Mathematical Biosciences and Engineering

MBE, 20(11): 19858–19870.
DOI: 10.3934/mbe.2023879
Received: 11 August 2023
Revised: 28 September 2023
Accepted: 25 October 2023
Published: 01 November 2023

http://www.aimspress.com/journal/MBE

Research article

Surface defect detection of steel based on improved YOLOv5 algorithm

Yiwen Jiang*

School of Intelligent Equipment, Changzhou College of Information Technology, Changzhou 213164, China

* **Correspondence:** Email: ginger_yw@yeah.net.

Abstract: To address the challenge of achieving a balance between efficiency and performance in steel surface defect detection, this paper presents a novel algorithm that enhances the YOLOv5 defect detection model. The enhancement process begins by employing the *K-means*++ algorithm to fine-tune the location of the prior anchor boxes, improving the matching process. Subsequently, the loss function is transitioned from generalized intersection over union (GIOU) to efficient intersection over union (EIOU) to mitigate the former's degeneration issues. To minimize information loss, Carafe upsampling replaces traditional upsampling techniques. Lastly, the squeeze and excitation networks (SE-Net) module is incorporated to augment the model's sensitivity to channel features. Experimental evaluations conducted on a public defect dataset reveal that the proposed method elevates the mean average precision (mAP) by seven percentage points compared to the original YOLOv5 model, achieving an mAP of 83.3%. Furthermore, our model's size is significantly reduced compared to other advanced algorithms, while maintaining a processing speed of 47 frames per second. This performance demonstrates the effectiveness of the proposed enhancements in improving both accuracy and efficiency in defect detection.

Keywords: defect detection; YOLOv5; SE-Net; EIOU; carafe upsampling

1. Introduction

Steel production is vital to industrialized nations and used extensively in aerospace, automotive and defense. The production quality directly affects safety. Recently, the demand for higher steel quality has grown, requiring not only performance standards but also excellent surface quality. Surface defects arising from environmental factors, raw materials and technology often reduce steel's wear resistance, corrosion resistance and fatigue strength. If not detected, these defects can pose risks in actual use. Early methods employed manual inspection, where operators used visual inspection to judge whether products were qualified. This method is characterized by its high costs, intermittent defect detection and challenges in data collection. With the increase in steel output, manual methods have become uneconomical. In recent years, scholars have conducted a lot of research on automated identification methods for steel surface defect detection [1–3]. As computer technology advanced, machine vision-based surface defect detection gained acceptance and widespread use in enterprises. Traditional object detection methods identify based on the color, texture and edge features of the detected object. For example, Wang et al. [4] proposed an automatic defect detection method based on template matching. By pixel-by-pixel detection and guiding a series of operations between the template and the sorted test image, defects can be accurately located. Nand et al. [5] proposed an entropy-based defect detection algorithm. By comparing the entropy of the image with the entropy of the background image through background subtraction, the defect part of the image was extracted from the entropy image and three types of defects such as water droplets, bubbles and scratches on the steel surface were successfully detected. Hu [6] introduced a novel approach for defect detection in textured surfaces by utilizing an optimized elliptical Gabor filter, which can be tuned by a genetic algorithm to match the texture features of a defect-free template. This method significantly enhances defect detection efficiency and effectiveness, as demonstrated by extensive experimental results.

On the other hand, deep learning technology has evolved significantly. Compared to traditional image features, it offers superior capabilities in extracting global features and contextual information from images. For example, Xu et al. [7] proposed a deep learning-based method to identify defects in subgrade profile images. The method is based on the Faster R-CNN framework [8] and it combined feature cascading and data augmentation improvement strategies according to the characteristics of subgrade defects, which improved the recognition accuracy. Abu et al. [9] evaluated four deep transfer methods including ResNet [10], VGG [11] and MobileNet [12], verifying the effectiveness of using the MobileNet model in the application of steel surface defect detection. He et al. [13] generated feature maps through CNNs for steel plate defect detection applications and used multilevel feature fusion networks to combine multiple levels of features into one feature, achieving improved defect detection performance. However, a notable limitation of the aforementioned methods is their inability to process in real time. Currently, the application of deep learning technology to defect detection still has shortcomings, such as similar defect features to the background, insufficient training samples and difficulty in actual model deployment. Some scholars have introduced the YOLO series algorithm [14–17] into surface defect detection applications, achieving excellent accuracy. However, directly applying these methods to steel surface defect detection cannot achieve satisfactory results, mainly because the detection system has high requirements for the features of different application defect images and recognition speed.

This paper proposes an enhanced YOLOv5 network [18]. This improved model is designed to be compact enough for embedded deployment while also boosting defect detection performance. The main contributions of the work are summarized as follows:

1) The *K-means*++ clustering algorithm [19] is used to more reasonably select the initial clustering centroid, and obtain suitable bounding boxes on the training dataset;

2) The existing loss function GIOU [20] in YOLOv5 is replaced with EIOU [21], effectively solving the degradation problem of GIOU and making the convergence speed faster;

3) The Carafe operator [22] is incorporated into the model, enhancing its feature extraction capability through the integration of an effective channel attention mechanism.

2. Proposed method

2.1. YOLOv5s network framework enhancement

YOLOv5 is a single-stage object detection algorithm released in June 2020. Building upon YOLOv4 [23], the algorithm introduces enhancements that significantly boost both speed and accuracy. YOLOv5 fundamentally retains the CSPDarkNet53 from YOLOv4 for feature extraction, while integrating added features for performance optimization. The architecture of YOLOv5 is systematically organized into four main components: Input, Backbone, Neck, and Prediction. In the Input section, techniques such as data augmentation, adaptive anchor box operations, and image scaling are employed to preprocess the input dataset. The Backbone network incorporates both Focus and CSP structures, with the Focus module designed to expand the local receptive field range and enhance network speed by segmenting the input image. The CSP structure includes two specific designs: CSP1_X and CSP2_X. These designs enhance the network's learning capability, ensuring accuracy while reducing operations. By splitting the original input into two parallel branches, each subjected to convolution operations, CSP enables the model to capture a richer set of features. Within this backbone, the CBL module, which stands for Conv + BN + LeakyReLU, plays a crucial role. This module combines a convolutional layer, batch normalization, and a leaky rectified linear unit (LeakyReLU) activation function to ensure efficient feature extraction and propagation through the network. The Neck section utilizes a combined feature pyramid network (FPN) + path aggregation network (PAN) structure to harmonize features across different levels, with FPN layers transmitting semantic features in a top-down manner, and PAN integrating low-level and high-level features through up and down sampling operations. Finally, the Prediction segment encompasses bounding box loss and non-maximum suppression, key elements in the object detection process.

In this study, we optimized the YOLOv5s network structure specifically for steel plate surface defect detection. We integrated an effective channel attention mechanism, SE-Net [24], into the backbone network. Additionally, we implemented the EIOU loss function to enhance performance. Additionally, the K-means++ clustering algorithm has been utilized to refine the selection of initial clustering centers, thus determining suitable initial anchor box positions. Lastly, we replaced conventional up-sample operations with the more efficient Carafe operator. A schematic representation of the refined YOLOv5s network structure is depicted in Figure 1.

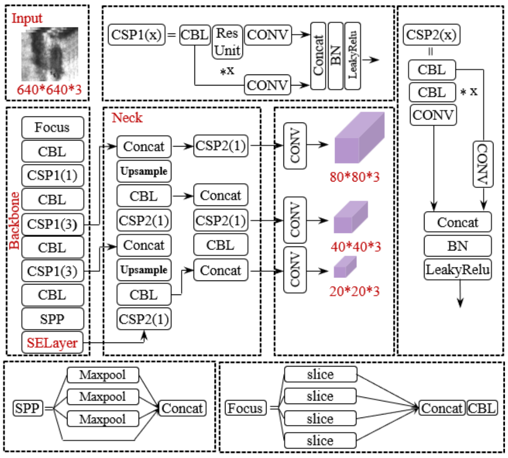

Figure 1. Structure of improved YOLOv5s.

2.2. K-means++ for anchor box optimization

While the original YOLOv5 uses the K-means method [25] for clustering anchor box positions, K-means has inherent limitations. First, being a heuristic method, K-means doesn’t ensure convergence to a global optimum. Second, the initial center selection directly influences the clustering outcome. As K-means randomly selects sample points as cluster centers, it can easily cause local convergence or require more iterations. In contrast, this paper employs the K-means++ algorithm, which refines the initial clustering center selection, ensuring optimal anchor boxes for steel plate defect detection. Differing from K-means, K-means++ treats the initial point selection as a probabilistic task. This conversion not only facilitates the acquisition of a more favorable initial clustering center but also accelerates the algorithm's convergence. The superiority of *K-means*++ over *K-means* lies in its deterministic initialization, which reduces the risk of poor convergence and ensures a more consistent and efficient clustering process. The specific steps of *K-means*++ are as follows:

1) Randomly choose a center x from dataset X;
2) For each data point not yet selected, calculate its Euclidean distance D(x) to the center position;
3) Randomly choose a new data point as the new center using a weighted probability distribution, where the probability of each point being selected, P(x), is calculated using the following formula:

$$P_{(x)} = \frac{{D_{(x)}}^2}{\sum_{x \in X} {D_{(x)}}^2} \tag{1}$$

4) Repeat steps 2 and 3 until k center points have been filtered out;
5) After obtaining the initial centers, continue using the standard *K-means* clustering.

2.3. Loss function optimization

Standard YOLOv5 configurations use the GIOU loss function to measure bounding box prediction discrepancies. GIOU is mathematically represented as:

$$GIOU = IOU - \frac{C-U}{C} , \tag{2}$$

where C is the smallest bounding box that encloses both the predicted and ground-truth bounding boxes, and U is the union of the predicted and ground-truth bounding boxes. However, GIOU has limitations, especially with bounding boxes of varied aspect ratios or misalignments, causing suboptimal training convergence. In these scenarios, GIOU reverts to a basic intersection over union (IOU) metric, which inadequately captures the true extent of overlap between the bounding boxes. To address GIOU’s limitations and degeneration issues, we introduce the EIOU loss function. EIOU provides stable gradients and handles misalignments and aspect ratio variations better. Unlike GIOU, which struggles with bounding boxes that are significantly different in aspect ratios or are misaligned, the EIOU algorithm disaggregates the aspect ratio of both the predicted and actual bounding boxes, thereby enabling independent calculations for width and height, and providing a more nuanced and accurate representation of the overlap between bounding boxes. The EIOU loss function is mathematically formulated as:

$$EIOU = (1 - IOU) + \alpha \cdot d_c + \beta \cdot (d_w + d_h). \tag{3}$$

In this equation, IOU quantifies the overlap between predicted and ground-truth bounding boxes, calculated as the intersection-to-union area ratio. The coefficient α functions as a modulatory weighting factor for the center distance loss, d_c, which quantifies the Euclidean distance between the centroids of the prognosticated and ground-truth bounding boxes. Analogously, β acts as a weighting coefficient for the dimensional loss terms, d_w and d_h, which denote the discrepancies in width and height between the prognosticated and the minimally circumscribed bounding boxes, respectively. By breaking down the aspect ratio, EIOU accelerates convergence and enhances regression accuracy for predicted bounding boxes. This ensures stable learning across varied object shapes and sizes, overcoming GIOU's inherent limitations.

2.4. Integration of visual attention mechanism module

Detecting defects on steel surfaces requires robust deep feature extraction. However, the challenge lies in the high similarity between images of steel surface defects and background images, coupled with significant variations among images of the same type of defects. Such similarities and variations complicate recognition. Recognizing the potential of the attention module to guide the network model in extracting defect features within the feature space [26], this study incorporates the efficient attention mechanism of SE-NET to augment the network's feature extraction capabilities. This approach has demonstrated promising results. Within the enhanced YOLOv5 network, the SE-NET module is strategically embedded in the feature fusion area located in the Backbone, as depicted in Figure 2. The module boosts the network's representational ability by modeling interdependencies between convolutional feature channels. In the SE-NET attention module, input features are sequentially processed through global average pooling on a channel-by-channel basis, followed by two fully connected layers, culminating in Sigmoid nonlinearity to determine the weights of each channel. The two fully connected layers capture cross-channel nonlinear interactions, aiding in dimension reduction.

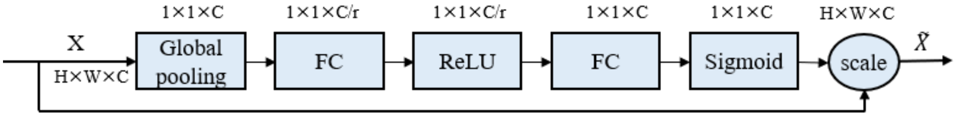

Figure 2. SE-Net module structure.

2.5. Carafe operator for feature enhancement

The YOLOv5's upsample operation uses standard interpolation, focusing on the spatial information of the input feature map and overlooking its semantic information. In surface defect detection, this can cause information loss, blurring, limited receptive fields, and decreased efficiency. To address these challenges, this paper introduces the Carafe operator. It's a versatile upsampling operator skilled at predicting and adjusting upsampling kernels based on the input feature map, enhancing the upsampling's efficacy. The workflow diagram of the Carafe operator is shown in Figure 3.

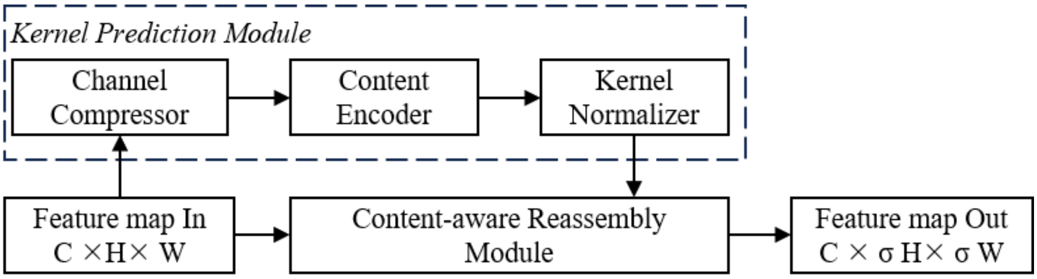

Figure 3. The overall framework of CARAFE.

Carafe consists of two primary modules: The upsampling kernel prediction module and the feature reorganization module. Given an input feature map of dimensions$C \times H \times W$ and an upsampling ratio of σ, the upsampling kernel prediction module first predicts the upsampling kernel. Subsequently, the feature reorganization module performs the upsampling, resulting in an output feature map of dimensions $C \times \sigma H \times \sigma W$.

To elaborate, the input feature map undergoes a 1 × 1 convolution to compress its channels. This mainly reduces the computational load for subsequent processes. After channel compression, a convolutional layer predicts the upsampling kernel for the modified input feature map. This kernel is then expanded in the spatial dimension to derive the final upsampling kernel. Each kernel channel is normalized with the softmax function, ensuring its weights total one. For every position in the output feature map, its corresponding location in the input feature map is identified. A central fixed-size region is extracted from this location, and a dot product with the predicted upsampling kernel for that position is computed to determine the output value.

The Carafe operator stands out with its lightweight design and adaptability to features of different content and scales. It possesses the ability to direct the formation of the upsampling kernel, guided by the semantic information inherent in the input feature map, thereby expanding the receptive field and enhancing the overall quality of the upsampling procedure.

3. Experiments and analysis

3.1. Dataset and environment

In our study, we utilized the NEU-DET [27] dataset from Northeastern University, featuring hot-rolled strip steel surface defects. This dataset collects six types of typical hot-rolled strip steel surface defects, including cracks (Cr), inclusions (In), patches (Pa), pitting (Ps), rolling scrap (Rs), and scratches (Sc). There are 300 samples for each type of defect, and the original resolution of each image is 200 × 200 pixels. This dataset presents two main challenges: 1) The defects within the same class have significant differences in appearance; 2) The defects between different classes have similar aspects, and due to the influence of lighting and material changes, the grayscale of images between classes will also change. Some examples of defect image samples are shown in Figure 4.

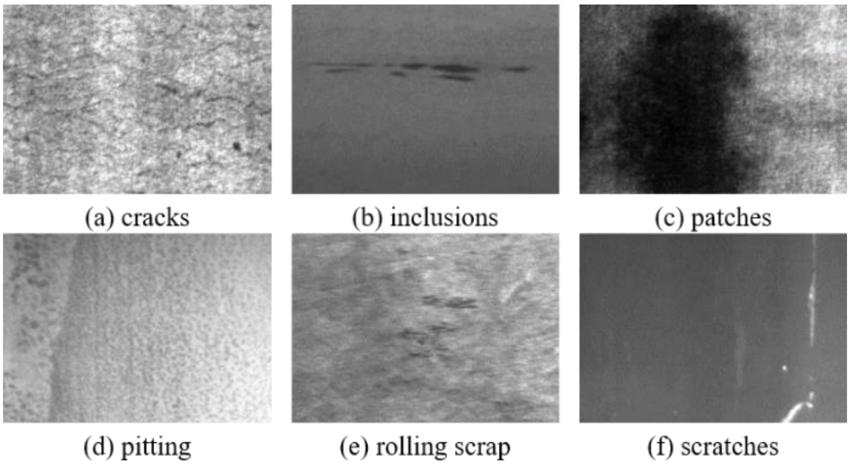

Figure 4. Sample defect images.

Given YOLOv5's fixed input size, we adopted the data expansion method from literature [28]. We resized the original image to 800 × 800, then cropped 640 × 640 sub-images centered on defects, padding edges with gray. Next, we slid the scaled defect area in all directions, maintaining a 30% pixel overlap between frames. We then augmented the data using image rotation, translation, brightness adjustment, and scaling. The final training dataset and test dataset consisted of 12,600 and 5,400 images, respectively. The initial learning rate is 0.01, and we use the cosine annealing strategy to reduce the learning rate. The batch size is set to 16, and a total of 30,000 iterations are trained. The experiment training software environment is Linux4.15.0-142-generic Ubuntu 18.04, the YOLOv5.5 version. The hardware includes an Intel quad-core i7-7700HQ, 2.80GHz CPU, 16GB memory and NVIDIA GeForce GTX1080Ti (11GB).

3.2. Qualitative results analysis

In this study, we selected three representative object detection methods, namely, Faster R-CNN, Single Shot MultiBox Detector (SSD) [29] and YOLOv4, for comparative analysis with the method proposed herein on the dataset. Faster R-CNN achieves superior detection using a two-stage network and region proposal network. SSD achieves efficient object detection by predicting multiple bounding boxes and class scores in a single pass. YOLOv4 optimizes various aspects including data processing, network training, activation and loss functions. The comparative visualization of recognition effects between our method and others is depicted in Figure 5.

As illustrated in Figure 5, the method introduced in this paper excels in accurately detecting defects of various types and sizes, consistently identifying all defects in the images and significantly surpassing YOLOv4. In contrast, SSD struggled to detect patches and scratches, showing weak localization for larger targets. Faster R-CNN outperformed the SSD algorithm in object detection, securing the highest recognition accuracy for patch targets. Both SSD and YOLOv4 had issues detecting cracks, often misidentifying or missing pitting and scratches.

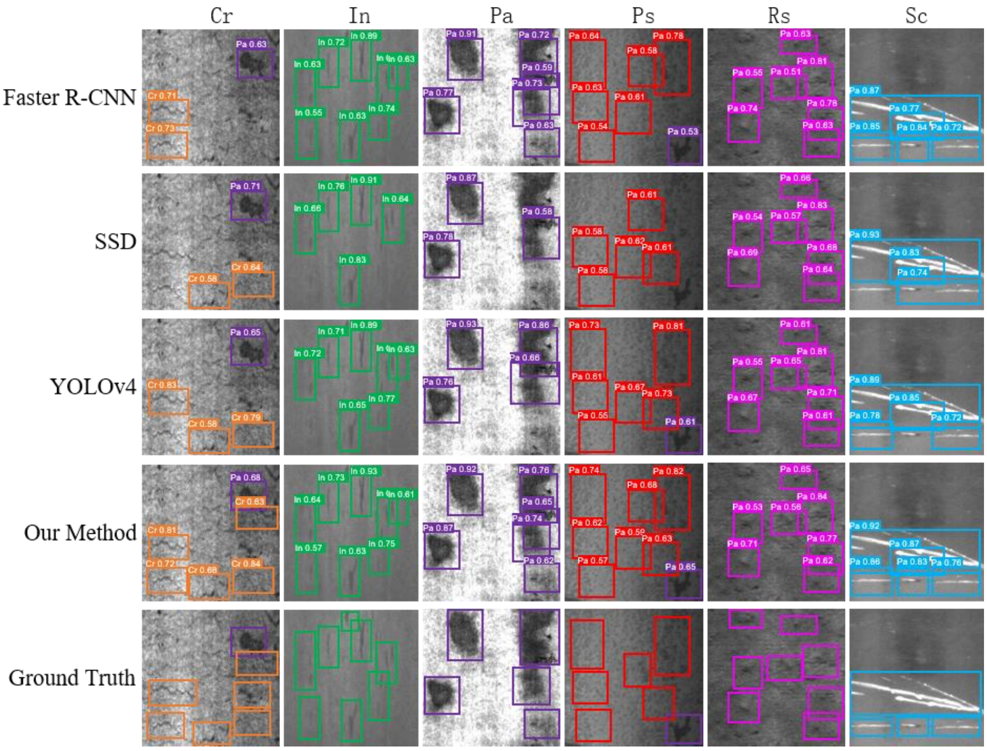

Figure 5. Sample defect images.

3.3. Quantitative result analysis

The evaluations were carried out to verify the effectiveness of the proposed model. The main indicators are selected: Precision (P), recall (R), average precision (AP), and mean average precision (mAP). The corresponding equations are as follows:

$$P = \frac{T_p}{T_p+F_P} \times 100\% \tag{4}$$

$$R = \frac{T_p}{T_p+F_N} \times 100\% \tag{5}$$

$$AP = \int_0^1 P(r)dr \,. \tag{6}$$

where T_p denotes the quantity of defects accurately identified by the detection model, F_P signifies the count of incorrect or unrecognized defects and F_N designates the number of falsely detected targets. The terms P and R correspond to the precision and recall, respectively. The average precision (AP) is defined as the integral of the precision rate with respect to the recall rate. The mean average precision (mAP), representing the average of AP across all categories, serves as a comprehensive metric for assessing the performance of the entire model. To substantiate the efficacy of the proposed method, ablation experiments were conducted with the objective evaluation and corresponding results delineated in Table 1.

Table 1. Ablation experiment.

Metric	Test set	mAP@.5(%)	Up(%)
YOLOv5s	540	73.63	0
+ Data Augmentation	5,400	76.30	2.67
+ K-means++	5,400	77.62	1.32
+ EIOU	5,400	79.17	1.55
+ Carafe	5,400	80.07	0.9
+ SE-Net	5,400	83.30	3.23

The experimental data reveals that the enhanced detection model possesses a more expansive receptive field range, yielding superior results on the dataset when contrasted with the original YOLOv5 model. Incorporating additional data augmentation operations resulted in a 2.67 percentage point increase in mAP. Using the K-means++ algorithm to adjust the initial anchor box position led to a 1.32 percentage point increase in mAP detection precision. Switching from the GIOU to the EIOU loss function improved regression accuracy and resulted in a 1.55 percentage point increase in mAP. By integrating the Carafe operator, the model's feature extraction capability was enhanced. This led to a 0.9 percentage point increase in mAP, underscoring its effectiveness. Incorporating the SE-Net attention mechanism improved the model's image data processing ability, though it slightly increased model complexity.

This study has implemented four key improvements to the original YOLOv5 algorithm model, aimed at enhancing defect detection. First, the *K-means*++ clustering algorithm was employed to ascertain the anchor box size specific to the defect dataset, thereby elevating the model's precision in object defect localization. Second, the EIOU loss function was introduced as a replacement for the original GIOU, a strategic move designed to harmonize negative and positive samples. Third, the integration of the Carafe operator into the model served to refine the feature extraction process, adapting to the semantic information of the input feature map, thereby contributing to the overall enhancement of the model's performance. Lastly, the integration of the SE-Net module into the optimized backbone network served to amplify the efficiency of image feature information extraction.

We also incorporated MSFT-YOLO [30] for comparative experiments. MSFT-YOLO is an advanced model specifically tailored for detecting defects on steel surfaces, leveraging a combination of CNNs and the transformer architecture. This unique blend allows the model to capture both local and global features, significantly enhancing its ability to discern defects from their surrounding backgrounds. Furthermore, its structure is an improvement over the YOLOv5 one-stage detector, optimizing real-time detection capabilities. A comparative analysis of the efficiency and model size between the improved algorithm model and other methods on the dataset is detailed in Table 2.

Table 2. Comparison of efficiency and size indicators.

Method	Precision(%)	Recall(%)	mAP@.5(%)	FPS(f/s)	Parm(MB)
Faster R-CNN	80.1	77.5	80.0	12	572
SSD	65.4	63.2	67.6	25	237
YOLOv4	73.7	69.3	72.3	33	265
YOLOv5s	78.3	72.5	76.3	52	42
MSFT-YOLO	79.3	73.9	77.0	30	86
Our method	82.2	79.6	83.3	47	78

Through a comparative analysis of the detection speed among various defect detection algorithms, it becomes evident that the enhanced YOLOv5 model exhibits a distinct advantage in detection speed over both SSD and YOLOv4. The MSFT-YOLO, tailored specifically for defect detection, achieves an mAP of 77.0% and operates at 30 frames per second (FPS), showcasing its balance between speed and accuracy. Concurrently, the method attains an mAP value of 83.3% on the dataset, marking an improvement of nearly 11 percentage points compared to YOLOv4. Given that the original YOLOv5's mAP accuracy stands at 76.3%, this substantial increase underscores the efficacy of the network feature fusion implemented in our approach. Additionally, the detection outcomes reveal that, while employing ResNet-101 as the backbone network ensures a commendable level of recognition accuracy within the Faster R-CNN network model, it does so at the expense of processing speed due to the high parameter requirements. The comparative performance results of the different algorithms are detailed in Table 3. The data therein illustrates that the integration of optimization strategies, such as initial anchor box positioning and loss function adjustment, enables the refined model to utilize image feature information more comprehensively. Furthermore, the incorporation of the SE-Net attention mechanism module into the backbone network ensures the model's real-time operation without a significant increase in model parameters.

Table 3. Comparison of performance indicators for different detection algorithms.

Method	mAP @.5(%)					
	Cr	In	Pa	Ps	Rs	Sc
Faster R-CNN	51.1	86.2	91.7	89.2	70.2	93.4
SSD	36.4	73.3	83.2	80.7	52.3	79.6
YOLOv4	39.9	77.6	90.1	84.8	54.1	87.5
YOLOv5s	41.3	78.2	91.4	85.7	56.8	88.3
MSFT-YOLO	38.9	85.2	91.0	84.9	70.3	92.1
Our method	56.5	88.6	90.4	92.5	77.3	94.3

4. Conclusions and future prospects

This study presented an optimized YOLOv5 model specifically designed for detecting defects on steel surfaces. Key enhancements to the model encompassed the use of the K-means++ method for optimal anchor box positioning and the adoption of the EIOU loss function in place of the traditional GIOU to elevate network performance. Additionally, we integrated the Carafe operator for better semantic information adaptation and feature extraction, and embedded the SE-Net attention module within the backbone network to further enhance feature extraction capabilities. Compared to other state of the art object detection techniques, our enhanced model achieved a notable mAP of 83.3%, underscoring its effectiveness. Beyond steel defect detection, the model's versatility allowed it to be applied in areas like recognizing the maturity of agricultural products and detecting diseases and pests. In future research, we aim to enlarge our defect sample dataset for training and to refine the network to enhance its defect detection range.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by Natural Science Foundation of Changzhou College of Information Technology (CXZK202106Y) and the Key Laboratory of Industrial Internet of Things Foundation of Changzhou College of Information Technology (KYPT201803Z).

Conflict of interest

The author declares there is no conflict of interest.

References

1. Q. Luo, X. Fang, L. Liu, C. Yang, Y. Sun, Automated visual defect detection for flat steel surface: A survey, *IEEE Trans. Instrum. Meas.*, **69** (2020), 626–644. https://doi.org/10.1109/TIM.2019.2963555
2. R. Mordia, A. K. Verma, Visual techniques for defects detection in steel products: A comparative study, *Eng. Failure Anal.*, **134** (2022), 106047. https://doi.org/10.1016/j.engfailanal.2022.106047
3. B. Tang, L. Chen, W. Sun, Z. Lin, Review of surface defect detection of steel products based on machine vision, *IET Image Process.*, **17** (2023), 303–322. https://doi.org/10.1049/ipr2.12647
4. H. Wang, J. Zhang, Y. Tian, H. Chen, H. Sun, K. Liu, A simple guidance template-based defect detection method for strip steel surfaces, *IEEE Trans. Ind. Inform.*, **15** (2018), 2798–2809. https://doi.org/10.1109/TII.2018.2887145

5. G. K. Nand, Noopur, N. Neogi,, Defect detection of steel surface using entropy segmentation, in *2014 Annual IEEE India Conference (INDICON)*, (2014), 1–6. https://doi.org/10.1109/INDICON.2014.7030439
6. G. H. Hu, Automated defect detection in textured surfaces using optimal elliptical Gabor filters, *Optik*, **126** (2015), 1331–1340. https://doi.org/10.1016/j.ijleo.2015.04.017
7. X. Xu, Y. Lei, F. Yang, Railway subgrade defect automatic recognition method based on improved faster R-CNN, *Sci. Program.*, **2018** (2018). https://doi.org/10.1155/2018/4832972
8. S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, *Adv. Neural Inform. Process. Syst.*, **28** (2015). https://doi.org/10.48550/arXiv.1506.01497
9. M. Abu, A. Amir, Y. H. Lean, N. A. H. Zahri, S. A. Azemi, The performance analysis of transfer learning for steel defect detection by using deep learning, in *Journal of Physics: Conference Series*, **1755** (2021), 012041. https://doi.org/10.1088/1742-6596/1755/1/012041
10. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016), 770–778. https://doi.org/10.1109/CVPR.2016.90
11. S. Tammina, Transfer learning using vgg-16 with deep convolutional neural network for classifying images, *Int. J. Sci. Res. Publ.*, **9** (2019), 143–150. https://doi.org/10.29322/IJSRP.9.10.2019.p9420
12. Y. Li, H. Huang, Q. Xie, L. Yao, Q. Chen, Research on a surface defect detection algorithm based on MobileNet-SSD, *Appl. Sci.*, **8** (2018), 1678. https://doi.org/10.3390/app8091678
13. Y. He, K. Song, Q. Meng, Y. Yan, An end-to-end steel surface defect detection approach via fusing multiple hierarchical features, *IEEE Trans. Instrum. Meas.*, **69** (2019), 1493–1504. https://doi.org/10.1109/TIM.2019.2915404
14. C. Zhao, X. Shu, X. Yan, X. Zuo, F. Zhu, RDD-YOLO: A modified YOLO for detection of steel surface defects, *Measurement*, **214** (2023), 112776. https://doi.org/10.1016/j.measurement.2023.112776
15. B. Zhu, G. Xiao, Y. Zhang, H. Gao, Multi-classification recognition and quantitative characterization of surface defects in belt grinding based on YOLOv7, *Measurement*, **216** (2023), 112937. https://doi.org/10.1016/j.measurement.2023.112937
16. Z. Ma, Y. Li, M. Huang, J. Cheng, S. Tang, A lightweight detector based on attention mechanism for aluminum strip surface defect detection, *Comput. Ind.*, **136** (2022), 103585. https://doi.org/10.1016/j.compind.2022.103585
17. Y. Li, M. Ni, Y. Lu, Insulator defect detection for power grid based on light correction enhancement and YOLOv5 model, *Energy Rep.*, **8** (2022), 807–814. https://doi.org/10.1016/j.egyr.2022.08.007
18. P. Jiang, D. Ergu, F. Liu, Y. Cai, B. Ma, A Review of Yolo algorithm developments, *Proc. Comput. Sci.*, **199** (2022), 1066–1073. https://doi.org/10.1016/j.procs.2022.01.135
19. K. Zhao, Y. Wang, Y. Zuo, C. Zhang, Palletizing robot positioning bolt detection based on improved YOLO-V3, *J. Intell. Robotic Syst.*, **104** (2022), 41. https://doi.org/10.1007/s10846-022-01580-w

20. H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, S. Savarese, Generalized intersection over union: A metric and a loss for bounding box regression, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2019), 658–666. https://doi.org/10.1109/CVPR.2019.00075
21. Y. F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, T. Tan, Focal and efficient IOU loss for accurate bounding box regression, Neurocomputing, **506** (2022), 146–157. https://doi.org/10.1016/j.neucom.2022.07.042
22. J. Wang, K. Chen, R. Xu, Z. Liu, C. C. Loy, D. Lin, Carafe: Content-aware reassembly of features, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2019), 3007–3016. https://doi.org/10.1109/ICCV.2019.00310
23. A. Bochkovskiy, C. Y. Wang, H. Y. M. Liao, Yolov4: Optimal speed and accuracy of object detection, preprint, arXiv: 2004.10934.
24. J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2018), 7132–7141. https://doi.org/10.1109/CVPR.2018.00745
25. M. Ahmed, R. Seraj, S. M. S. Islam, The k-means algorithm: A comprehensive survey and performance evaluation, *Electronics*, **9** (2020), 1295. https://doi.org/10.3390/electronics9081295
26. Z. Niu, G. Zhong, H. Yu, A review on the attention mechanism of deep learning, *Neurocomputing*, 452 (2021), 48–62. https://doi.org/10.1016/j.neucom.2021.03.091
27. K. Song, Y. Yan, A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects, *Appl. Surface Sci.*, **285** (2013), 858–864. https://doi.org/10.1016/j.apsusc.2013.09.002
28. A. Van Etten, You only look twice: Rapid multi-scale object detection in satellite imagery, preprint, arXiv: 1805.09512. https://doi.org/10.48550/arXiv.1805.09512
29. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, et al., Ssd: Single shot multibox detector, in Computer Vision--ECCV 2016: 14th European Conference, (2016), 21–37. https://doi.org/10.1007/978-3-319-46448-0_2
30. 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. https://doi.org/10.3390/s22093467