Fusion Lightweight Steel Surface Defect Detection Algorithm Based on Improved Deep Learning

Fei Ren[1], Jiajie Fei[2], HongSheng Li[3]*, Bonifacio T. Doma Jr[4]*
School of Information Technology / School of Grad Studies, Mapua University, Manila, Philippines[1, 4]
School of Automation, Nanjing Institute of Technology, Nanjing, China[2, 3]

***Abstract*—In industrial production, timely and accurate detection and identification of surface defects in steel materials were crucial for ensuring product quality, enhancing production efficiency, and reducing production costs. This study addressed the problem of surface defect detection in steel materials by proposing an algorithm based on an improved version of YOLOv5. The algorithm achieved lightweight and high efficiency by incorporating the MobileNet series network. Experimental results demonstrated that the improved algorithm significantly reduced inference time and model file size while maintaining performance. Specifically, the YOLOv5-MobileNet-Small model exhibited slightly lower performance but excelled in inference time and model file size. On the other hand, the YOLOv5-MobileNet-Large model achieved a slight performance improvement while significantly reducing inference time and model file size. These results indicated that the improved algorithm could achieve lightweighting while maintaining performance, showing promising applications in steel surface defect detection tasks. It provided an efficient and feasible solution for this important domain, offering new insights and methods for similar surface defect detection problems and contributing to research and applications in related fields.**

***Keywords*—Deep learning; improved YOLOv5; YOLOv5-mobilenet; surface defects**

I. INTRODUCTION

Surface defect detection [1] was paramount in industrial production as it aided in promptly identifying and rectifying flaws on product surfaces, ensuring product quality adhered to standards, enhancing production efficiency, and reducing reject rates. This not only helped in cost and resource savings but also enhanced product safety and reliability, maintaining a company's reputation and market competitiveness. Extensive research had been conducted by numerous scholars in this field.

In existing research, Zhang Guo et al. [2] proposed an FFS-YOLO model based on the improved YOLOv4-tiny model for detecting PCB surface defects. While this model enhanced detection accuracy and light weighted the model, the detection metrics only included mAP@0.5, FPS, and model size, lacking comprehensive evaluation metrics such as recall and precision, requiring further research and validation. Dong Yongfeng et al. [3] presented a defect detection joint optimization algorithm based on attention mechanism, showing promising results in classifying multiple defect types. However, the algorithm's joint loss function involved numerous hyperparameters, making manual adjustments challenging, and it did not address real-time issues. Divyanshi Dwivedi et al. [4] tackled renewable energy asset surface defect detection using the latest deep learning model ViT. While effective in image classification tasks, this approach still needed to address challenges related to data quality and environmental adaptability. Wu Jiling et al. [5] proposed an improved Faster R-CNN algorithm, optimizing feature extraction networks, region of interest pooling, and anchor box sizes. Additionally, they introduced feature pyramids and deformable convolutions, achieving satisfactory detection results. Future research should focus on lightweighting detection models while enhancing detection speed without compromising accuracy to facilitate proactive industrial deployment.

YOLOv5 exhibited efficient end-to-end detection capabilities, while MobileNet was a lightweight convolutional neural network. Their combination addressed the need for both detection performance and model efficiency, aligning with the requirements for real-time operation and deployment convenience in steel surface defect detection tasks. In contrast, existing models like Faster R-CNN, although they demonstrated good detection accuracy, were less suitable for industrial real-time detection scenarios due to their complex network structures and substantial computational demands, which resulted in low inference efficiency. Additionally, they lacked optimization designs targeted at lightweighting.

This study addressed the issue of detecting surface defects in steel materials by proposing an algorithm based on an improved version of YOLOv5. Compared to existing methods, our algorithm incorporated a lightweight MobileNet network, which significantly reduced the model inference time and file size while maintaining detection performance. Additionally, it notably enhanced real-time capabilities and deployment convenience.

Furthermore, our evaluation metrics were more comprehensive, including not only common metrics such as mean Average Precision (mAP) and inference time but also precision and recall rates, providing a more objective reflection of the algorithm's performance. Our algorithm demanded less in terms of data quality and environmental adaptability, demonstrating stronger generalization capabilities. It was highly practical and applicable, offering a new efficient solution for quality control in the steel industry and bringing fresh insights and methods to similar surface defect detection issues, thereby possessing significant theoretical and practical value.

Overall, this study was dedicated to proposing an efficient, lightweight, and high-performing algorithm for detecting surface defects in steel materials. It aimed to address the shortcomings of existing methods in terms of real-time performance, lightweight design, comprehensive evaluation metrics, and generalization capabilities. This work contributed to research and applications in related fields.

II. Deep Learning YOLO Algorithm

YOLOv5 was regarded as the pinnacle of the YOLO series, highly favored by both the academic and industrial communities for its outstanding detection accuracy and fastest detection speed [6]. The network architecture of YOLOv5 followed the overall layout of YOLOv3 and YOLOv4, mainly comprising four parts: the input layer, backbone network, neck network, and prediction layer, as shown in Fig. 1.

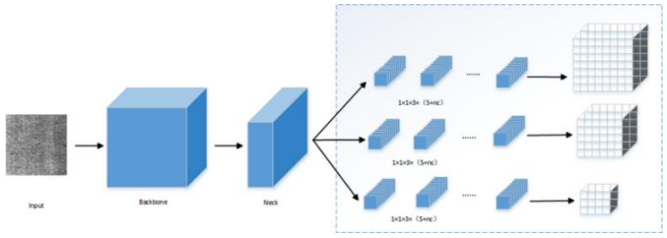


Fig. 1. YOLOv5 structure.

Input: The mosaic data augmentation method was employed, which involved randomly cropping four images (each with corresponding bounding boxes) and then stitching them together into a new image. This method significantly increased the background information of target objects.

Backbone: The Focus structure and CSP structure played different but complementary roles in deep learning models. The Focus structure was primarily used to reduce computational complexity and improve inference speed, while the CSP structure, through reasonable branch design, enabled the model to learn more features while reducing computational complexity, thereby enhancing the model's performance. The combination of these two structures could effectively optimize the model's performance, making it more efficient and reliable in practical applications.

Neck: FPN+PAN structure was utilized. FPN (Feature Pyramid Network) and PAN (Path Aggregation Network) were two common network structures for object detection, used to handle multi-scale feature maps. FPN propagated strong semantic features through upsampling, while PAN propagated strong localization features through downsampling. Combining FPN and PAN enhanced semantic expression and localization capabilities at multiple scales, thereby improving the performance and robustness of object detection models at different scales.

Prediction: GIoU Loss was introduced as the loss function for bounding boxes. This loss function effectively addressed the problem of non-overlapping bounding boxes, thereby improving the accuracy and precision of object detection. The application of GIoU Loss enabled the model to better understand the position and shape of objects, thereby improving detection accuracy. NMS helped to find the optimal position of detected objects and removed overlapping detection boxes, further enhancing the accuracy and robustness of object detection. This step made the model's output results clearer and more reliable, providing a more trustworthy solution for object detection tasks in real-world scenarios [7-8].

III. Improved YOLOv5 Algorithm with MobileNet

A. MobileNet Algorithm

In 2017, the Google team introduced MobileNetv1, which replaced ordinary convolutional modules with depthwise separable convolutions to achieve lightweight convolutional neural networks [9]. By using depthwise separable convolutions, the parameter count of MobileNetv1 was reduced to around 1/8 to 1/9 of its original size. Compared to VGG16, it only sacrificed approximately 0.9% of classification accuracy while reducing the parameter count to only 1/32.

MobileNetv2 introduced "residual modules" on the basis of MobileNetv1. These residual modules first used 1x1 convolutions for dimensionality expansion, followed by 1x1 convolutions for dimensionality reduction, also known as inverted residual modules. Furthermore, to prevent significant loss of low-dimensional information under the ReLU activation function, MobileNetv2 used linear activation functions for the last layer convolution [10].

MobileNetv3 is an improved version of MobileNetv2, with superior accuracy and smaller model size. MobileNetv3-Large and MobileNetv3-Small are neural network structures optimized for mobile devices using Neural Architecture Search (NAS) technology. Although the backbone network structures of the two are similar, they contain different numbers of Bneck modules [11-12]. MobileNetv3-Large has 15 Bneck modules, while MobileNetv3-Small contains only 11 Bneck modules. The specific structure of the Bneck module is illustrated in Fig. 2. These improvements enable MobileNetv3 to maintain its lightweight nature while enhancing model accuracy, making it an ideal choice for mobile devices.

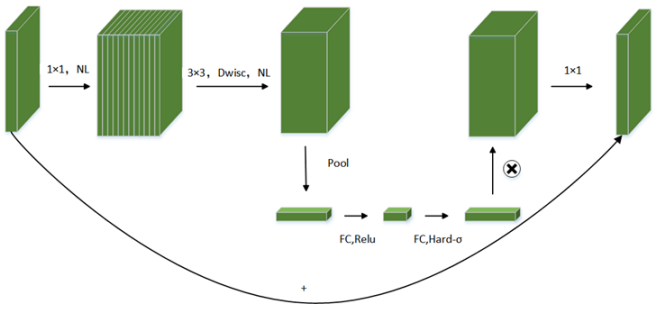


Fig. 2. Specific structure of Bneck module.

The MobileNetV3 network integrates various advanced neural network structures, including depthwise separable convolutions from MobileNetV1, linear bottleneck inverted residual structures from MobileNetV2, and the lightweight attention model from MnasNet. Additionally, it introduces the non-linear Swish function, computed as shown in Eq. (1).

$$swish[x] = x * \sigma(x) \tag{1}$$

In the Eq. (1), swish[x]represents the non-linear activation function, where x denotes the input feature, and $\sigma(x)$ represents the sigmoid activation function.

MobileNetV3 ultimately adopts a new activation function, denoted as h-swish[x], to replace the original Swish[x] function. This change is due to the high computational cost of computing the Sigmoid function on mobile devices. The new activation function h-swish[x] significantly improves detection speed, especially in deep networks. The computation process is shown in Eq. (2).

$$h-swish[x] = x * Relu6\frac{x+3}{6} \quad (2)$$

B. The Improved YOLO Algorithm

YOLOv5 was considered a regression-based one-stage object detection algorithm [13-14]. In order to enhance the model's performance, the MobileNetv3 network was utilized to replace the original backbone network, CSPDarkNet53. Apart from this change in the backbone network, the rest of YOLOv5 remained consistent with the original model. MobileNetv3, compared to CSPDarkNet53, featured a more lightweight network structure and higher computational efficiency. Consequently, it facilitated accelerated execution of object detection while preserving model accuracy. This improvement contributed to YOLOv5's superior performance in real-time object detection scenarios. The structure is depicted in Fig. 3.

In the improved version of YOLOv5 after the incorporation of MobileNetv3, the obtained feature matrix underwent a series of transformations. Initially, it was processed through a 1x1 convolution, followed by input into the pyramid spatial module. Down-sampling occurred at three parallel max-pooling points, and the resulting outputs were added to the feature matrix of the input module in depth before convolution. In the neck section, a spatial pyramid structure was employed to propagate strong semantic features from top to bottom, while the path aggregation network propagated robust displacement features from bottom to top. The fusion of these two mechanisms enhanced the capability to extract feature information.

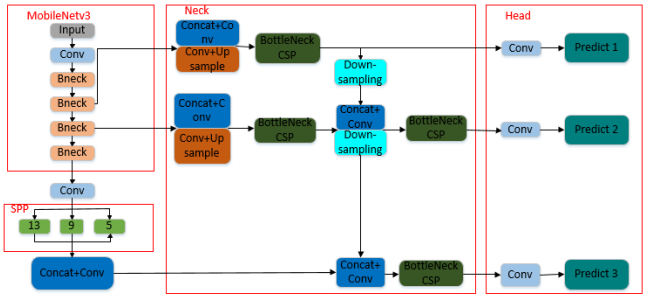


Fig. 3. Improved YOLOv5 network structure diagram.

In the improved YOLOv5, after being processed through MobileNetv3, the obtained feature matrix was initially processed through a 1x1 convolution. This step helped in reducing feature dimensions, thereby lowering computational costs and model complexity. Subsequently, the feature matrix, post 1x1 convolution, was input into the pyramid spatial module, which performed down-sampling at three parallel max-pooling points. This process aided in extracting features at different scales and enhanced the model's multi-scale perception of targets.

The down-sampled results were then added in depth to the feature matrix of the input module before undergoing convolution. This method of depth addition facilitated the fusion of features from different levels, thereby enhancing the model's representational capacity. A spatial pyramid structure was employed in the neck section to propagate strong semantic features from top to bottom. Simultaneously, the path aggregation network propagated robust displacement features from bottom to top. Through the combined use of this structure, feature information was adequately extracted and fused, thereby improving the accuracy and robustness of object detection. This design enabled the model to better understand and accurately detect and locate targets, resulting in more stable and reliable detection results in various complex scenarios for YOLOv5.

IV. Experiment Validation and Comparison

A. Experiment Environment and Dataset

The experiment was conducted on a Windows 10 system with an Intel i7-11700 CPU running at 2.50GHz and an NVIDIA GeForce RTX 3080Ti GPU, along with 32 GB of RAM. The development environment utilized PyCharm Community 2018.3.5 with Python 3.8 as the interpreter. The experimental data were sourced from the NEU-CLS dataset [15], comprising a total of 1800 steel surface defect images, with 300 images for each of the six defect types, the hyperparameters used in this study were as follows: the initial learning rate was set at 0.01, the cyclic learning rate at 0.2, the number of training epochs at 200, and the weight decay was set at 0.0005.

Performance metrics of the YOLOv5 object detection algorithm are typically validated using three evaluation metrics: precision, recall, and mean Average Precision (mAP).

Precision: Precision is the ratio of true positive data (TP) correctly classified by the classifier to all data classified as positive by the classifier (TP + false positive (FP)). The specific calculation method is as shown in Eq. (3):

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

where, TP represents the number of true positive samples predicted as positive, and FP represents the number of negative samples falsely predicted as positive.

Recall: Recall refers to the ratio of true positive data (TP) correctly classified by the classifier to all data classified as positive by the classifier (TP + false negative (FN)). The specific calculation method is as shown in Eq. (4):

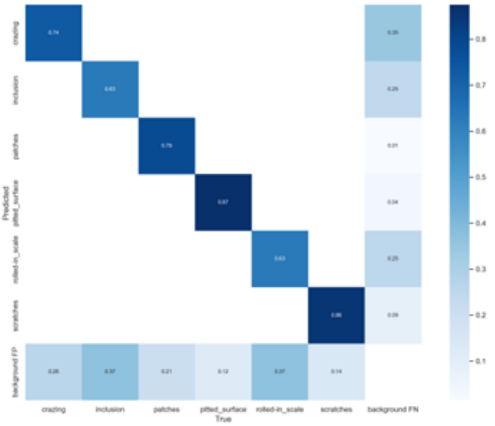
$$Recall = \frac{TP}{TP+FN} \quad (4)$$

where, FN represents the number of positive samples falsely predicted as negative.

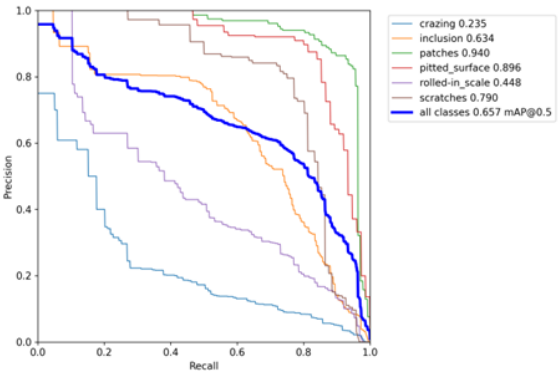
MAP (mean Average Precision): mAP indicates the average precision of the detector on different categories. In object detection tasks, AP (Average Precision) is usually used as the precision metric, and then the average of APs for all categories is calculated to obtain mAP. Here, AP is the area enclosed by the PR (Precision-Recall) curve and the two axes, namely, X and Y.

B. Experimental Results

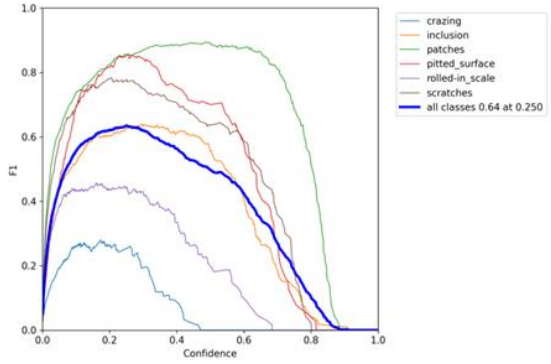
1) Improved YOLOv5-Mobilenet-Small: After training for 200 epochs to obtain the optimal weights, the results on the test set were as follows: mAP@0.5 was 0.657, and F1 score was 0.640. The results are shown in Fig. 4.



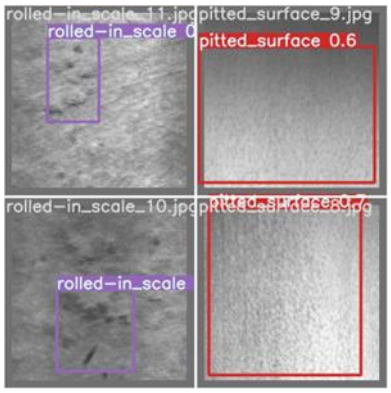
(a) Confusion Matrix



(b) PR Curve



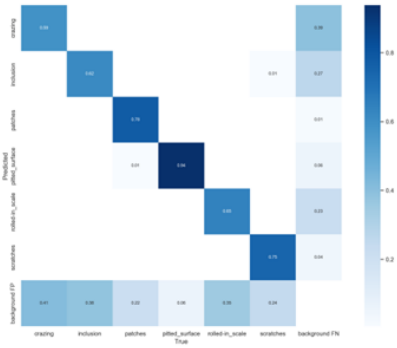
(c) F1 Score Graph



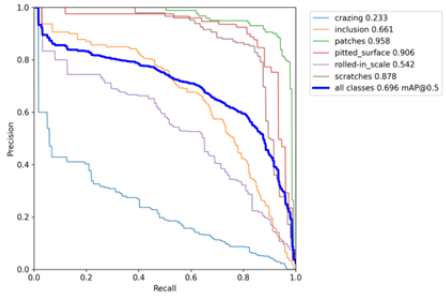
(d) Validation Detection Image

Fig. 4. Results of YOLOv5-mobilenet-small.

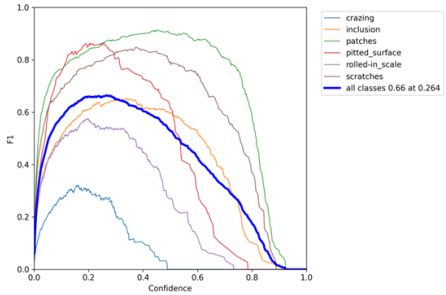
2) Improved YOLOv5-Mobilenet-Large: After training for 200 epochs to obtain the optimal weights, the results on the test set were as follows: mAP@0.5 was 0.696, and F1 score was 0.660. The results are shown in Fig. 5.



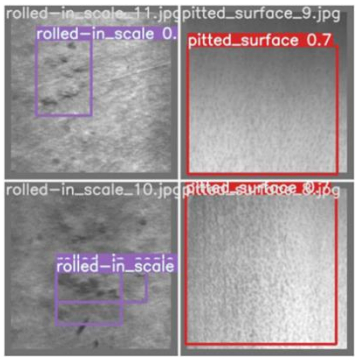
(a) Confusion Matrix



(b) PR Curve



(c) F1 Score Graph



(d) Validation Detection Image

Fig. 5. Results of YOLOv5-Mobilenet-Large.

TABLE I. COMPARISON OF RESULTS FOR DIFFERENT NETWORK ARCHITECTURES

	Precision	Recall	mAP@.5	parameters	Inference Time（ms）	PT File Size (MB)
Yolov5	0.689	0.671	0.682	20873139	3.9	40.1
Yolov5-Mobilenet-Small	0.665	0.631	0.657	5160643	3.2	10
Yolov5-Mobilenet-Large	0.667	0.676	0.696	5899785	3.5	11.5

In Table I, the detection results of YOLOv5, YOLOv5-Mobilenet-Small, and YOLOv5-Mobilenet-Large were compared. From the experimental results, it was observed that the improved network YOLOv5-Mobilenet-Small showed slight decreases (not exceeding 0.04) in Precision, Recall, and mAP@0.5, while significantly reducing inference time and decreasing the size of the .pt file. Similarly, the enhanced network YOLOv5-Mobilenet-Large exhibited slight decreases (not exceeding 0.03) in Precision but slight improvements in Recall and mAP@0.5 compared to YOLOv5. Additionally, it also reduced inference time significantly and considerably decreased the size of the .pt file. These experimental results demonstrated that the improved algorithms achieved the goal of lightweight performance while maintaining detection effect. Notably, YOLOv5-Mobilenet-Small had the fewest parameters, inference time, and model file size, making it suitable for applications with limited computational resources and memory. Conversely, although YOLOv5-Mobilenet-Large required higher computational resources compared to YOLOv5-Mobilenet-Small, it exhibited slight performance improvements and may have been more suitable for tasks requiring higher detection result.

Additionally, we found that the algorithm exhibited variations in performance across different types of defects in the dataset. For some defect types with lower recall values, such as crazing (approximately 0.2), the complexity of their appearance features might have made it difficult for the algorithm to effectively capture and recognize them. In such cases, our algorithm needed further optimization, utilizing more refined feature extraction or improved data augmentation strategies to achieve higher recall.

For defect types with moderate recall values, such as rolled-in scale (approximately 0.5), inclusion (approximately 0.6), and scratches (approximately 0.7), it was evident that the algorithm possessed a certain detection capability for these types, yet there was still room for improvement. We could consider adjusting the model's hyperparameters and refining the anchor box settings to achieve higher recall.

For defect types with high recall values, such as patches and pitted surface (approximately 0.9), the detection performance of the algorithm was quite ideal. This was because their appearance features, such as distinct shapes and contrasts, were more readily captured and recognized by the algorithm.

Overall, the variability in the algorithm's performance across different defect types may have stemmed from the characteristics of the dataset itself, the varying complexity of defect appearances, and the algorithm's differing adaptability to certain specific patterns. In future work, we will continue to optimize the algorithm, striving to enhance its detection capabilities for various defect types and to explore more effective methods for handling complex and diverse defect patterns.

V. CONCLUSION

This study focused on the detection of surface defects in steel materials and proposed an improved steel surface defect detection algorithm based on YOLOv5. The algorithm replaced the backbone network of YOLOv5 with the MobileNet series network, enabling the model to have a more lightweight network structure and higher computational efficiency. In the task of steel surface defect detection, the algorithm's performance was enhanced, allowing for faster defect detection and improved detection effect. Experimental results indicated that by introducing MobileNet, the YOLOv5 architecture improved its performance to some extent, exhibiting clear advantages not only in terms of parameter count, inference time, and model file size but also in enhancing the result of object detection. Among them, the YOLOv5-Mobilenet-Large model slightly outperformed in performance, while the YOLOv5-Mobilenet-Small model showed more efficiency. This is significant for industries such as steel production and quality control, as it promises higher levels of production efficiency and quality assurance.Future work will further optimize the network structure and improve data augmentation strategies, focusing on enhancing the recognition capabilities for these challenging defect types. Additionally, techniques such as model compression will be considered to develop more accurate, versatile, and efficient lightweight defect detection solutions.

ACKNOWLEDGMENT

The authors declare no competing financial interest. This research was funded by Office of Directed Research for Innovation and Value Enhancement (DRIVE) of Mapua University.We also would like to express our sincere gratitude to the editor and anonymous reviewers for their valuable comments, which have greatly improved this paper.

REFERENCES

[1] Song Yubin, Kong Weibin, Chen Xi et al. A review of research on surface defect detection of steel [J/OL]. Software Guide, 1-9 [2024-02-09] http://kns.cnki.net/kcms/detail/42.1671.tp.20240126.0858.002.html.

[2] Zhang Guo, Chen Fei, Wang Jianping, et al. Lightweight PCB surface defect detection algorithm [J/OL]. Journal of Beijing University of Posts and Telecommunications, 1-7 [2024-02-08] https://doi.org/10.13190/j.jbupt.2023-139.

[3] Dong Yongfeng, Sun Songyi, Wang Zhen, et al. Surface defect detection using fusion attention mechanism and joint optimization [J/OL]. Journal of Computer Aided Design and Graphics: 1-10 [2024-02-08] http://kns.cnki.net/kcms/detail/11.2925.tp.20240109.1933.006.html.

[4] Dwivedi D, Babu M S V K, Yemula K P, et al. Identification of surface defects on solar PV panels and wind turbine blades using attention based deep learning model [J] Engineering Applications of Artificial Intelligence, 2024131107836.

[5] Wu Jiling, Jin Yuzhen. Research on surface defect detection of aluminum profiles based on improved Faster R-CNN [J]. Computer Age, 2023 (11): 52-57. DOI: 10.16644/j.cnki.cn33-1094/tp.2023.11.010.

[6] Huang Jiahui, Wu Shilin, Xu Jiawei. Research and application of cone bucket recognition technology based on YOLOv5 [J]. Journal of Wuhan Textile University, 2024,37 (01): 89-93.

[7] Li Chen, Xu Zunyi, Yan Chun, et al. Design and Implementation of Intrusion Detection System Based on Monocular Vision and YOLOv5 Algorithm [J/OL]. Software Guide, 1-6 [2024-02-09] http://kns.cnki.net/kcms/detail/42.1671.TP.20240130.1603.002.html.

[8] Hao Bo, Gu Jiming, Liu Liwei. Target detection based on BF-YOLOv5 infrared and visible light image fusion [J/OL]. Electrooptics and Control, 1-7 [2024-02-09] http://kns.cnki.net/kcms/detail/41.1227.TN.20240130.1653.002.html.

[9] Ma Zairong, Lou Xufeng, Wu Maonian, etc. Design of intelligent glasses for the blind tactile paving based on MobilenetV1 [J]. Internet of Things Technology, 2023,13 (12): 76-80.DOI: 10.16667/j.issn.2095-1302.2023.12.020.

[10] Niu Siqi, Ma Rui, Xu Xiaolin, et al. Research on MobileNetV2 maize seed variety recognition based on improved CBAM attention mechanism [J/OL]. Chinese Journal of Cereals and Oils, 1-12 [2024-02-09] https://doi.org/10.20048/j.cnki.issn.1003-0174.000697.

[11] Zhao Jinfang, Li Quan, Zhao Jinli. Vehicle recognition and tracking based on improved SSD-MobileNetV3 network and SORT [J]. Automation and Instrumentation, 2023, (11): 16-19+24. DOI: 10.14016/j.cnki.1001-9227.2023.11.016.

[12] Xiong Zheng, Che Wengang, Bao Yongli, et al. Improved MobileNetV3 Hot Rolled Steel Strip Surface Defect Classification Algorithm [J]. Journal of Shaanxi University of Technology (Natural Science Edition), 2023,39 (05): 30-37.

[13] Linde Aluminum, Liu Chang, Chen Qi, et al. YOLO Lightweight Object Detection Model Based on Low Rank Decomposition [J/OL]. Locomotive Electric Transmission, 1-7 [2024-02-09] https://doi.org/10.13890/j.issn.1000-128X.2024.01.120.

[14] Qin Zijun, Deng Jun, Chen Kunhao, et al. A fall alarm system based on YOLO object detection [J]. Mechanical and Electrical Engineering Technology, 2024,53 (01): 224-227.

[15] Yanqi Bao, Kechen Song, Jie Liu, Yanyan Wang, Yunhui Yan, Han Yu, Xingjie Li, “Triplet- Graph Reasoning Network for Few-shot Metal Generic Surface Defect Segmentation,” IEEE Transactions on Instrumentation and Measuremente.2021.