DOI: 10.28919/ejma.2024.4.13
Eur. J. Math. Appl. (2024)4:13
URL: http://ejma.euap.org


# STEEL DEFECT DETECTION BASED ON MULTI-ANGLE AND MULTI-FEATURE FUSION NETWORK

FEIHONG YU AND JINSHAN ZHANG*

Abstract. With the rise of deep learning, various industries are adopting deep learning-based object detection algorithms to streamline general engineering operations. Particularly, a single-stage object detection algorithm striking a better balance between accuracy and speed has gained prominence. For the YOLO V5 algorithm, which is widely used in the industrial field, there are some problems in steel defect detection. This paper proposes some improvements to the YOLO V5 algorithm to make up for the shortcomings. Firstly, in the aspect of feature extraction, we propose a new building block for convolutional neural networks (CNN), namely Mres2Net, through the hierarchical feature fusion of single feature and the fusion of residual mechanism, the perception field of the network in the feature extraction part is increased. It makes the network more sensitive to the target. Second, in the aspect of feature fusion, this paper uses cross-fusion of adjacent scale features to fuse features. This method weakens the semantic difference between different scales in the fusion stage, making feature fusion more stable and less conflicting. Experimental results indicate a 6% accuracy improvement over traditional YOLO V5, with enhanced recall stability during training using the proposed method.

## 1. Introduction

Steel is one of the main materials used currently, and its quality directly affects the quality of steel products. Any issues with the raw materials can lead to serious consequences for the final product, especially in housing construction, potentially causing structural instability and even the collapse of the building. The task of detecting steel defects itself involves identifying small targets and various types of defects, making the task inherently challenging. Therefore, choosing the appropriate detection algorithm is particularly important.

At present, the mainstream object detection algorithms are divided into two categories: one is the object detection algorithm based on Transformer, the other is the object detection algorithm based on CNN. Transformer is based on the attention mechanism. Compared with the CNN object detection algorithm, it has a larger perception field, and the weight ratio is more diverse and flexible. Beal, J et al. [1] The Transformer is used as the backbone of the network to extract features, and the encoder performs global feature fusion on the picture. After that, all the outputs of the encoder are reconstructed into spatial features, which achieves good results. Liu et al. [2] proposed a Swin Transformer model using a sliding window, which

College of Mathematics and Statistics, Sichuan University of Science and Engineering, Zigong, 643000, P.R. China

*Corresponding author

*E-mail addresses*: 1318770539@qq.com, zjscdut@163.com.

*Key words and phrases.* deep learning, target detection, feature fusion.

*Received 09/04/2024.*

builds a hierarchical Transformer. This module restricts the attention calculation within the window, increasing the computational efficiency. CNN extracts local features gradually through convolution operation, and accurately extracts image semantic features from shallow features to deep features. Object detection algorithms with CNN as the main body can be divided into two categories. One is a two-stage network represented by Faster R-CNN [3], whose main process is to extract the target area first, and then classify and identify it by CNN. The other is a single-stage network represented by Yolo [4] and SSD [5], which only needs to perform feature extraction once to perform object detection tasks.

Although Transformer has advantages in accuracy, it has disadvantages such as large computation cost, long training time and large amount of training data in actual industrial applications. Considering the actual application, it is not suitable for a large number of production use at present. In the CNN network,the advantage of the two-stage model in the CNN network is high accuracy, but the inference speed is too slow to complete real-time monitoring, so it is not widely used in the production process.Single-stage network achieves a good balance between inference speed and precision in use, so it is widely used in practical industrial operations.

However, in the face of complex environment, it is necessary to transform the network according to practical problems. He, Y et al. [6] proposed a multi-level feature fusion network (MFN). The main task is to combine the hierarchical features extracted by CNN into a single feature, then use the Region Recommendation Network (RPN) to generate the recommended region, and finally a classifier to complete the inference. On the dataset NEU-DET, it has achieved 3.34/50.300 mAP. Zheng et al. [7] proposed a chained atrous spatial pyramid pooling network (CASPPNet) to detect steel defects, which greatly improved the reasoning speed. Guo et al. [8] designed a Transformer-based TRANS module and added it to the backbone and detection joint. With a series of effective improvements, the accuracy was improved compared with the original algorithm. Chen et al. [9] proposed an Extended Feature Pyramid network (EFPN) and a new feature fusion module (FFM) embedded into yolov3 network, which was more accurate and faster than the original network.

In order to compensate for the insensitivity of YOLOv5 in detecting steel defects, this paper proposes a mixed residual module (Mres2Net), inspired by [10]. By segmenting the input features and convolving them separately, we integrate distinct features with their own counterparts, thereby enhancing the network's feature extraction capability and broadening its perception field.

Additionally, we modify the feature fusion process, drawing inspiration from [11]. Initially, we blend and fuse neighboring features and progressively merge them with high-level features, ultimately minimizing the semantic gap between high-level and low-level features. We validate the effectiveness of these alterations on the NEU-DET dataset, noting improvement compared to the original algorithm.

## 2. Related work

2.1. **YOLOV5.** The YOLO series stands as a prominent technology in the field of first-level target detection. In 2020, the fifth generation of YOLO, known as YOLOv5 [12], emerged as a state-of-the-art object detection algorithm based on deep learning. YOLOv5 represents a notable advancement over its predecessor, YOLOv4, resulting in significant improvements in

detection performance. While a direct comparison between the performance of YOLOv5 and YOLOv4 was not explicitly conducted and analyzed, YOLOv5's test results on the COCO dataset demonstrated exceptional performance. Extensive tests have been conducted on commonly employed deep learning techniques, and specific effective methods have been selected to achieve commendable experimental results. Remarkably, when deployed on the Tesla V100, the real-time detection speed of the COCO2017 dataset reaches an impressive 156 FPS, while maintaining an accuracy rate of 56.8% AP.

Currently, YOLOv5 enjoys widespread adoption across diverse application scenarios, including agriculture [13–15], industry [8,16,17], and other industries. For the purpose of this paper, YOLOV5s has been chosen as the fundamental algorithm, considering the delicate balance between target detection accuracy and speed.

The YOLOV5 algorithm is structured into four integral parts, namely the input terminal, backbone network, feature fusion module, and prediction terminal. The network structure diagram is visually depicted in Figure 1. At the input end, the mosaic data enhancement technique is employed, which randomly combines and splices multiple images to improve sample diversity.

For feature extraction, the backbone network utilizes the model architecture of DarkNet-52, incorporating the C3, CBS, and SPPF [18] modules.

The C3 module, a residual module comprising multiple parallel convolutions, works to refine feature representations. The Conv module combines convolution, batch normalization, and activation functions for further feature processing. Meanwhile, the SPP module employs spatial pyramid pooling to capture features with different receptive fields, thereby increasing the model's generalization ability.

The feature fusion module incorporates the pyramid attention structure [19], skillfully fusing features of different scales to enhance the overall representation capability. Finally, at the prediction terminal, the network outputs three groups of information, including categories, positions, and confidence scores of different sizes, culminating in the object detection results.

By leveraging YOLOv5s, this research aims to strike the optimal balance between target detection accuracy and computational efficiency, making it a suitable choice for diverse real-world applications. The utilization of mosaic data enhancement, coupled with the integration of sophisticated modules, allows the algorithm to achieve impressive performance levels in object detection tasks. As deep learning techniques and YOLO series algorithms continue to evolve, there remains promising potential for further improvements in detection capabilities, leading to continued advancements in accuracy and speed across various domains.

### 2.2. **Related improvement.**

2.2.1. *Mres2Net.* In order to increase the overall perception field of the network, many excellent people have proposed different methods to design the network module. These designs are based on increasing the granularity of the perception field of the network while not increasing the network parameters too much.

Zhu et al. [20] proposed a variant DETR, which increases the convergence speed training period of the network as a whole, and the sensitivity to small targets is greatly increased. Li et al. [21] proposed a new perceptual generative adversarial network which can improve the resolution of small targets and reduce the difference between small targets and large targets. Li

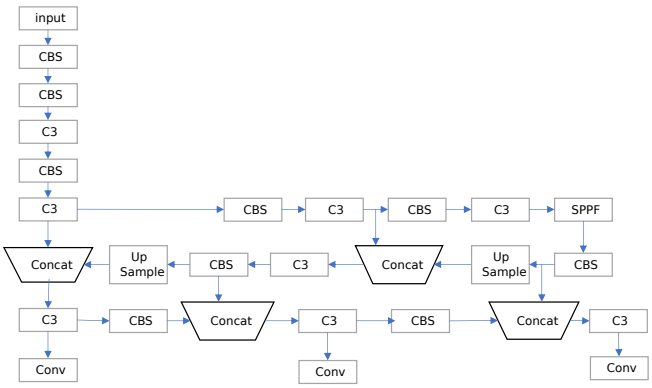


FIGURE 1. yolov5 network structure.

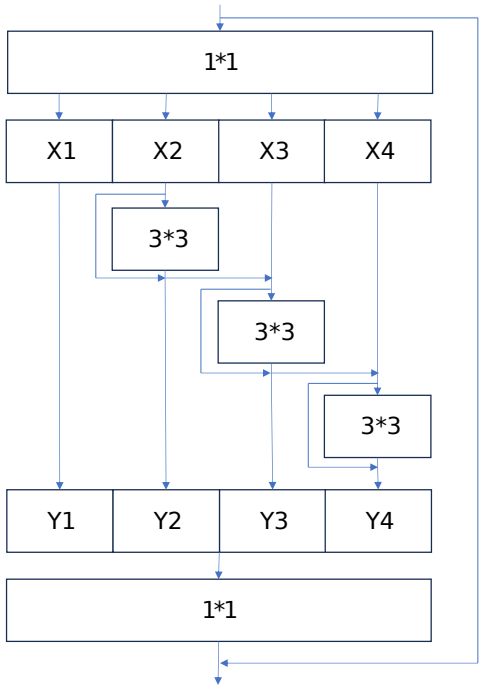


FIGURE 2. Mres2net(the scale dimension s = 4).

et al. [22] proposed WavesNet based on EfficientDet. They widened the channel on the original basis, so that the speed and efficiency of feature fusion were improved. At the same time, they improved the normalization formula to make the features more stable. Liu et al. [23] proposed a feature fusion method for small object detection based on PANet and BiFPN, and introduced the feature extraction part of spatial Pyramid Pooling (SPP) into feature fusion, which effectively improved the algorithm. He et al. [24] proposed an improved algorithm (YOLO-MXANet), which combined channel and spatial attention mechanism to reduce the number of parameters and better extract object features. At the same time, it used Complete Intersection (CIoU) to change the loss function, so that the network could accurately locate small targets.

In this paper, we propose a new multi-scale feature fusion method. Unlike most fusion methods through network features at different scales, the proposed feature fusion method uses residual styles to connect multiple times at a low-fine-grained perspective to increase multiple perception fields. This is shown in Figure 2.

As in Figure 2, we separate the input features into t parts (divided into 4 parts in the figure) after passing them through 1×1 convolution, each of which is denoted using $X_i$, where i∈{1,2,3...s}, In addition to the first separated feature $X_1$, all other features have its own 3×3 convolution (using $F_i()$ representation) to extract features, the subsequent features are required to join the output of the previous feature before passing through its own convolution kernel, which makes the network field of view wider, more sensitive to the target, in addition to joining the input of the previous feature also introduces residual structure, so that in the process of looping to add up the results of each branch, and the network training is more stable to alleviate the gradient explosion and other problems. $X_i$ after the above changes to $Y_i$, so $Y_i$ can be expressed as:

$$Y_i = \begin{cases} X_i & \text{i=1} \\ K_i(X_i) + X_i & \text{i=2} \\ K_i(X_i + Y_{i-1}) + X_i + Y_{i-1} & 2 < \text{i} \leq \text{s} \end{cases} \tag{1}$$

We retained the $X_i$ features in order to reduce the increase in model computation for this improvement, the fusion between individual features increased the extraction of multi-scale information, and the residuals were introduced to ensure the stability of the model. Finally, we re-spliced the separated modules to change the number of channels by 1×1 convolution to facilitate subsequent feature fusion. Because he deals with both multiscale and residual modules, he named it Mres2Net.

2.2.2. *Asymptotic Feature Pyramid Network(AFPN).* Feature fusion is also an important part of the final result. One of them, Huang et al. [25] Proposed a dense convolutional network (DenseNet) that successfully mitigates the gradient explosion due to the network being too deep, as well as enhancing the propagation between features. Fu et al. [26] Combined Residual-101 and Inverse Convolution with each other and applied them to SSD networks, adding rich pre- and post-textual semantics, especially for small targets. Lin et al. [27] Who added lateral connections between different scales to progressively propagate from low-level to high-level features and maintain the accuracy of the semantic information. Chen et al. [28] have proposed the atrous space pyramid pooling (ASPP), which employs filters with different sampling rates to detect features in the incoming feature layer and obtain better contextual features. For steel defective target, we use Asymptotic Feature Pyramid Network (AFPN), the feature fusion network is shown in Figure 3.

This feature fusion method is a feature fusion method of different scales extracted from the Backbone part. We named the different scale features from small to large in turn $\{G_1, G_2, G_3\}$.

AFPN first fuses low-level features ($G_2$ *and* $G_3$) to produce new fusion features, and then fuses the new features with the highest level features. Since the new features have mixed semantics of $G_3$ and $G_2$ at the same time, the new features are fused with the high-level features after fusing the new features, because $G_2$ and $G_1$ belong to adjacent levels of features, which reduces the problem of semantic loss in the fusion process. Of course, in the low-level features, we also use the same method to fuse the high-level features, which greatly increases the perception field of the network at all scales. Finally, we get the prediction results of three different scales $\{P_1, P_2, P_3\}$.

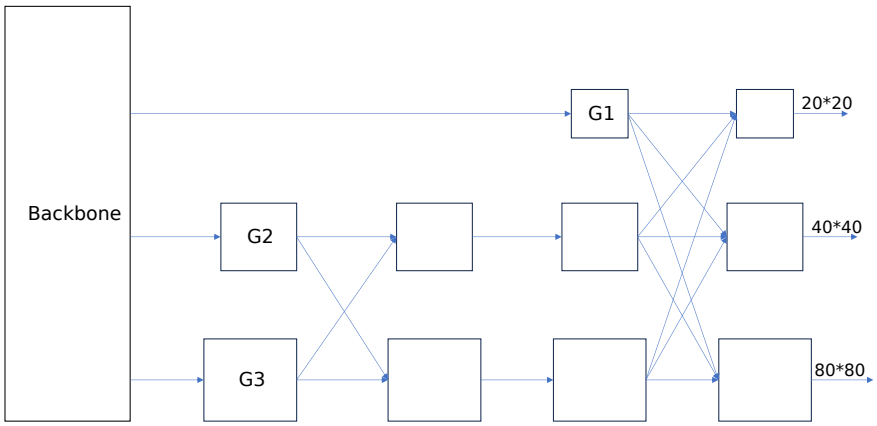


FIGURE 3. Asymptotic Feature Pyramid Network

2.2.3. *Network improvement.* The new network model is shown in Figure 4. We replaced the C3 module in Backbone with the Mres2Net (MR2) module, which makes the original network increase the perception field as much as possible in the process of extracting the network. Because the module itself has a multi-angle residual structure, the network is more stable in the training process. In the subsequent experiments, the number of parameters of the model is also reduced.

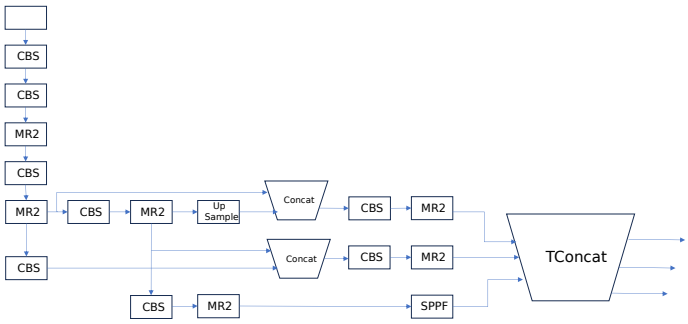


FIGURE 4. Improving Network Structure

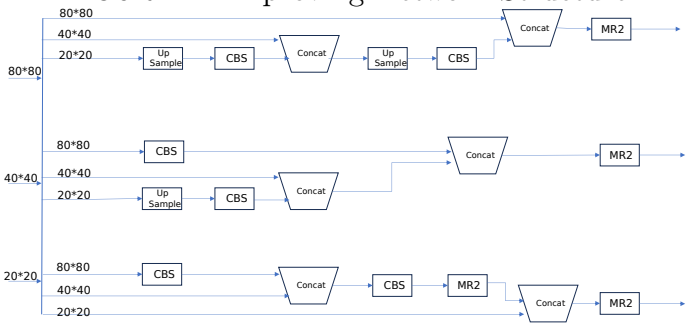


FIGURE 5. TConcat

FIGURE 6. Network improvement: where a is the network structure diagram, CBS,Up Sample, SPPF and Concat in a are consistent with the original yolo network, where MR2 is the Mres2net module introduced above, and the TConcat module is the content shown in Figure 5

Secondly, we improved the feature fusion, as shown in Figure 4. We fused the sixth and eighth layers of Backbone to prepare for the final prediction fusion, and the specific steps are as

follows: We first upsample the output features of the eighth layer and directly fuse them with the sixth layer features, and then change their dimension after convolution operation. Similarly, we change the scale of the sixth layer features after convolution operation and directly fuse them with the eighth layer features before convolution operation and change their dimension. Finally, the two newly generated features and the last layer feature are input into the TConcat structure.

Finally, in TConcat (as illustrated in Figure 5), three scale features are obtained: 80*80, 40*40, and 20*20. To obtain the new 80*80 scale feature, we first fuse the upsampling and convolution of the 20*20 scale with the 40*40 scale feature. Subsequently, we fuse the upsampling and convolution with the 80*80 scale to obtain the latest prediction feature after convolution. For the new 40*40 scale feature, the 20*20 scale feature must first undergo upsampling and then be fused with the 40*40 scale feature prior to convolution, resulting in a fused 80*80 scale feature. The 80*80 feature scale is initially convolved and then fused with the 40*40 feature scale to generate a 20*20 feature scale through fusion. Subsequently, the resulting fusion undergoes convolution and further fusion with the existing 20*20 feature scale before being ultimately convolved.

## 3. Experiment

To assess the efficacy of the proposed approach, we conducted an experiment employing the publicly available NEU-DET dataset [29]. This dataset served as the evaluation benchmark for assessing the performance of new network as well as other contemporary models. The NEU-DET dataset comprises instances of six distinct defect categories: namely, scratches, patches, pitted surfaces, inclusions, crazing, and rolled oxide scales. Each defect category encompasses 300 images, each with a resolution of 200 by 200 pixels, resulting in a total of 1800 grayscale images. To facilitate experimentation, the NEU-DET dataset was partitioned into two subsets, a training set and a test set, with a distribution ratio of 90% for training and 10% for testing. As such, the training set, composed of 1620 samples, was employed for the purpose of optimizing network parameters through the minimization of the loss function.

3.1. **Experimental environment and parameter configuration.** The experiment used NVIDIA T1000 graphics card with 8G memory, Windows 10 operating system, and Pytorch deep learning framework. The model is trained through 100 epochs with an initial learning rate of 0.01.

3.2. **Evaluation Index.** The performance evaluation of the proposed network was conducted comprehensively using several key metrics, including the mean average precision (mAP), recall (Recall), floating-point operations (FLOPs), parameters (Params), and frames per second (FPS). In the context of target detection tasks, the evaluation of network performance heavily relies on crucial indicators, specifically accuracy and recall rate. These indicators play a significant role in assessing the recognition effectiveness of the network, and their definitions are provided below:

$$Precision = TP/(TP + FP) \tag{2}$$

$$Recall = TP/(TP + FN) \tag{3}$$

The terms True Positives (TP), False Positives (FP), and False Negatives (FN) are elucidated as follows:

- True Positive (TP): This term refers to a situation in which a prediction made by the model is accurate and aligns with the actual condition of the sample. In other words, the model correctly identifies a positive instance as positive.
- False Positive (FP): This term denotes a scenario where the model's prediction is incorrect, yet it incorrectly indicates a positive result for the given sample. In essence, the model incorrectly identifies a negative instance as positive.
- False Negative (FN): This expression signifies a situation in which the model's prediction is inaccurate, causing it to mistakenly classify a positive example as negative. In simple terms, the model fails to identify a positive instance correctly.

The metric mAP50:95 encapsulates the mean average precision across varying Intersection over Union (IoU) thresholds, spanning from 0.5 to 0.95 with an incremental step of 0.05 (0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95). This extended range offers a more comprehensive assessment of model performance. Consequently, mAP50:95 was employed in lieu of mAP50 for evaluating the efficacy of our model. Furthermore, to facilitate the comparison of computational complexities among different networks, we opted to utilize the computational time complexity (FLOPs) and computational space complexity (Params, denoting the number of parameters). These metrics were employed to discern disparities between distinct methods.

3.3. **Ablation Experiment.** We validated the efficacy of the proposed improvement in this study through ablation experiments, as presented in Table 1. Upon replacing the feature fusion module with AFPN, a substantial enhancement was observed in precision (P), while recall (R) remained largely unchanged. Consequently, there was an approximate 6% increase in mean average precision (mAP) and a 1% increment in mAP50:95. However, we encountered issues related to recall accuracy instability during training. To address this concern, we introduced the M2R module which resulted in a significant boost in recall value (R), albeit at the expense of decreased precision (P). As a result, there was an additional 1% improvement in mAP and a 0.2% increase in mAP50:95 scores while ensuring training stability.

The algorithm detailed in this paper effectively accomplishes the task of recognizing both the defect's category and its corresponding location. Notably, it has been observed that YOLOv5 can encounter issues of undetected instances. The enhancement introduced by this method is anticipated to enhance the detection performance to a notable degree, resulting in improved accuracy of detection and more precise localization.

Table 1. Ablation experiments (NEU-DET dataset).

| Number | AFPN | M2R | P | R | mAP | mAP50:95 |
|---|---|---|---|---|---|---|
| 1 | | | 59.8 | 73.8 | 77.8 | 40.1 |
| 2 | ✓ | | 73 | 73 | 82.5 | 41.2 |
| 3 | ✓ | ✓ | 70 | 82 | 83.8 | 41.4 |

3.4. **Advanced Model Comparison.** In order to validate the efficacy of our promotion in detecting defects on strip surfaces, we conducted a comparative analysis against several contemporary models. These models encompass YOLOv3, YOLOv3-tiny, Faster R-CNN, SSD, RetinaNet, and YOLOv5 networks. Through this comparison, we aimed to assess the performance of our approach within the realm of strip surface defect detection.

TABLE 2. Ablation experiments (NEU-DET dataset).

| Methed | mAP | FPS |
|---|---|---|
| YOLOv3 | 70.3 | 61.4 |
| YOLOv3-tiny | 57 | 163 |
| Faster R-CNN | 80.2 | 10 |
| SSD | 71.6 | 70.2 |
| RetinaNet | 70.2 | 48.2 |
| YOLOv5 | 77.8 | 52 |
| ours | 83.8 | 41.6 |

Analysis of Table II reveals that in comparison with the accuracy-centric two-stage architecture, Faster R-CNN, our approach exhibits an accuracy improvement exceeding 3%, alongside a frame rate per second (FPS) surpassing 31.6. When juxtaposed with one-stage algorithms, encompassing YOLOv3, YOLOv3-Tiny, Faster R-CNN, SSD, RetinaNet, and YOLOv5, our proposed algorithm demonstrates varying degrees of superiority in terms of accuracy. However, there exists a slight trade-off in terms of FPS, attributable to heightened network complexity and an expanded overall perceptual field within the network. This augmentation results in an elevated computational workload. Nevertheless, given the context that the algorithm's operational environment places greater emphasis on accuracy and imposes stringent accuracy requirements while maintaining modest FPS prerequisites, our proposed algorithm emerges as the superior choice within this domain.

## 4. CONCLUSIONS

This paper addresses the practical challenge of steel defect detection by enhancing the YOLOv5 object detection algorithm. We introduce a novel and adaptable convolution module named M2R, engineered to deliver robust multi-scale feature extraction capabilities without significantly escalating computational demands. This augmentation contributes to a discernible network performance enhancement and fosters greater stability in the training process, owing to the incorporation of a distinctive multi-angle residual mechanism.

Additionally, we introduce an innovative feature fusion approach, employing a progressive fusion methodology to bridge semantic discrepancies across diverse networks. Simultaneously, this approach optimizes information amalgamation at each prediction scale, effectively expanding the network's perceptual field. This methodology yields commendable results, as demonstrated by the achieved mAP on the NEU-DET dataset. Our approach outperforms YOLOv3, YOLOv3-Tiny, Faster R-CNN, SSD, RetinaNet, and YOLOv5 in terms of accuracy; however, there is room for improvement in terms of FPS.

Future endeavors will concentrate on refining the algorithm further, with the aim of attaining heightened accuracy, swifter detection speeds, and decreased model complexity.

**Acknowledgments.** The research was partially supported by the Opening Project of Sichuan Province University Key Laboratory of Bridge Nondestruction Detecting and Engineering Computing (2022QYY06). The authors thank the anonymous reviewers for their helpful suggestions.

## References

[1] J. Beal, E. Kim, E. Tzeng, D. Park, A. Zhai, D. Kislyuk, Toward transformer-based object detection, arXiv:2012.09958 [cs.CV], (2020).

[2] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: IEEE/CVF International Conference on Computer Vision (ICCV), (2021), 9992-10002.

[3] S. Ren, K. He, R.B. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2015), 1137-1149.

[4] J. Redmon, S.K. Divvala, R.B. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2016), 779-788.

[5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S.E. Reed, C. Fu, A.C. Berg, SSD: Single shot multibox detector, European Conference on Computer Vision, (2015).

[6] Y. He, K. Song, Q. Meng, Y. Yan, An end-to-end steel surface defect detection approach via fusing multiple hierarchical features, IEEE Trans. Instrument. Measure. 69 (2020), 1493-1504.

[7] Z. Zheng, Y. Hu, Y. Zhang, H. Yang, Y. Qiao, Z. Qu, Y. Huang, CASPPNet: A chained atrous spatial pyramid pooling network for steel defect detection, Measure. Sci. Technol. 33 (2022), 085403.

[8] Z. Guo, C. Wang, G. Yang, Z. Huang, G. Li, MSFT-YOLO: Improved YOLOv5 based on transformer for detecting defects of steel surface, Sensors, 22 (2022), 3467.

[9] X. Chen, J. Lv, Y. Fang, S. Du, Online detection of surface defects based on Improved YOLOV3, Sensors, 22 (2022), 817.

[10] S. Gao, M. Cheng, K. Zhao, X. Zhang, M. Yang, P.H. Torr, Res2Net: A new multi-scale backbone architecture, IEEE Trans. Pattern Anal. Mach. Intell. 43 (2019), 652-662.

[11] G. Yang, J. Lei, Z. Zhu, S. Cheng, Z. Feng, R. Liang, AFPN: Asymptotic feature pyramid network for object detection, arXiv:2306.15988 [cs.CV], (2023).

[12] X. Zhu, S. Lyu, X. Wang, Q. Zhao, TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios, in: IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), (2021), 2778-2788.

[13] J. Qi, X. Liu, K. Liu, F. Xu, H. Guo, X. Tian, M. Li, Z. Bao, Y. Li, An improved YOLOv5 model based on visual attention mechanism: Application to recognition of tomato virus disease, Comput. Electron. Agric. 194 (2022), 106780.

[14] I.A. Ahmad, Y. Yang, Y. Yue, C. Ye, M. Hassan, X. Cheng, Y. Wu, Y. Zhang, Deep learning based detector YOLOv5 for identifying insect pests, Appl. Sci. 12 (2022), 10167.

[15] H. Wang, S. Zhang, S. Zhao, Q. Wang, D. Li, R. Zhao, Real-time detection and tracking of fish abnormal behavior based on improved YOLOV5 and SiamRPN++, Comput. Electron. Agric. 192 (2021), 106512.

[16] X. Zhu, S. Lyu, X. Wang, Q. Zhao, TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios, in: IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), (2021), 2778-2788.

[17] W. Liu, K. Quijano, M.M. Crawford, YOLOv5-tassel: Detecting tassels in RGB UAV imagery with improved YOLOv5 based on transfer learning, IEEE J. Selected Top. Appl. Earth Observ. Remote Sens. 15 (2022), 8085-8094.

[18] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (2014), 1904-1916.
[19] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, (2018), 8759-8768.
[20] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable DETR: Deformable transformers for end-to-end object detection, arXiv:2010.04159 [cs.CV], (2020).
[21] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, S. Yan, Perceptual generative adversarial networks for small object detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2017), 1951-1959.
[22] Z. Li, J. Zhao, X. Lu, Small object detection based on WavesNET, in: 10th International Conference on Information Systems and Computing Technology (ISCTech), (2022), 287-294.
[23] H. Liu, F. Sun, J. Gu, L. Deng, SF-YOLOv5: A lightweight small object detection algorithm based on improved feature fusion mode, Sensors, 22 (2022), 5817.
[24] X. He, R. Cheng, Z. Zheng, Z. Wang, Small object detection in traffic scenes based on YOLO-MXANet, Sensors, 21 (2021), 7422.
[25] G. Huang, Z. Liu, K.Q. Weinberger, Densely connected convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2017), 2261-2269.
[26] C. Fu, W. Liu, A. Ranga, A. Tyagi, A.C. Berg, DSSD: Deconvolutional single shot detector, arXiv:1701.06659 [cs.CV], (2017).
[27] T. Lin, P. Dollár, R.B. Girshick, K. He, B. Hariharan, S.J. Belongie, Feature pyramid networks for object detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2017), 936-944.
[28] L. Chen, G. Papandreou, I. Kokkinos, K.P. Murphy, A.L. Yuille, DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (2017), 834-848.
[29] Y. Bao, K. Song, J. Liu, Y. Wang, Y. Yan, H. Yu, X. Li, Triplet-graph reasoning network for few-shot metal generic surface defect segmentation, IEEE Trans. Instrument. Measure. 70 (2021), 1-11.