IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 June 2024, accepted 2 July 2024, date of publication 10 July 2024, date of current version 18 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3426318

RESEARCH ARTICLE

Optimization Algorithm of Steel Surface Defect Detection Based on YOLOv8n-SDEC

XING JIANG, YIHAO CUI, YONGCHENG CUI, RUIKANG XU, JINGQI YANG, AND JISHUAI ZHOU

School of Mechanical and Automotive Engineering, Qingdao University of Technology, Qingdao 266033, China

Corresponding author: Xing Jiang (jiangxing@qdlgdxlyxo.wecom.work)

This study was supported by the Natural Science Foundation of Shandong Province (Grant ZR2021QE289) under the Department of Science and Technology of Shandong Province.

ABSTRACT Considering steel as one of the most widely utilized materials, the detection of defects on its surface has always been a paramount area of research. Traditional target detection algorithms often face challenges such as low detection accuracy, missed and false detections, insufficient feature extraction capabilities, and inadequate feature fusion in tasks related to steel surface defect detection. To address these issues, this study proposes an enhanced algorithm, YOLOv8n-SDEC, utilizing the open-source dataset NEU-DET from Northeastern University as the sample dataset. Initially, the study improves the original SPPF module to the SPPCSPC module, enabling the network to better emphasize the features of the target. Furthermore, to augment the network's feature extraction capability, a fusion with deformable convolution is introduced, enhancing the extraction of features from defective targets. The traditional CIoU loss function is substituted with the EIoU loss function in YOLOv8n aiming to minimize the discrepancies in height and width between predicted boxes and ground truth boxes. This substitution is intended to hasten model convergence and improve localization performance. Lastly, CARAFE is employed to replace the nearest neighbor algorithm, reducing the loss of feature information due to upsampling operations. Experimental outcomes reveal that the accuracy of the enhanced model reaches 76.7%, marking a 3.3% increase over the traditional model. Compared to conventional steel surface defect detection algorithms, the algorithm introduced in this study achieves more precise detection of steel surface defects.

INDEX TERMS YOLOv8n, steel defect detection, SPPCSPC, deformable conv, CARAFE, EIoU.

I. INTRODUCTION

As the most important material in the industrial field, steel has various defects in its manufacturing and usage process, such as Crazing and Inclusion, which not only have a serious impact on the performance and reliability of steel, but can also lead to equipment failures and safety accidents, making exploring an accurate and efficient method for surface defect detection an urgent need in current industrial development.

Major traditional methods for steel defect detection include visual inspection, magnetic leakage detection, and eddy current detection, among which visual inspection is prone to causing visual fatigue for inspectors, magnetic leakage detection is not effective for detecting closed cracks and eddy current detection is greatly affected by the environment and has certain limitations while require manual and precise instrument involvement, resulting in higher detection costs.

With the continuous development of computer vision [1], [2], deep learning [3], artificial intelligence, and other technologies, recent years has witnessed steel defect detection methods based on image processing and pattern recognition gradually becoming a research hotspot. These methods, which analyze the surface images or magnetic images of steel to automatically identify and locate defects, excel in fast detection speed and high accuracy, which can effectively improve the quality control level of steel production lines. Currently, there are two main types of deep learning object detection algorithms, namely, the two-stage object detection algorithm, with the RCNN series [4] as a typical representative, and the one-stage object detection algorithm, with the YOLO series [5], SSD (Single Shot MultiBox Detector) [6], CenterNet [7], etc. as the main representatives.

The associate editor coordinating the review of this manuscript and approving it for publication was Abdullah Iliyasu.

At present, many scholars have applied deep learning object detection algorithms to steel surface defect detection. The advantage of two-stage object detection algorithms now lies in their high accuracy, but they require a large amount of computation and have slower detection speeds. One-stage object detection algorithm can simultaneously obtain the location of the target during the classification process, saving a lot of time compared to two-stage object detection algorithms and making it more likely to meet real-time detection requirements while requiring a large amount of data to support and imposing certain requirements for computational power. Liu et al. [8] proposed a multi-scale context steel defect detection network based on Faster R-CNN. The parallel convolution architecture composed of dilated convolutions was used to capture multi-scale context information. The feature enhancement and selection module could both enhance the discriminability of features and reduce information confusion. In reference [9], a new surface defect detection network based on Mask R-CNN was presented with a novel pyramid designed for multi-scale fusion. A new evaluation metric CIoU (complete intersection over union) was used in the region proposal network to overcome the limitations of IOU (intersection over union) in some special cases, effectively improving the detection accuracy and enabling more accurate defect localization. The two-stage object detection algorithm, although can achieve high accuracy, has high computational complexity and slow detection speed. Therefore, many scholars are making efforts to achieve more efficient steel surface defect detection. In reference [10], a defect recognition system based on convolutional neural networks in the combination with image classification and feature extraction was proposed to achieve better detection accuracy. Moreover, in reference [11], an efficient scale-aware defect detection network based on YOLOv4 was revealed. This model focused on enhancing shallow features that contain rich geometric information to reduce information loss for small targets. Additionally, it introduced a detection head with a dynamic receptive field to alleviate the problem of mismatch between the detection head's receptive field and the target scale. In reference [12], the EFD-YOLOv4 algorithm, which effectively expanded the receptive field using residual connections, was put forward, which, nevertheless, had a high time complexity. In reference [13], the YOLOv5-CD algorithm, which incorporated the Coordinate Attention (CA) mechanism into the backbone network and adopted decoupled head detectors to effectively improve the accuracy of model detection, was shown with dissatisfied real-time performance of model detection. Besides, in reference [14], an improved YOLOv8n algorithm by introducing the GhostNetv2 module, which enhanced the model's expressive power, was constructed with high time complexity. Reference [15] proposes an improved YOLOv4-tiny method for real-time detection of surface defects on strip steel, which is a lightweight target detector based on convolutional neural networks, and although the model size is small and the detection speed is fast, the mAP value on the NEU-DET dataset is 73.29%, which is relatively low in accuracy. Reference [16] proposes an improved model based on YOLOv5s, although the model has a mAP value of 76.6% on the NEU-DET dataset, which is improved by 2.3% compared with the original model, but the FPS has decreased by nearly half compared with the original model, and the model detection speed has decreased more. The steel defect detection environment is more complex, in order to improve the accuracy and robustness of steel defect detection, the reference [17] proposes an energy-based course for gradually adjusting the model to mitigate pseudo-labeling noise due to domain changes. It is used to solve the problem of unsupervised domain adaptation for robust object detection. Reference [18] proposes a large-scale dataset called the Iran Autonomous Driving Dataset (IADD) is presented, aiming to improve the generalization capability of the deep networks outside of their training domains. Reference [19] proposes a Multi-Teacher Knowledge Distillation (MTKD) based approach for training robust semantic segmentation models. Their method significantly improves the performance of student models on different datasets by integrating the knowledge of multiple expert teachers.

For the YOLOv8n algorithm in dealing with the shape of the variable and irregular defective target there is insufficient recognition. The original nearest, SPPF module can not make full use of semantic information, feature fusion is not sufficient. cIoU Loss regression accuracy and stability is insufficient and other problems. The paper uses YOLOv8n as the baseline model to improve it, and the research contents and innovations of this paper are as follows:

1) Replaced the original spatial pyramid pooling – fast (SPPF) structure of YOLOv8n with the spatial pyramid pooling cross stage partial concat (SPPSPC) structure to enhance the expressive power of the network and the perception of defects at different scales.

2) Deformable ConvNets v2(DCNv2) is fused with the penultimate, third, and fourth C2F modules of the Neck layer to enhance the network's ability to learn information about defective targets.

3) In order to address that the nearest algorithm in the Neck layer not fully utilized semantic information, the Content-Aware ReAssembly of Features (CARAFE) [15] module was used to replace the original nearest algorithm, allowing the model to aggregate contextual information within a larger perceptual area.

4) Replaced the original Complete-IoU (CIoU) loss function with the Efficient-IoU (EIoU) [16] loss function to improve the regression accuracy and stability of the network.

The structure of this research work is as follows: Section II introduces the structure of the YOLOv8n algorithm, Section III introduces the modules we referenced, section IV introduces different experiments, and V section summarizes the research methods proposed in this paper and proposes suggestions for further research.

II. YOLOv8 ALGORITHM INTRODUCTION

Release by Ultralytics in January 2023, the YOLOv8 model has its structure similar to the YOLOv5 model also from Ultralytics. To meet different scene requirements, YOLOv8 provides five models: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x, based on the size of the network model. The model weights increase in size in order, and in this study, the YOLOv8n model was selected for training. Considering the need for defect detection and the good real-time performance and accuracy of YOLOv8, the minimal version of YOLOv8n was employed for improvement. The YOLOv8n network structure mainly includes the input end, backbone layer, Neck, and output end Head part.

Input: Used Mosaic data augmentation to execute operations such as splicing on images, whereas traditional data enhancement is utilised to perform operations such as scaling and rotation on images, thereby enabling the model to be more effectively adapted to complex scenes in real-world scenarios. Furthermore, this approach facilitates improvements in the model's robust performance, detection speed and accuracy.

Backbone: Mainly composed of Conv module, C2f module, and SPPF module. Compared to the YOLOv5 algorithm, YOLOv8 improves the CSP Bottleneck with 3 convolutions (C3)to CSP Bottleneck with 2 convolutions(C2f), the structure of the C3 module and C2f module is shown in Figure 1. Compared to the C3 module, the C2f module adopts a multi-branch flow design, providing the model with richer gradient information, enhancing the model's feature extraction capability, and improving the learning efficiency of the network.

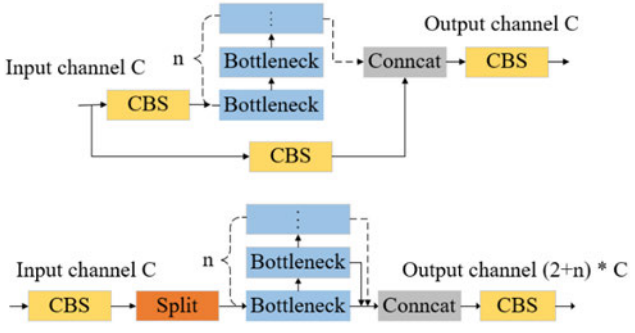


FIGURE 1. Schematic diagram of C3 module and C2 module.

Neck: The neck section adopts the Path Aggregation Network and Feature Pyramid Networks (PAN-FPN) structure to achieve feature fusion of multiple feature maps with different sizes. The C2f module is also used as the main feature extraction module in its structure. This mechanism can effectively enhance the robustness and generalization ability of the model.

Output terminal: The original coupled header structure has been modified to the popular decoupled header structure, adopting an anchor-free design, which improves the positional accuracy and model generalization ability, making it more flexible.

III. YOLOv8n IMPROVEMENT

In the study, an improved algorithm YOLOv8n-SDEC for surface defect detection of steel materials was proposed with its framework shown in Figure 2.

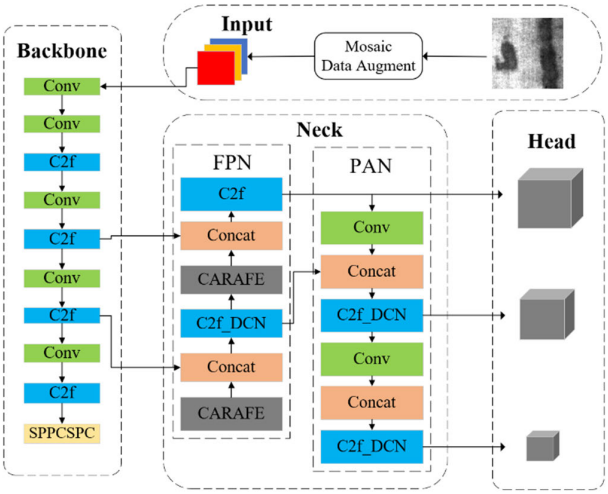


FIGURE 2. YOLOv8n network architecture.

A. LOSS FUNCTION IMPROVEMENT

In the original YOLOv8n network, the CIoU loss function is used to calculate the predicted bounding boxes. CIoU penalizes the distance between the center points and the aspect ratio. The calculation formula is as follows:

$$\upsilon = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{1}$$

$$\upsilon = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{2}$$

$$\alpha = \frac{\upsilon}{1 - IoU + \upsilon} \tag{3}$$

In the formula: w^{gt} and h^{gt} represent the width and height of the ground truth box, while w and h represent the width and height of the predicted box. Meanwhile, $\rho^2\left(b, b^{gt}\right)$ stands for the Euclidean distance between the center points of the predicted box and the ground truth box. IoU represents the intersection over union between the predicted box and the ground truth box. C represents the diagonal length of the minimum bounding rectangle of the predicted box and the ground truth box, α the weight, and υ the parameter that measures the consistency of aspect ratio.

Although CIoU considers the overlap area between the ground truth box and the predicted box, the distance between their center points, and the aspect ratio, its aspect ratio is a relative value, and the overall aspect ratio ignores the differences between the predicted width and height values and their respective ground truth values.

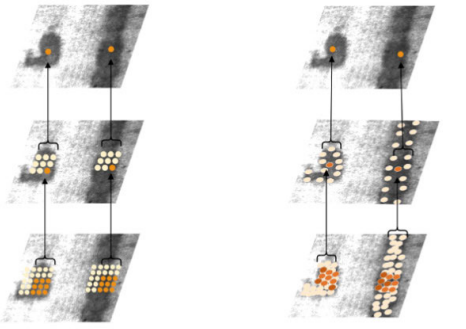
EIoU was chosen as the bounding box loss function for the improved model in the study. The penalty term of EIoU separated the aspect ratio influence factor from CIoU, and calculated the length and width of the target box and the predicted box separately, which accelerated convergence and improves regression accuracy. The calculation formula is shown in equation (4), including three parts: overlap loss, center distance loss, and height-width loss.

$$L_{EIoU} = L_{IoU} + L_{dis} + L_{asp} = 1 - IoU + \frac{\rho^2\left(b, b^{gt}\right)}{c^2} + \frac{\rho^2\left(w, w^{gt}\right)}{c_w^2} + \frac{\rho^2\left(h, h^{gt}\right)}{c_h^2} \quad (4)$$

In the formula: c_w and c_h represent the width and height of the minimum bounding rectangle of the predicted bounding box and the ground truth bounding box, and ρ is the Euclidean distance between two points.

B. DEFORMABLE CONVOLUTION

The traditional convolution of YOLOv8n uses a fixed convolution kernel size. The traditional convolution operation divides the feature map into parts that are the same size as the convolution kernel, and then performs convolution. The position of each part is fixed, but this method cannot better handle surface defects on steel with large geometric shape changes. Inspired by deformable convolution [18], DCNv2 was made compatible with the C2f module in this study. Compared with the traditional convolution operation, the deformable convolution can better adapt to different image contents by learning the deformation parameters and sampling near the current sampling point, so that the feeling field is no longer limited to a single square, but closer to the real shape of the object. Visual example is as shown in Figure 3, where the left side a is normal convolution and the right side b is deformable convolution.



(a) stanndar convolution (b) deformable convolution

FIGURE 3. Standard convolution and deformable convolution.

DCN actually adds an offset during the standard convolution process, allowing deformable convolution to perform different transformations on different targets while increasing its receptive field. The operation process of deformable convolution is shown in Figure 4, where the offsets are obtained by applying separate convolution layers on the same

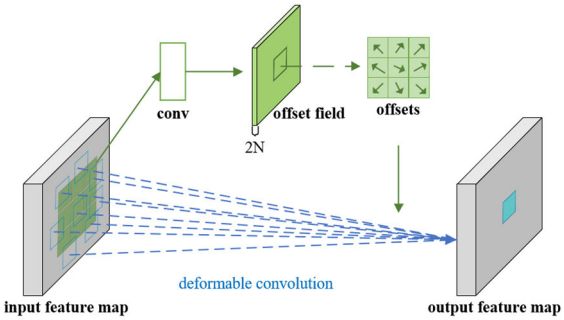


FIGURE 4. Deformable convolutional structure.

input feature map through training. The formula obtained by deformable convolution is:

$$y\left(p_0\right) = \sum_{k=1}^{K} w_k \cdot x\left(p_0 + p_k + \Delta p_k\right) \quad (5)$$

In the formula: x and y are the input and output feature maps, K and k are the total number of sampling points and the sampling position point, Δp_k is the offset corresponding to the kth position, p_0 is the current position of the output feature map, w_k and p_k are the projection weights of the kth sampling point and the kth position of the predefined convolutional network sampling. However, the receptive field of this version of deformable convolution may cause the receptive field to exceed the target range at the corresponding position. Therefore, the deformable convolution DCNv2 compared to DCN was proposed in the study. DCNv2 extended the deformable convolution and enhanced its modeling ability. Meanwhile, a feature simulation scheme was proposed to guide network training. This method introduces weight terms for punishment and sets the weight of uninterested positions to 0. The formula is as follows:

$$y\left(p_0\right) = \sum_{k=1}^{K} w_k \cdot x\left(p_0 + p_k + \Delta p_k\right) \cdot \Delta m_k \quad (6)$$

In the formula: Δm_k is the modulation scalar for the k-th position, which ranges from 0 to 1.

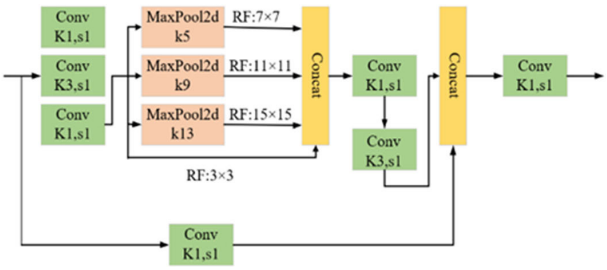


FIGURE 5. SPPCSPC structure.

C. IMPROVED SPATIAL PYRAMID POOLING

SPP (Spatial Pyramid Pooling, SPP), first proposed by Amirkhani et al. [19], obtains different receptive fields by

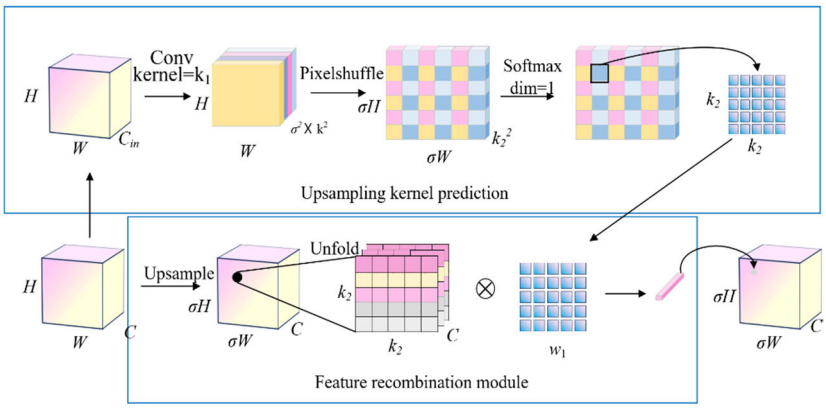


FIGURE 6. CARAFE module.

pooling the feature map at different scales, allowing the algorithm to effectively avoid distortion caused by cropping and scaling operations on image regions. The YOLOv7 algorithm proposes the SPPCSPC structure based on the SPP structure, which draws on the idea of the Cross Stage Partial Network (CSPNet) [20], combining the CSP module with the SPP module. CSPNet introduces local connection operations to partially connect low-level features with high-level features, achieving cross-stage information transmission, which improves the expressive power of network features, captures richer image features, enhances feature representation and receptive field, and improves network performance and robustness. Therefore, the SPPCSPC module was applied in this study instead of the original SPPF module in YOLOv8.

Dqwe The SPPCSPC structure is shown in Figure 5. First, the features were divided into part1 and part2 and the CSP structure was used here. Then, a regular convolution operation was performed on part1, and part2 underwent 1×1, 3×3, and 1×1 convolution operations followed by SPP spatial pyramid pooling operations. The feature maps were first passed through MaxPool layers of size 5×5, 9×9, and 13×13, respectively. Then, these feature maps were concatenated in the channel direction. After that, a 1×1, 3×3 convolution was performed. Finally, the obtained feature map was concatenated with the result of part1 and underwent a 1×1 convolution operation.

D. UPSAMPLING METHOD IMPROVEMENT

In YOLOv8, the nearest upsampling method is used on the Neck network. Although this method has a smaller computational cost, it only uses the grayscale value of the pixel closest to the sampling point as the grayscale value of the current sampling point, without considering the influence of other adjacent pixels. As a result, the quality of the image after sampling will be severely degraded. Therefore, a new upsampling operator CARAFE was employed in the study, which, compared to traditional upsampling modules, further expanded the receptive field of CARAFE, without relying on sub-pixel neighborhoods to operate, but instead integrated information within a larger receptive field, and could dynamically generate adaptive kernels for specific content, enabling better content awareness.

The basic structure is shown in Figure 6. For the input feature map of $X \in R^{C \times H \times W}$, the upsampling ratio is σ, which is an integer. CARAFE generated a new feature map X' with a size of $C \times \sigma H \times \sigma W$. This process includes feature content prediction and feature recombination. In the feature content prediction module, first, a 1×1 convolution was used to compress the channels to reduce computation. Then, a convolution layer with a kernel size of k1 was applied to predict the upsampling kernel. The upsampling kernel size was set to k2. In order to use different upsampling kernels for each position of the output feature map, a tensor $\sigma H \times \sigma W \times k_2^2$ with a shape of should be obtained, corresponding to $\sigma H \times \sigma W$ upsampling kernels. Subsequently, softmax was used to normalize the obtained upsampled kernel, so that the sum of the convolutional kernel weights was 1. For each position in the output feature map, the feature recombination module mapped it back to the input feature map, extracted a k2$\times$ k2 region centered around it, and took the dot product with the predicted upsampling kernel at that point to obtain the output value. Different channels at the same position shared the same upsampling kernel.

IV. EXPERIMENT AND RESULTS ANALYSIS

A. DATASET

In the study, the effectiveness of the improved YOLOv8n algorithm was validated using the steel surface defect dataset (NEU-DET). The dataset is dedicated to the task of hot rolled steel strip surface defect detection. The image type is a high-resolution hot rolled steel strip surface image, and the collection of steel surface defects is conducted in the following way: two LED light sources are symmetrically tilted and mounted on top of the steel surface. The steel to be tested is placed in the centre axis of the two light sources, which are mounted on top of an industrial camera. The camera captures images of the surface defects on the steel, which are then preprocessed to eliminate high-frequency

TABLE 1. Types of defects and their characteristics.

Type	Characteristics
Crazing	It is a small, dense crack-like defect that is typically caused by stress.
Inclusion	The main cause of impurities is coal ash, coal slag, and other non-metallic substances falling onto the surface of the slab and being pressed into the plate during rolling.
Patches	The irregular spots on the surface appear in block or strip shapes, generally due to the mixing of small iron oxide scales during the heating process, or the uneven and unstable cooling liquid during the cooling process.
Pitted surface	The surface appears continuous or locally uneven and rough, mainly caused by oxidation and mechanical damage.
Rolled-in Scale	The surface is uneven and the shapes are varied, mostly brownish-red or black. Generally, improper operation or unreasonable settings result in incomplete removal of iron oxide scale, which is pressed into the surface of the steel during rolling.
Scratches	Usually manifested as bright fine straight lines, mainly caused by abnormal friction between steel and mechanical parts or various human factors during transportation.

noise and perform a grey scale transformation. The dataset comprises six distinct grey-scale maps of steel surface defects, each with a resolution of 200 × 200. A total of 300 sample images are available for each defect type. The six types of defects are Crazing (Cr), Inclusion (In), Patches (Pa), Pitted surface (Ps), Rolled-in Scale (Rs), and Scratches (Sc), as shown in Figure 7. The total number of images in Table 1 is 1,800 and the dataset is randomly divided into 1440 training sets and 360 test sets in an 8:2 ratio.

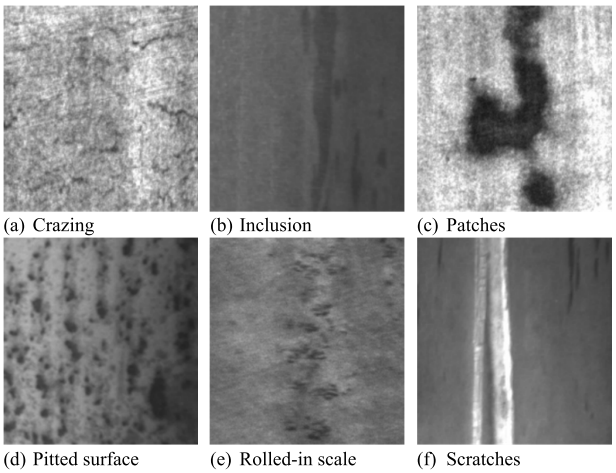


FIGURE 7. Defect category chart.

In order to enhance the detection efficacy, a combination of Mosaic data enhancement and traditional data enhancement is employed to augment the model's generalisation capacity. The principle of Mosaic data enhancement is illustrated in Figure 8. Mosaic data enhancement involves the random selection of four images from a batch. These images are then randomly scaled, cropped, flipped, and subjected to colour gamut changes. Additionally, rows of operations, such as Figure 9 randomly splicing the images into a training sample of a set side length, are performed. This process enriches the dataset and effectively alleviates the problem of insufficient image samples, while also strengthening the robustness of the network.

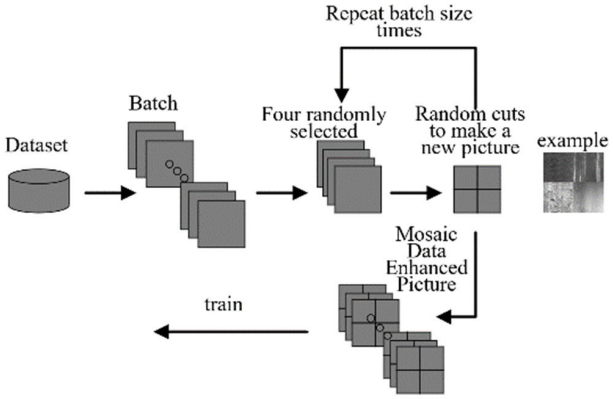


FIGURE 8. Mosaic data enhancement schematic diagram.

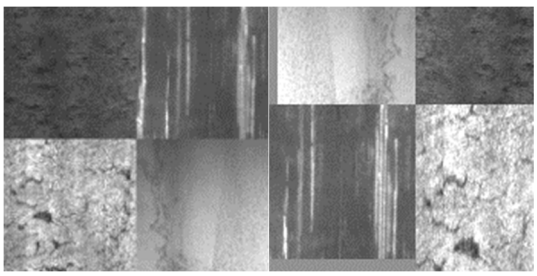


FIGURE 9. Data augmentation.

B. EXPERIMENTAL ENVIRONMENT AND PARAMETER SETTINGS

In the experiment, the Windows 11 operating system, PyCharm was used as the compilation software, equipped

with an AMD Ryzen758008-Core Processor, NVIDIA GeForce RTX 3070 Ti graphics card, Python 3.8 as the development language, and Pytorch-GPU version 1.13 as the deep learning framework, with CUDA version 12.2. Experimental parameter settings are detailed in Table 2.

TABLE 2. Experimental parameter settings.

Parameter	Configuration
Epochs	100
Workers	0
Batch Size	16
Image size	[640,640]
Optimizer	auto

C. EVALUATING INDICATOR

Accuracy, recall rate, average accuracy of all categories, detection speed, and parameter quantity were selected in the study as the performance evaluation indicators for improving the YOLOv8n model. The calculation formula is shown below.

$$p = \frac{T_P}{T_P + F_P} \times 100\% \tag{7}$$

$$p = \frac{T_P}{T_P + F_N} \times 100\% \tag{8}$$

$$AP = \int_0^1 P(R)\, dR \times 100\% \tag{9}$$

$$mAP = \frac{\sum_{i=1}^{n} AP_i}{n} \times 100\% \tag{10}$$

In the formula, T_P refers to the number of true positive samples predicted as positive; F_P refers to the number of negative samples predicted as positive; F_N refers to the number of true positive samples predicted as negative. And n represents the number of data categories in the dataset, in this paper n=6; i is the number of detections; AP is the average precision for a single category; $P(R)$ represents the curve formed by precision and recall.

FPS was used to measure the processing speed of a model. The larger the *FPS* value of the model, the faster the detection speed. The calculation formula is shown in equation (11). Parameters represent the number of parameters occupied by the model, for ordinary convolutional layers, the calculation formula is shown in equation (12).

$$FPS = \frac{Framenum}{ElapsedTime} \tag{11}$$

$$Parameters = (K_h \times K_w \times C_{in}) \times C_{out} + C_{out} \tag{12}$$

In the formula: *Framenum* and *ElapsedTime* represent the total number of images detected and the total time the model runs; *Cin* and *Cout* represent the number of input and output feature map channels, and K_w and K_h represent the width and height of the convolution kernel.

D. MODEL COMPARISON EXPERIMENTS

1) LOSS FUNCTION COMPARISON EXPERIMENT

To verify the performance of introducing the EIoU loss function for defect detection, CIoU, DIoU (Distance-IoU) [21], SIoU (Smoothed-IoU), and WIoU (Wise-IoU) were selected. These commonly used border loss functions were compared in experimental trials, and the results are shown in Table 3. From the experimental results, it can be seen that on the NEU-DET dataset, except for EIoU and WIoU, which significantly improved the mAP value, the other IoU Losses had no significant effect on the model. Among them, DIoU did not consider the aspect ratio of the bounding box during regression, so the mAP value of the model only improved by 0.4%; SIoU only focused on the number of pixels in the defect area and did not consider the shape information of the defect, which may cause false detections when there was a large difference in shape between the predicted result and the true label, so the mAP value of the model only improved by 0.2%; EIoU and WIoU improved the average precision by 1.7% and 1.1% respectively, but WIoU had a slight decrease.

TABLE 3. Comparison of experimental results with different loss functions.

IoU	P	R	MAP	FPS	Parameters
CIoU	0.662	0.699	0.734	385	3006818
DIoU	0.648	0.731	0.738	357	3006818
EIoU	0.705	0.702	0.751	385	3006818
SIoU	0.72	0.661	0.736	370	3006818
WIoU	0.729	0.682	0.745	370	3006818

Comparison of Experimental Results with Different Loss Functions in detection speed compared to the original model, while EIoU had no significant change in detection speed. Therefore, EIoU, which has better overall performance, was selected as the loss function in the study.

2) UPSAMPLING OPTIMIZATION

To address the issue of reduced quality of the upsampled feature maps caused by using nearest neighbor in the YOLOv8nNeck network, it was proposed in the study to use the CARAFE module to achieve higher quality upsampled feature maps. To verify the improvement effect, the effects of CARAFE, Bilinear, and nearest were compared. The experimental results are shown in Table 4, presenting that both CARAFE and Bilinear methods had significantly improved mAP values, with both methods improving by 0.5%. Among them, the Precision of the CARAFE operator increased by 6%, while the Recall decreased by 2.1%. On the other hand, the Precision of the Bilinear algorithm only increased by 4.9%, while the Recall decreased by 2.5%. In the meantime, the detection speed of both slightly decreased.

According to the experimental results, both the Bilinear algorithm and the CARAFE operator have good performance, with the same improvement in mAP value. However, compared with the CARAFE operator, the Bilinear algorithm

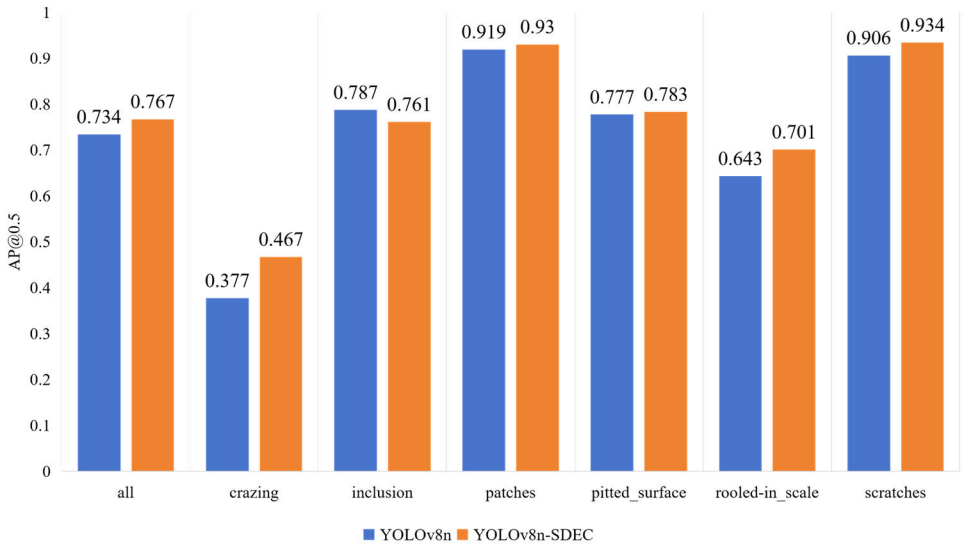


FIGURE 10. Comparison of accuracy before and after improvement.

TABLE 4. Comparison of pyramid pooling effects.

Module	P	R	MAP	FPS	Parameters
Nearest	0.662	0.699	0.734	370	3006818
Carafe	0.722	0.678	0.739	357	3140874
Bilinear	0.711	0.674	0.739	357	3006818

ignores the spatial relationship between features, while the CARAFE operator can better capture the spatial relationship between features. Therefore, the model introduced with the CARAFE operator has higher accuracy. Due to the recombination and weighting operations of features involved in the CARAFE operator, the number of parameters increased. Considering the characteristics of steel defect recognition tasks, the detection speed of the model introduced with the CARAFE operator did not decreased significantly. Therefore, the original nearest upsampling operator of YOLOv8n was replaced with the CARAFE upsampling operator.

3) COMPARISON OF PYRAMID POOLING EFFECTS

SPP is a technique used in CNN networks to handle images of different sizes that can use pooling to fuse feature maps of different scales, establish connections between targets of different scales, and enhance the neural network's ability to detect targets of different scales. In order to better detect defects in steel materials, it was proposed to replace the original SPPF module with the SPPCSPC module.

To test the detection effect, the effects of SPPF, ASPP (Atrous Spatial Pyramid Pooling), SimSPPF (Simplified SPPF), and BasicRFB (Basic Receptive Field Block) were compared. The calculation results are shown in Table 5. From the experimental results, the mAP values of the ASPP, SimSPPF, BasicRFB, and SPPCSPC models all increased varying degrees compared to the baseline model using SPPF. The introduction of ASPP and SPPCSPC had the most obvious improvement effect on the mAP value of the model. Among them, the mAP value of the ASPP model increased by 0.9%, Precision increased by 5.3%, and Recall decreased by 2.4%; the mAP value of the SPPCSPC model increased by 1.9%, Precision increased by 5.3%, and Recall decreased by 1.8%. However, both ASPP and SPPCSPC required the introduction of additional convolutional layers and pooling layers, resulting in a 68.7% increase in the parameter volume of the ASPP model and a 53.4% increase in the SPPCSPC model. Therefore, SPPCSPC, which has the highest increase in mAP value and the smallest increase in model parameters, was chosen.

TABLE 5. Comparison of experimental results using different upsampling methods.

Module	P	R	MAP	FPS	Parameters
SPPF	0.662	0.699	0.734	385	3006818
ASPP	0.715	0.675	0.743	345	5072354
SimSPPF	0.657	0.716	0.739	357	3007202
BasicRFB	0.696	0.681	0.737	357	2918850
SPPCSPC	0.715	0.681	0.753	345	4613858

E. DEFECT DETECTION EFFECT

This paper presents a comparison of the detection accuracies of two YOLOv8n models, one of which has undergone an improvement process. The results of this comparison are shown in Figure 10. As can be seen from the figure,

the average detection accuracy of the improved YOLOv8n model is 76.7%, which is 3.3% higher than the benchmark model. YOLOv8n-SDEC has improved detection accuracy in Crazing, Patches, Pitted_surface, Rolled-in_scale, and Scratches. The most pronounced improvement is observed in Crazing, with an average detection accuracy increase of 9%. The average detection accuracy of the Patches class is also elevated by 1.1%. The average detection accuracy of the Pitted Surface class is improved by 0.6%, while the average detection accuracy of the Rolled-in Scale class has an average detection accuracy improvement of 5.8%. The average detection accuracy of the Scratches class has an average detection accuracy improvement of 2.8%. The Crazing and Pitted Surface detection accuracies are the lowest of all the categories, due to the fact that both of them resemble the surface traces of the steel itself, and both of them suffer from a high false positive rate of manual detection. In summary, the improved model has higher detection accuracy and is more accurate in localising and identifying small-scale targets.

In order to intuitively evaluate the effect of this paper's improved algorithm, respectively, the YOLOv8n algorithm and YOLOv8-SDEC for visual analysis, rotate the picture, change the brightness and other processing methods, respectively, with the YOLOv8n algorithm and YOLOv8- SDEC to detect the same target object, as shown in Figure 11, due to the defects of the background interference is strong, the original model Due to the strong interference of defective background, the original model has serious leakage and misdetection, even Crazing and Rolled-in_scale cannot be detected, and Scratches are all incorrectly recognised as other categories. In this paper, YOLOv8-SDEC is added with the modules of SPPSPC and CARAFE, which can provide a richer representation of the features, and help to detect the targets of smaller sizes, and the situation of leakage detection is significantly improved. Inclusion and Scratches size varies, adding Deformable Conv can adapt to the size change of defects, improve detection accuracy, reduce the probability of leakage and misdetection.

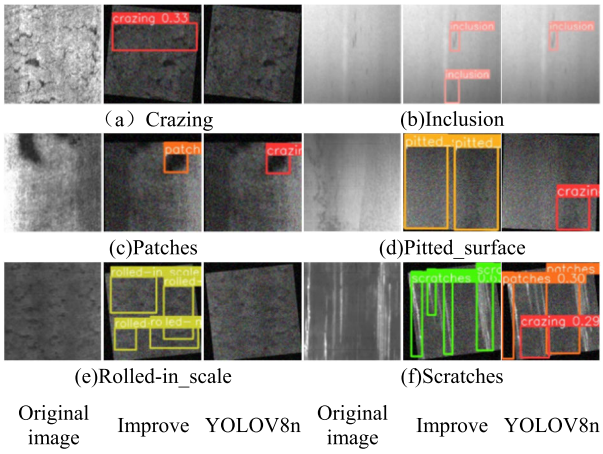
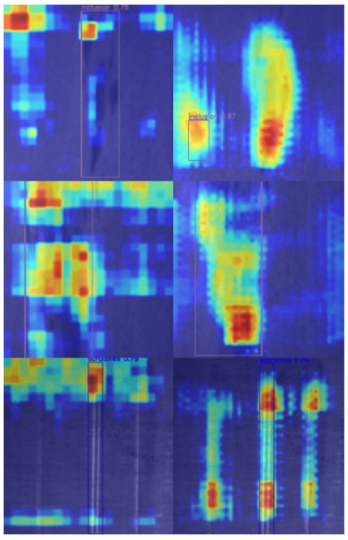


FIGURE 11. Graph of detection results.



(a) Unimproved (b) Improved

FIGURE 12. Heatmap.

Separately make the unimproved YOLOv8n algorithm and the improved algorithm for heat map visualisation and analysis, as shown in Figure 12, the 2 algorithms respectively detect the same target, the red area highlights the area that the model pays more attention to, as can be seen in Figure 12, the improved model, which pays more attention to the area where the target is located, is able to more accurately extract the features, and is more focused and accurate on the scope of attention.

F. ABLATION EXPERIMENT

In order to verify the effectiveness of the improved module, this paper conducted ablation experiments on NEU-DET. A total of 5 groups of experiments were designed, where $\surd$ indicates that the module has been added. The experimental results are shown in Table 6, and Experiment 1 is the result of the original model YOLOv8n. Experiment 2 increased the receptive field without adding too many parameters, resulting in a 1.9% increase in mAP and a 5.3% increase in Precision. Experiment 3 added the C2f-DCNv2 module on the basis of Experiment 2, which can preserve more defect information during the feature fusion process. Although the detection speed has slightly decreased, the mAP, Precision, and Recall all increased, with increases of 0.2%, 0.8%, and 1.4% respectively. In Experiment 4, EIoU was used instead of the original CIoU based on Experiment 3. Without increasing the parameter quantity and reducing the detection speed, the mAP value increased by 0.5%, indicating that EIoU separated the aspect ratio influence factor, calculated the differences in height and width separately, and could improve regression accuracy, thereby enhancing defect detection effectiveness. Experiment 5 introduced the CARAFE operator based on Experiment 4 to capture more detailed information and prevent the loss of feature information for small and occluded targets after multiple downsampling. The mAP value increased by 0.7% and the Recall increased by 1.7%.

TABLE 6. Results of ablation experiment.

Experiment	Sppcspc	c2f-dcn	eiou	carafe	P	R	MAP	FPS	Parameters
1					0.662	0.699	0.734	385	3006818
2	√				0.715	0.681	0.753	345	4613858
3	√	√			0.723	0.695	0.755	323	4723780
4	√	√	√		0.722	0.696	0.76	333	4723780
5	√	√	√	√	0.712	0.713	0.767	303	4857836

Compared to the baseline model, the mAP increased by 3.3%. Although there was a slight decrease in detection speed, it still met the real-time requirements.

G. GENERALITY TEST

In order to verify the scalability and generality of the model, we conduct YOLOv8 and YOLOv8-SDEC comparison tests on the dataset GC10-DET [27], which includes 10 types of surface defects: Punching, Welding Line, Crescent-shaped Crack, Water Spot, Oil Stain, Wire Mark, Inclusion, Rolling Pit, Crease, and Waist Fold; the training and validation sets are divided according to the ratio of 8:2, and the experimental environment and parameters are unchanged, the validation results are shown in Table 7 As shown in the table, except for a slight decrease in the detection speed of YOLOv8n-SDEC, the precision, recall and map50 of YOLOv8n-SDEC are optimal, and compared with the benchmark model, they are improved by 2.1%, 5.6%, and 2.3% respectively, which shows that the YOLOv8n-SDEC algorithm has a certain degree of versatility.

TABLE 7. Comparison of performance testing on GC10-DET.

Experiment	P	R	mAP	FPS
YOLOv8n	0.664	0.573	0.635	370
YOLOv8n-SDEC	0.685	0.629	0.658	303

H. COMPARISON EXPERIMENT WITH MAINSTREAM ALGORITHMS

In order to reflect the effectiveness of the improved method in this paper, the improved algorithm is compared with YOLOv5,YOLOv7,Faster-RCNN.Through Table 8, it can be seen that the YOLOv8n- SDEC algorithm proposed in this paper is only larger than YOLOv5 in terms of the number of parameter in YOLOv8n-SDEC, but is much smaller than YOLOv7 and Faster-RCNN, it has better performance in mAP, Precision, Recall, FPS, 10.5% higher than YOLOv5 in Precision, 4.6% higher than YOLOv7 in Recall, 9.5% higher than YOLOv7 in Map, and much higher than the other algorithms in detection speed, which is capable of meeting the defect detection precision and detection accuracy. meet the requirements of detection accuracy and detection speed for defect detection.

TABLE 8. Comparison results with mainstream algorithms.

Experiment	P	R	MAP	FPS	Parameters
YOLOv5	0.696	0.743	0.746	81.5	1767283
YOLOv7	0.607	0.667	0.672	58.47	37221635
Faster-rcnn	0.634	0.721	0.756	19	41375000
YOLOv8n-SDEC	0.712	0.713	0.767	303	4857836

I. COMPARISON EXPERIMENT WITH MAINSTREAM ALGORITHMS

In comparison to existing steel surface inspection algorithms, the algorithm proposed in this paper exhibits a certain degree of superiority. The detection effect is illustrated in Table 9.

TABLE 9. Horizontal comparison experiment.

Experiment	Baseline	mAP	FPS	parameters
yolov8-SDEC (this paper)	Yolov8	0.767	303	4857836
Reference [16]	Yolov5	0.766	66.667	7679000
Reference [28]	YoloX	0.757	109	14400000

Table 9 shows that the yolov8-SDEC model proposed in this paper achieves a mAP value of 76.7%, which is higher than the reference [16] It is improved by 0.1%. This is an improvement of 0.1% compared to the reference [28] improved by 1% compared to reference [16] and reference [28], the algorithm in this paper is optimised in terms of detection accuracy, detection speed and model size. Taken together, the model proposed in this paper performs better than other mainstream algorithms in terms of comprehensive performance.

V. CONCLUSION

This paper presents an enhanced model based on YOLOv8n designed to tackle the challenges associated with the varied types, forms, and complex backgrounds of surface defects in steel. Initially, the original SPPF module was upgraded to the SPPCSPC module. This modification allowed for more effective pooling operations on feature maps within the neural

network, facilitating the extraction of feature information across different scales and improving the resolution of feature maps, thereby boosting the object detection performance. Furthermore, the model integrated the C2f with deformable convolution modules to bolster the detection capabilities for objects of complex shapes. In an effort to enhance the regression accuracy and stability of the model, the EIoU loss function was employed in lieu of the CIoU loss function used in the baseline model. Additionally, the conventional nearest neighbor upsampling operator was replaced by the CARAFE upsampling operator to mitigate information loss during the feature map upsampling process.

The experimental results show that the improved model based on YOLOv8n proposed in this article has an increase of 3.3% in mAP value compared to the baseline model, although the detection speed decreased by almost 20% from the baseline model, the detection speed was still able to reach 303 frames per second, which is better than most models and meets the terminal push requirements. Although there is a slight increase in parameter count due to the use of more complex convolution and matrix operations, the parameter count of the model is still much smaller than that of fast rcnn, YOLOv7, reference [16] and reference [28].The number of parameters is only 12% of that of Faster RCNN, while the detection speed reaches 15 times that of Faster RCNN. This indicates that the model can maintain reasonable computational efficiency in resource constrained environments. Can meet the real-time requirements in the steel production process.

Although the enhanced algorithm proposed in this paper can effectively enhance the detection accuracy of steel defects, there is still scope for improvement in the accuracy of defects in the crack category and in the lightweighting of the model. Furthermore, given that the steel surface inspection system is susceptible to wear and tear and environmental changes, such as physical wear and tear on sensors such as cameras, Furthermore, changes in temperature, humidity, and the angle of illumination and detection may have a long-term impact on the system's performance. Consequently, subsequent work will focus on implementing adaptive algorithms, regular maintenance and calibration, robust algorithms, an expanded dataset, and a simplified network structure to enhance the model.

REFERENCES

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587.

[2] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[3] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digit. Signal Process.*, vol. 126, Jun. 2022, Art. no. 103514, doi: 10.1016/j.dsp.2022.103514.

[4] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 6154–6162.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 21–37.

[7] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, Korea (South), Oct. 2019, pp. 6568–6577.

[8] R. Liu, M. Huang, Z. Gao, Z. Cao, and P. Cao, "MSC-DNet: An efficient detector with multi-scale context for defect detection on strip steel surface," *Measurement*, vol. 209, Mar. 2023, Art. no. 112467, doi: 10.1016/j.measurement.2023.112467.

[9] H. Wang, M. Li, and Z. Wan, "Rail surface defect detection based on improved mask R-CNN," *Comput. Electr. Eng.*, vol. 102, Sep. 2022, Art. no. 108269, doi: 10.1016/j.compeleceng.2022.108269.

[10] L. Yi, G. Li, and M. Jiang, "An end-to-end steel strip surface defects recognition system based on convolutional neural networks," *Steel Res. Int.*, vol. 88, no. 2, Feb. 2017, Art. no. 1600068, doi: 10.1002/srin.201600068.

[11] X. Yu, W. Lyu, D. Zhou, C. Wang, and W. Xu, "ES-net: Efficient scale-aware network for tiny defect detection," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022, doi: 10.1109/TIM.2022.3168897.

[12] S. Li, F. Kong, R. Wang, T. Luo, and Z. Shi, "EFD-YOLOv4: A steel surface defect detection network with encoder–decoder residual block and feature alignment module," *Measurement*, vol. 220, Oct. 2023, Art. no. 113359, doi: 10.1016/j.measurement.2023.113359.

[13] B. Wang, M. Wang, J. Yang, and H. Luo, "YOLOv5-CD: Strip steel surface defect detection method based on coordinate attention and a decoupled head," *Meas., Sensors*, vol. 30, Dec. 2023, Art. no. 100909, doi: 10.1016/j.measen.2023.100909.

[14] M. Huang and Z. Cai, "Steel surface defect detection based on improved YOLOv8," in *Proc. Int. Conf. Algorithms, High Perform. Comput., Artif. Intell. (AHPCAI)*, Dec. 2023, pp. 1356–1360.

[15] W. Zou and C. Ji, "An improved real-time detection method for steel surface defects based on YOLOv4-tiny," *J. Mech. Sci. Technol.*, vol. 42, no. 6, pp. 883–889, 2023, doi: 10.13433/j.cnki.1003-8728.20230034.

[16] H. J. Xu, "Research on optimization of YOLOv5s algorithm for steel surface defect detection," *Comput. Eng. Appl.*, vol. 60, no. 7, pp. 306–314, 2024.

[17] A. Banitalebi-Dehkordi, A. Amirkhani, and A. Mohammadinasab, "EBCDet: Energy-based curriculum for robust domain adaptive object detection," *IEEE Access*, vol. 11, pp. 77810–77825, 2023, doi: 10.1109/ACCESS.2023.3298369.

[18] A. Khosravian, A. Amirkhani, M. Masih-Tehrani, and A. Yazdanijoo, "Multi-domain autonomous driving dataset: Towards enhancing the generalization of the convolutional neural networks in new environments," *IET Image Process.*, vol. 17, no. 4, pp. 1253–1266, Mar. 2023, doi: 10.1049/ipr2.12710.

[19] A. Amirkhani, A. Khosravian, M. Masih-Tehrani, and H. Kashiani, "Robust semantic segmentation with multi-teacher knowledge distillation," *IEEE Access*, vol. 9, pp. 119049–119066, 2021, doi: 10.1109/ACCESS.2021.3107841.

[20] J. Wang, K. Chen, R. Xu, Z. Liu, C. C. Loy, and D. Lin, "CARAFE: Content-aware reassembly of features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, Korea (South), Oct. 2019, pp. 3007–3016.

[21] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 12993–13000.

[22] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, and W. Zuo, "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," *IEEE Trans. Cybern.*, vol. 52, no. 8, pp. 8574–8586, Aug. 2022, doi: 10.1109/TCYB.2021.3095305.

[23] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 764–773.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015, doi: 10.1109/TPAMI.2015.2389824.

[25] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Seattle, WA, USA, Jun. 2020, pp. 1571–1580.

[26] R. Ortega, N. Monshizadeh, P. Monshizadeh, D. Bazylev, and A. Pyrkin, "Permanent magnet synchronous motors are globally asymptotically stabilizable with PI current control," *Automatica*, vol. 98, pp. 296–301, Dec. 2018, doi: 10.1016/j.automatica.2018.09.031.
[27] X. Lv, F. Duan, J.-J. Jiang, X. Fu, and L. Gan, "Deep metallic surface defect detection: The new benchmark and detection network," *Sensors*, vol. 20, no. 6, p. 1562, Mar. 2020, doi: 10.3390/s20061562.
[28] Y. Liu and S. Jiang, "Research on steel surface defect detection based on improved YOLOX," *Modern Electron. Technol.*, vol. 47, no. 9, pp. 131–138, 2024, doi: 10.16652/j.issn.1004-373x.2024.09.024.

XING JIANG was born in Yantai, Shandong, China, in 2002. He is currently pursuing the bachelor's degree with Qingdao University of Technology. His research interests include deep learning and machine vision. He is a member of the Chinese Association for Artificial Intelligence (CAAI).

YIHAO CUI was born in Zaozhuang, Shandong, China, in 2003. He is currently pursuing the bachelor's degree with Qingdao University of Technology. His research interest includes deep learning-based defect detection on steel surfaces.

YONGCHENG CUI was born in Qingdao, Shandong, China, in 2004. He is currently pursuing the B.S. degree with Qingdao University of Technology. His research interest includes target detection.

RUIKANG XU was born in Dongying, Shandong, China, in 2003. He is currently pursuing the B.S. degree with Qingdao University of Technology. His research interest includes deep learning-based defect detection on steel surface.

JINGQI YANG was born in Zaozhuang, Shandong, China, in 2001. He is currently pursuing the bachelor's degree with Qingdao University of Technology. His research interest includes deep learning-based steel surface defect detection.

JISHUAI ZHOU was born in Dezhou, Shandong, China, in 2002. He is currently pursuing the bachelor's degree with Qingdao University of Technology. His research interest includes deep learning-based steel surface defect detection.