	80.07	8.859
YOLOv3	0.698	0.449	70.45	44.616
YOLOv8-n	0.695	0.729	75.80	60.387
YOLOv8-s	0.790	0.688	75.40	59.102
YOLOv8-m	0.744	0.704	75.20	31.368
PPYOLOE+	0.744	0.688	74.57	20.350
SSD	0.751	0.524	75.32	36.390
RT-DETR	0.737	0.707	77.60	28.568

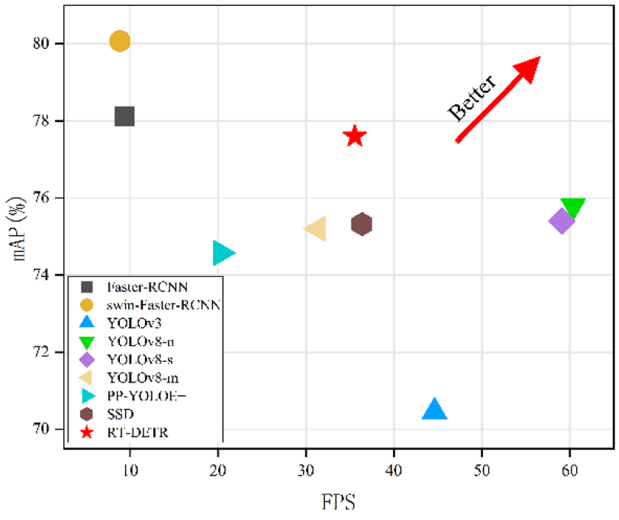

Figure 7: The Speed and Accuracy of Algorithms.

further improvements in accuracy becomes challenging. The emergence of vision Transformers has provided new insights for object detection algorithms. RT-DETR, has been improved to achieve higher accuracy and speed, demonstrating its capability to perform real-time end-to-end detection tasks. However, Transformer-based object detection algorithms rely more on the quality and scale of annotated samples compared to CNNs. Challenges still exist in detecting small targets with few samples, making the detection of fewer samples and smaller targets an ongoing challenge for researchers. Further research and exploration are needed for object detection algorithms based on transformers.

ACKNOWLEDGMENTS

This work is supported by key research and development program of Jiangxi Province (Grant No. 20232BBH80017) and Jiangxi Students' innovation and entrepreneurship training program.

REFERENCES

[1] People's Government of Jiangxi Province. 2023. Notice of General Office of Jiangxi Provincial People's Government on issuing the Implementation Plan for Digital Transformation of Manufacturing Industry in Jiangxi Province.http://www.jiangxi.gov.cn/art/2023/6/17/art_4975_4501662.html?

[2] Tao, X., Hou, W., & Xu, D. 2021. Overview of surface defect detection methods based on deep learning. Acta Automatica Sinica (05),1017-1034. doi:10.16383/j.aas.c190811.

[3] Fei, J., Li, H., Ren, F., Wu, M., & Wang, G. 2023. A review of steel surface defect detection methods based on deep learning. Modern information technology (19), 107-112. doi:10.19850/j.cnki.2096-4706.2023.19.023.

[4] Li, J., Du, J., Zhu, Y., & Guo, Y. 2023. Review of target detection algorithms based on Transformer. Computer engineering and applications (10), 48-64.

[5] Girshick, R., Donahue, J., Darrell, T., & Malik, J. 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 580–587. https://doi.org/10.1109/cvpr.2014.81

[6] Girshick, R. 2015. Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV), 1440–1448. https://doi.org/10.1109/iccv.2015.169

[7] Ren, S., He, K., Girshick, R.B., & Sun, J. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 1137-1149.
[8] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779–788. https://doi.org/10.1109/cvpr.2016.91
[9] Redmon, J., & Farhadi, A. 2017. YOLO9000: Better, Faster, Stronger. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2017.690
[10] Redmon, J., & Farhadi, A. 2018. YOLOv3: An Incremental Improvement. Arxiv.org. https://doi.org/10.48550/arXiv.1804.02767
[11] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. 2016. SSD: Single Shot MultiBox Detector. Computer Vision – ECCV 2016, 9905, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2
[12] Li, Z., & Zhou, F. 2017. FSSD: Feature Fusion Single Shot Multibox Detector. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1712.00960
[13] Fu, C.-Y., Liu, W., Ananth Ranga, Tyagi, A., & Berg, A. C. 2017. DSSD: Deconvolutional Single Shot Detector. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1701.06659
[14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. 2017, December 5. Attention Is All You Need. ArXiv.org. https://doi.org/10.48550/arXiv.1706.03762
[15] Zhu, X., Su, W., Lu, L., Li, B., Wang, X., & Dai, J. 2020. Deformable DETR: Deformable Transformers for End-to-End Object Detection. ArXiv, abs/2010.04159.