WEP Warwick Evans Publishing

International Journal of Computer Science and Information Technology

ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 7, Number 1, Year 2025
DOI: https://doi.org/10.62051/ijcsit.v7n1.05
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Surface Defect Detection of Steel Based on Improved YOLOv5s

Zongtai Sha, Lu Liu, Tianyang Wang, Hao Jiang

College of Mechanical Engineering, Tianjin University of Technology and Education, Tianjin 300222, China

ABSTRACT

To address the issues of low accuracy and frequent occurrences of missed and false detections in steel surface defect detection, this paper proposes an algorithm named GSM-YOLO, which is based on YOLOv5s. Firstly, the Ghost Bottleneck module is introduced to improve the C3 module of the backbone network. The Ghost module is used to generate diverse feature maps to obtain more detailed information and increase the detection accuracy of the model. Secondly, a Spatial and Channel Reconstruction Convolution (SCConv) is incorporated to refine the feature information generated by the C3 module, making the output feature information more precise and consequently improving the detection accuracy of the algorithm. Finally, a Multi-dimensional Collaborative Attention (MCA) mechanism is added at the bottom of the SPPF module. Spatial and channel feature information from the three-branch structure of MCA is fused to enhance the model's attention for multi-scale target features, and then missed and false detections are reduced. The experimental results show that the proposed algorithm achieves a recall rate (R), precision (P), and mean average precision (mAP) which are 2.8%, 5.3% and 5.2% higher than those of the original network, respectively, on the NEU-DET dataset. This effectively enhances the ability to detect surface defects in steel materials.

KEYWORDS

Steel Surface; YOLOv5; C3Ghost; Attention Mechanism

1. INTRODUCTION

The surface quality of steel has a critical impact on its performance and service life. Due to various factors such as equipment wear, improper operation, and environmental conditions, various defects may occur on the surface of steel, including rust, scratches, pitting, cracks, and others. These defects not only affect the appearance quality of the steel but may also severely compromise its mechanical properties and corrosion resistance, thereby threatening the overall quality and safety of the product in use. Therefore, advancing the technology for surface defect detection is of significant importance for improving the performance, quality, and longevity of industrial products [1].

Traditional methods for steel surface inspection primarily rely on visual observation by human inspectors. This approach not only consumes substantial human resources but also suffers from inefficiencies and accuracy issues that fail to meet the demands of modern automated production environments. With the application and development of industrial machine vision, computers have provided robust support for visual inspection in industrial fields. Traditional target detection methods mainly depend on manually designed feature extraction rules for identifying objects, followed by feeding these features into classification models to recognize and locate defects. Techniques such as LBP, GLCM, SVM, Adaboost, and others have been employed for steel surface defect detection [2-5]. However, these traditional target detection algorithms require extensive prior knowledge and involve cumbersome steps to execute the detection process, which is not only complex but also time-

consuming. Due to their heavy reliance on feature extraction, the robustness and generalization capabilities of these traditional target detection algorithms cannot be guaranteed.

In recent years, machine vision target detection algorithms based on deep learning have been widely applied in various industrial production fields. Among them, two-stage algorithms represented by Faster RCNN [6] and single-stage algorithms represented by YOLO (You Only Look Once) are the most prominent. Two-stage algorithms detect objects in two phases: first, they generate candidate regions, and then classify and recognize these regions. One-stage algorithms, on the other hand, perform object detection through a single forward propagation without the need to generate candidate boxes. Due to the differences in their detection processes, the detection accuracy of these two types of algorithms also varies accordingly.

Xia et al. [7] proposed an improved steel surface defect detection algorithm based on the YOLOv5 algorithm. By utilizing the RepLK module to enhance the C3 and SPP modules, and adding an additional target detection layer, they applied the improved RepLK-SPP module after each scale feature map. This method effectively enhanced the algorithm's ability to extract features of different scales. Although the improved model outperformed the original algorithm in terms of performance, it still had insufficient detection accuracy for multi-scale targets in steel defects.

Wang et al. [8] proposed a real-time steel surface defect detection technique based on the YOLOv5 detection network. They improved the algorithm by introducing multi-scale exploration blocks and spatial attention mechanisms. Although the improved algorithm shows certain enhancements compared to the original one, it still suffers from issues such as low average detection accuracy.

Zhou et al. [9] improved the YOLOv5 algorithm by integrating the Swin Transformer module with the original C3 module, replacing the traditional upsampling with the CARAFE upsampling operator, and adding a 3-D weight attention mechanism SimAM. Although the algorithm's accuracy was effectively improved, the enhanced model still had the issue of insufficient improvement in detection accuracy for multi-scale targets.

Although the aforementioned research methods have effectively improved the detection accuracy of algorithms, the field of steel surface defect detection still faces challenges such as inaccurate recognition of multi-scale target information, low detection accuracy, and issues of missed and false detections. To address these problems, this paper proposes a steel surface defect detection algorithm based on the YOLOv5s algorithm, with the following improvements:

(1) The Ghost Bottleneck [10] is introduced to replace the CSP Bottleneck in C3, and the generated C3Ghost module is used to replace some of the C3 modules in the backbone network. By performing linear operations on the original feature maps, the Ghost module generates feature maps with diverse information such as different gradient directions and resolutions. This enables the model to learn more details of the features and improve its accuracy.

(2) An efficient convolution SCConv [11] that reduces redundant information between spatial and channel features is introduced in the Backbone part of the network. By utilizing the spatial reconstruction and channel reconstruction modules of SCConv, the redundant information in the feature information generated by the C3 module is removed. This allows the output feature information of the C3 module to retain more effective information, enhancing the model's ability to learn features and thereby improving the detection accuracy of the algorithm.

(3) A multi-dimensional collaborative attention MCAAttention [12] is introduced after the SPPF module to further process the feature information output by the SPPF module. By utilizing the three-branch structure of MCA attention to fuse spatial and channel feature information, the model can better focus on both channel and spatial feature information during the subsequent detection process. This helps to reduce missed detections and false positives in the algorithm's detection results.

Finally, the effectiveness of the improved model is verified through the public NEU-DET dataset and its generalization ability is validated in the GC10-DET dataset.

2. RELATED WORK

With the continuous improvement of computing power, steel surface defect detection technology based on deep learning has been widely applied. The YOLO series of algorithms have continuously evolved, achieving an optimized balance between detection accuracy and speed. This has gradually attracted the attention of researchers which has led to extensive research and application in the field of defect detection.

Xin et al. [13] proposed the YOLOV5-ACCOF model for steel surface defect detection. They improved the C3 module in the backbone and neck network using CBAM and ODConv, replaced the SPPF module with ASPP, substituted the upsampling module with CARAFE, and replaced the CIOU loss function with Focaler-Iou. These improvements effectively enhanced the detection performance of the detector.

Fan et al. [14] addressed the issue of low detection accuracy by proposing an improved YOLOv5 method called ACD-YOLO. They optimized the anchor boundaries using an improved genetic algorithm, added context enhancement modules to both the head and backbone of the network, and employed efficient convolution operators. This approach not only solved the problem of increased computational complexity caused by the addition of context enhancement modules but also effectively improved the detection accuracy of the algorithm.

Zhang et al. [15] proposed an improved steel surface defect detection algorithm based on YOLOv5s. By introducing dilated spatial convolutional pooling pyramid, SK attention mechanism, and SIOU loss function, they effectively enhanced the detection accuracy of the algorithm.

Lv et al. [16] proposed an improved steel surface defect detection algorithm based on YOLOv5s. They introduced the Swin Transformer structure, added a coordinate attention mechanism, added a detection layer, and used the SIOU loss function, which effectively improved the accuracy of steel surface defect detection.

Regarding YOLOv7, Gao et al. [17] enhanced the network's feature fusion capability by incorporating the lightweight CARAFE upsampling operator. They redesigned the YOLOv7 detection head by integrating a cascaded attention mechanism and a decoupled head, which not only effectively improved the algorithm's detection accuracy but also significantly reduced the rate of missed defects. Liang et al. [18] adopted a progressive and cross-layer approach to fully fuse target semantic information. They embedded a dual-layer routing attention module and introduced the SIoU loss function, achieving an effective balance between detection accuracy and speed.

For YOLOv8, Song et al. [19] introduced deformable convolution DCNv2 and Biformer into the backbone network, replaced the neck network structure with BiFPN, and finally substituted the WIoUv3 loss function. These modifications effectively enhanced the algorithm's detection accuracy. Zhang et al. [20] introduced a novel EP module to optimize efficiency, a large separable convolutional attention module, and a spatial pyramid pooling module SPPF-LSKA to improve detection accuracy. They also replaced the loss function with Inner-CIOU to enhance both detection accuracy and speed, enabling the algorithm to perform detection tasks efficiently and accurately.

In this paper, we select YOLOv5s as the baseline model for three main reasons:

(1) Improvement and Optimization Space: The network structure of YOLOv5 is flexible for further improvement and optimization, enabling it to adapt to specific application requirements. For example, detection performance can be enhanced by introducing new attention mechanisms and improved network structures.

(2) Adaptive Multi-scale Detection Capability: YOLOv5s introduces adaptive anchor box computation and multi-scale prediction mechanisms, which enable it to dynamically adjust the size and aspect ratio of detection boxes. This better accommodates the diversity and complexity of steel surface defects. As a result, the model demonstrates higher detection accuracy when dealing with

defects of varying sizes and shapes, especially in the detection of small targets and complex backgrounds.

(3) Rich data augmentation methods:Built in rich data augmentation methods such as random scaling, cropping, flipping, color transformation, etc., can effectively increase the diversity of training data, improve the generalization ability of the model, and enable it to better cope with various complex scene object detection tasks.

3. INTRODUCTION TO THE YOLOV5 ALGORITHM

The YOLOv5 algorithm is a one-stage method in object detection algorithms. Its main idea is to divide the entire image into several grids, and each grid predicts the category and position information of the objects within the grid. Then, the target boxes are filtered based on the IoU value between the predicted boxes and the ground-truth boxes, and finally, the category and position information of the predicted boxes are output.

The YOLOv5 network structure mainly consists of three parts: Backbone, Neck, and Head.

The Backbone's primary function is to transform the original input image into multi-layer feature maps for subsequent object detection tasks. It is mainly composed of Conv modules, C3 modules, and SPPF modules.

Neck network: The object detection network often inserts some layers between the Backbone and the final Head output layer. Both YOLOv4 [21] and YOLOv5 adopt the FPN+PAN structure, but in YOLOv5, C3 based on CSPNet design is used to enhance the network's feature extraction ability.

The Head part, for an input image of size 640×640, performs downsampling by factors of 8, 16, and 32 to obtain three different sizes of feature maps, namely 80×80, 40×40, and 20×20, which are used to predict small, medium, and large objects, respectively.

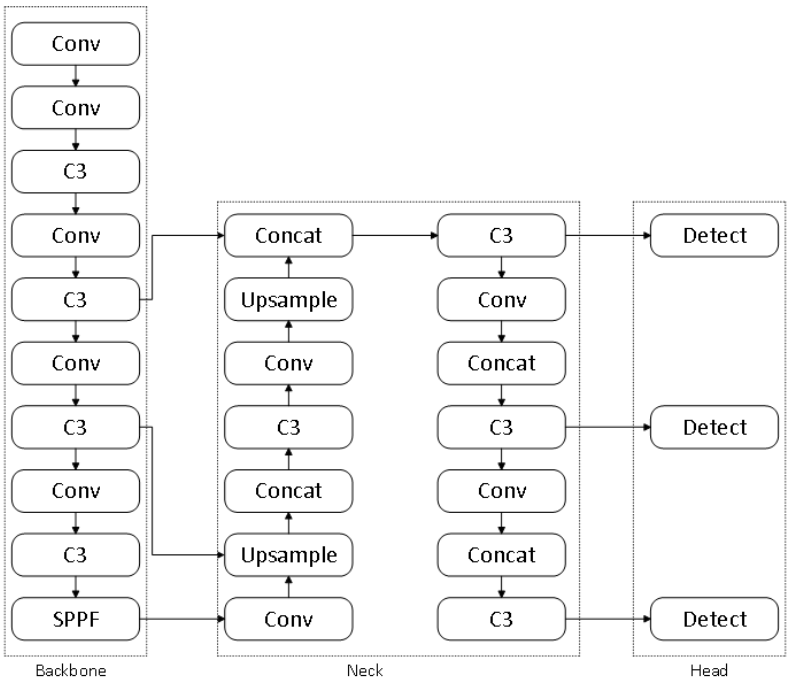

Figure 1. YOLOV5s model structure diagram

4. IMPROVEMENTS OF THE YOLOV5 ALGORITHM

To address the issues of insufficient detection accuracy and frequent occurrences of missed and false detections in steel surface defect detection, this study optimized the YOLOv5s architecture. By incorporating the GhostBottleneck concept to improve the local C3 module, embedding the Multi-Dimensional Collaborative Attention (MCA) mechanism, and replacing the deep convolution layers in the backbone network with SCConv convolutions, the network structure was redesigned and optimized. The improved algorithm structure is shown in Figure 2.

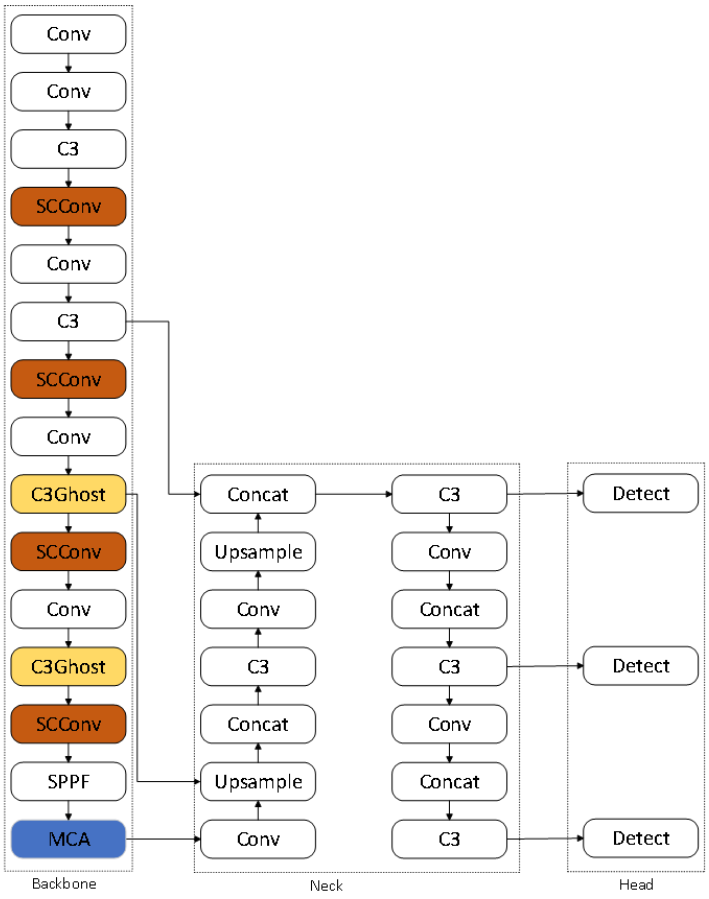

Figure 2. Structural diagram of the improved YOLOv5s network

4.1. Ghost Bottleneck Module

Traditional convolution operations process feature information using a one-to-one mapping approach, which may result in incomplete feature information. Additionally, some of these features contain a significant amount of redundant information, thereby limiting the model's ability to learn diverse features and ultimately affecting the detection accuracy of the algorithm. To address this issue, the Ghost bottleneck module is introduced. It first generates several intrinsic feature maps from the input information and then uses linear operations to produce feature maps with diverse informationfrom different gradient directions and resolutions feature maps.

The Ghost Bottleneck employs two different fusion operations, primarily determined by the value of the stride parameter. When stride=1, the first Ghost module acts as an expansion layer to increase the number of channels, while the second Ghost module reduces the number of channels to match the shortcut path. When the stride is 2, the difference compared to when the stride is 1 is that a depthwise convolution (DWConv) with stride 2 is inserted between the two Ghost modules.

As shown in Figure 3, the Ghost module generates diverse feature maps, and during the final output stage, these diverse feature maps are fused with the original feature maps. This ensures that the final

output features include not only the original features but also the diverse feature maps. This innovative design effectively addresses the model's limitations in learning diverse features. Therefore, this paper introduces the Ghost Bottleneck structure, constructed by stacking Ghost modules, to generate diverse feature maps that contain more target information and details. This enables the model to learn more intricate features, thereby resolving the issue of insufficient feature extraction capability that impacts the detection accuracy of the algorithm.

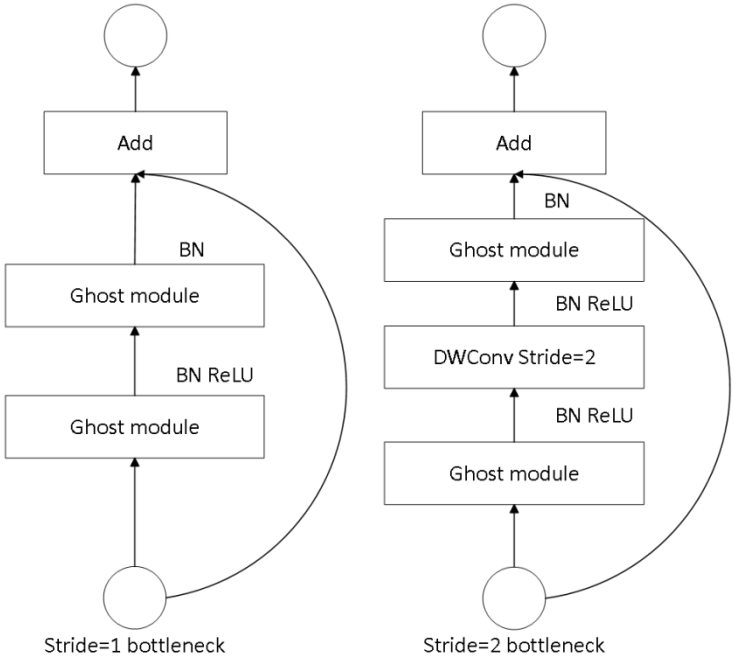

Figure 3. Ghost Bottleneck Structure

4.2. Patial and Channel Reconstruction Convolution

In YOLOv5, the Backbone plays a central role, extracting detailed feature information from low-level to high-level as it processes the image layer by layer. Specifically, the C3 module, as the core feature extraction module of the Backbone, relies heavily on convolutional operations to capture these features. However, this process often generates excessive redundant information, which not only increases computational burden but may also interfere with the precise learning of features by subsequent modules, ultimately leading to reduced detection accuracy.

The design of SCConv ingeniously addresses this issue. It consists of two core components: the Spatial Reconstruction Unit (SRU) and the Channel Reconstruction Unit (CRU). First, the input information undergoes feature separation through the SRU, distinguishing feature maps rich in spatial information from those with sparser content. Subsequently, through a refined reconstruction process, the feature maps with optimized weights are fused with the sparser feature maps, effectively reducing spatial redundancy in the feature information. Following this, the CRU further processes the output from the SRU through steps such as splitting, transforming, and merging, thereby reducing channel redundancy in the feature information. As a result, SCConv significantly reduces feature redundancy in both spatial and channel dimensions.

In light of this, we introduce SCConv to optimize the feature information extracted by the C3 module. By leveraging SCConv to reduce redundancy in both spatial and channel dimensions of the feature information, the output features from the C3 module become more precise. This enhancement improves the model's ability to learn features, ultimately boosting the detection accuracy of the algorithm.

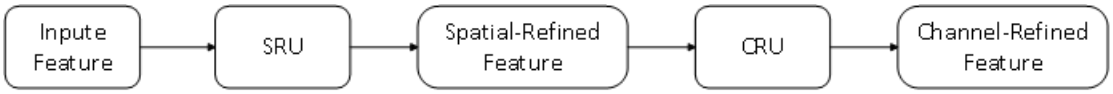

Figure 4. SCConv convolution structure

4.3. Multi-Dimensional Collaborative Attention

Existing attention mechanisms predominantly focus on either the spatial or the channel dimension, neglecting the feature information in the other dimension. This one-sided approach, constrained by the limited feature information it captures, often leads to inaccurate feature extraction by the model. To address this issue, this paper introduces the Multi-Dimensional Collaborative Attention (MCA) mechanism to improve YOLOv5. By doing so, the model can reduce its focus on irrelevant features, thereby mitigating the problems of missed and false detections during the detection process.

The Multi-Dimensional Collaborative Attention (MCA) integrates both spatial and channel dimensions, employing a three-branch structure to process feature information. The top two branches are responsible for capturing the interdependencies of features in the width and height dimensions of the spatial domain, respectively. The last branch primarily captures the interactions among channels. During the integration phase, the outputs from all three branches are simply averaged after being recalibrated by the attention weights generated from different dimensions, resulting in the final refined feature map.

Therefore, by adding MCA attention at the bottom of SPPF, the attention weights generated from different dimensions are used to recalibrate the feature information produced by SPPF. This enables the model to better focus on both channel and spatial feature information in the subsequent detection process, thereby reducing issues such as missed detections and false detections in the algorithm.

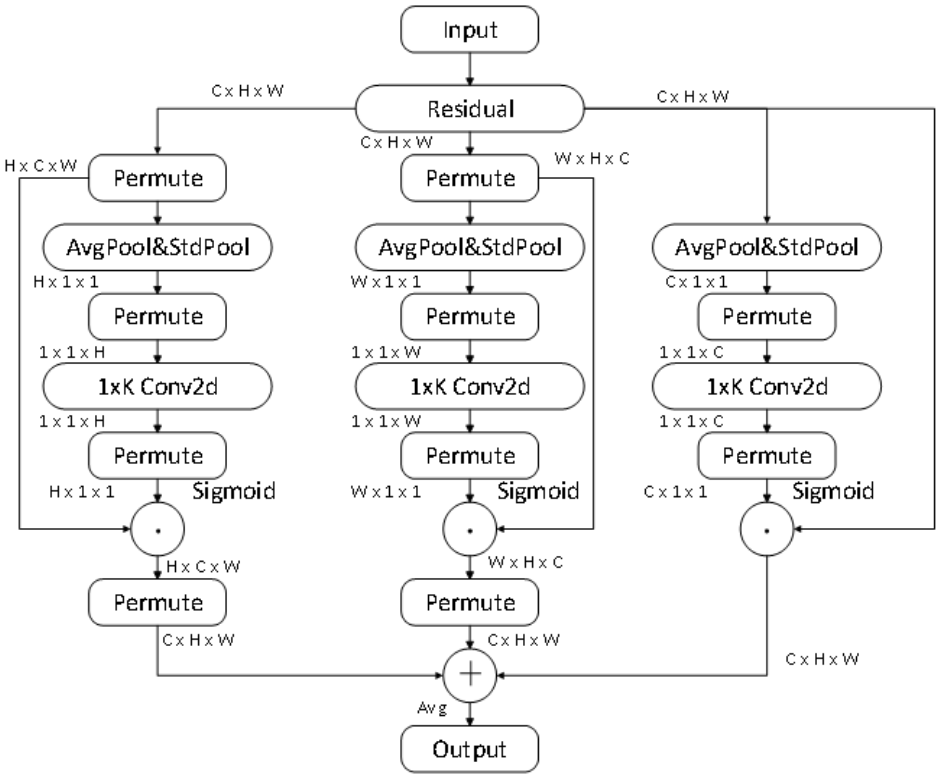

Figure 5. MCA attentional structure

5. EXPERIMENTS AND DISCUSSION

5.1. Datasets

To validate the improved algorithm proposed in this study, the NEU-DET dataset from Northeastern University was employed. This dataset contains 1,800 images covering six types of defects: crazing (Cr), inclusion (In), patches (Pa), pitted surface (PS), rolled-in scale (RS), and scratches (Sc). Each defect category contains 300 images, with all images having a resolution of 200×200 pixels. The dataset was divided into training and testing sets at a ratio of 8:2. The images of each defect type are shown in the figure below.

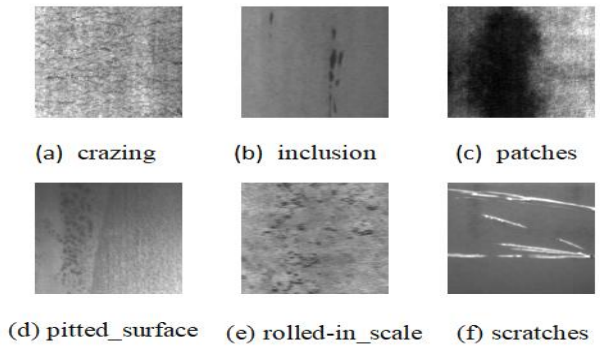

Figure 6. Some pictures of the data set

5.2. Experimental Environment And Parameter Settings

The experiments were made on a cloud server with the specifications detailed in Table 1. During the model training process, the learning rate was set to 0.01, the momentum to 0.0937, and the weight decay coefficient to 0.0005. The training was performed for 300 epochs with a batch size of 16. The input image size was set to 224×224 pixels, and the Stochastic Gradient Descent (SGD) optimizer was employed to update the model parameters.

Table 1. Experimental environment and configuration table

Name	Environmental configuration
GPU	NVIDIA GeForce RTX 2080Ti
CPU	Xeon E5-2678 v3
internal storage	64G
Python	Python3.8
Pytorch	Pytorch1.10

5.3. Evaluation Metrics

To verify the effectiveness of the improved algorithm presented in this paper, the Average detection accuracy (mAP) metric is primarily used to assess the detection accuracy across multiple categories. Additionally, the precision (P) and recall (R) are compared to ensure the reliability of the experimental data. The mAP is the arithmetic mean of the average precision (AP) values for all categories, and it is calculated using the formulas (1-4) provided below:

$$AP = \int_0^1 P(R)dR \tag{1}$$

$$\mathrm{mAP} = \frac{\sum_{1}^{n} AP_i}{n} \tag{2}$$

$$P = \frac{TP}{TP + FP} \tag{3}$$

$$R = \frac{TP}{TP + FN} \tag{4}$$

In the context of this study, the terms TP, FP, and FN stand for True Positives, False Positives, and False Negatives, respectively. The number of categories, denoted by n, is 6 in this paper.

5.4. Comparative Experiment

To validate the effectiveness of the modules proposed in this paper, a series of comparative experiments were conducted to verify their superiority over similar existing modules. All experiments were based on the YOLOv5 algorithm, with consistent experimental conditions and environments maintained throughout to ensure the fairness of the comparisons.

5.4.1. Comparative analysis of C3Ghost modules

To verify the effectiveness of the C3Ghost model in YOLOv5s, C3_SimAM [22], C3-ParNet [23] and C3-EffSE [24] were added to the same position in the YOLOv5s model and compared with C3Ghost in experiments (Table2). The experimental results showed that the C3Ghost performs best and the P and mAP are improved by 2.1% and 2.5% respectively compared to the original model.

Table 2. Comparison of C3 results for different modules

Module	R (%)	P (%)	mAP (%)
C3	71.6	72.8	75
C3_SimAM	72.7	72.3	76.3
C3_ParNet	73.4	73.1	76.3
C3_EffSE	72.7	72.3	75.9
C3Ghost	72.4	74.9	77.5

5.4.2. Comparative analysis of MCA Attention

To verify the effectiveness of introducing MCA attention in detecting surface defects in steel, we selected four different types of attention, CA [25], CBAM [26], ECA [27] and GAM [28] for comparative experiments (Table3). The experimental results showed that after introducing MCA attention, the recall rate was increased by 2.1% and mAP was increased by 1.8% compared to the original model, and the overall effect was also better than the four types of attention.

Table 3. Table of attention comparison experiments

Attention	R (%)	P (%)	mAP (%)
CA	73.4	71.8	74.6
CBAM	74.4	70.9	76.7
ECA	71.4	73.3	75.4
GAM	72.8	72.1	76
MCA	73.7	73.2	76.8

5.4.3. Comparative analysis of SCConv

To verify the effectiveness of adding SCConv to the model, we conducted comparative experiments by incorporating four common convolutional modules at the same position: ODConv [29], RFCBAMConv [30], RFCAConv [30] and RFAConv [30] (Table 4). The experimental results show that after introducing SCConv, the P and mAP increased by 3.3% and 2.3%, respectively, compared to the original model. Additionally, the mAP achieved with SCConv was consistently higher than that of the other types of convolutions.

Table 4. Convolution and comparison test table

Conv	R (%)	P (%)	mAP (%)
ODConv	73.2	71.5	75
RFCBAMConv	73.1	65.6	72.8
RFCAConv	73.7	71.9	76.4
RFAConv	71.4	71.8	75.5
SCConv	71.2	76.1	77.3

To more intuitively analyze the effect of adding SCConv to the model, visualization feature maps as shown in Fig7 were generated. It can be seen from the figure that after adding SCConv, the redundant feature information in the output of the C3 module is effectively removed, making the output feature information more accurate and clearer.

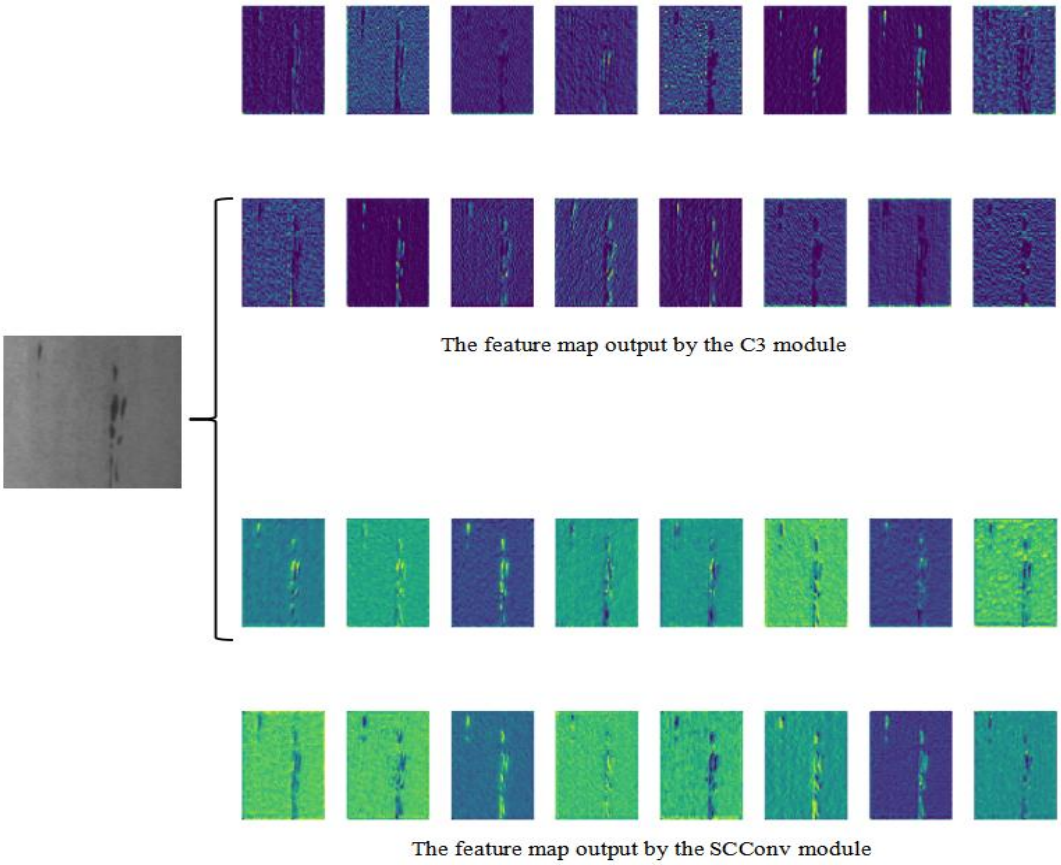

Figure 7. Comparison of SCConv and C3 feature map visualization

5.5. Comparison Experiment Between Ablation And Algorithm

5.5.1. Ablation experiment

To verify the feasibility of the proposed modules seperately, ablation experiments are conducted. For the benchmark model YOLOV5s, the added C3Ghost, C3Ghost+MCA, C3Ghost+SCConv, and C3Ghost+SCConv+MCA modules were trained and tested separately. The results are shown in Table5.

Table 5. Comparison table of ablation experiments

Module	R(%)	P(%)	mAP(%)
YOLOV5s	71.6	72.8	75
YOLOV5S+C3Ghost	72.4	74.9	77.5
YOLOV5S+C3Ghost+MCA	74.2	74.4	78.3
YOLOV5S+C3Ghost+SCConv	74.7	72.1	78.6
YOLOV5S+C3Ghost+SCConv+MCA	74.4	78.1	80.2

It is shown that replacing C3 with C3Ghost, mAP was increased by 2.5%; Adding C3Ghost+MCA increases mAP by 2.8%; Adding C3Ghost+SCConv increases mAP by 3.6%; When all modules are combined together, mAP was increased by 5.2%.

5.5.2. Comparative experiments of different algorithms

To further verify the effectiveness of the algorithm proposed in this paper, comparative experiments were conducted using the same dataset as in this paper. The following 10 mainstream algorithms were selected for comparison in sequence: SSD [31], RE-DETR [32], Faster-RCNN, YOLOV5, YOLOV7 [33], YOLOV8 [34], YOLOV9 [35], YOLOV10 [36], YOLOV11 [37] and YOLOV12 [38] (Table6).

Table 6. Table of the performance comparison of the different algorithms on the dataset

algorithm	Cr (%)	In (%)	Pa (%)	Ps (%)	Rs (%)	Sc (%)	mAP (%)
SSD	36.8	79.2	85.4	69.5	61.9	82.2	69.2
RE-DETR	27.3	76.3	91	69	57.8	89.5	68.5
Faster_RCNN	50.4	75.1	87.2	73.4	68	92	74.4
YOLOV5	47.1	82	94.4	74.2	61.5	90.6	75
YOLOV7	51.3	84.8	93.6	72.6	68.4	90.4	76.9
YOLOV8	41.9	86.2	93.5	71.7	64.2	94.4	75.3
YOLOV9	36.8	83.8	94.5	76.7	56.8	95.1	73.9
YOLOV10	38.9	79.9	91.1	67.8	56.7	85.7	70
YOLOV11	43.3	77.5	93.4	75.4	66.6	90.8	74.5
YOLOV12	45.4	85.1	93.1	73	69.4	94.2	76.7
Ours	57.2	84.6	94.5	77.3	74.7	92.8	80.2

The experimental results show that compared with SSD and Faster RCNN algorithms, the mAP of the proposed algorithm was improved by 11% and 5.8% ,respectively. For the two difficult-to-recognize defects, Cr (crazing) and Rs (rolled-in scale), the average accuracy is improved by 10.1% and 14.9% compared with the original algorithms, respectively, and the mAP is increased by 5.2%. Therefore, the proposed algorithm in this paper performs excellently in defect detection and effectively improves the accuracy of defect detection.

In order to verify the advantages of the proposed algorithm, we selected four improved models from recent literatures to compare (Table7). The experimental results showed that the algorithm performed well in two types of defect detection tasks, Cr (Crazing) and Rs (rolled-in scale). Although there was a certain gap in the performance improvement of Ps (pitting_surface), the improvement effect of the

algorithm in the other three types of defects is relatively small compared to the other four models. Overall, the improvement method proposed in this article has more advantages.

Table 7. Comparison of Results from Different Papers

algorithm	Cr (%)	In (%)	Pa (%)	Ps (%)	Rs (%)	Sc (%)	mAP (%)
Paper [13]	52.9	87.5	92.4	82	68.2	77.5	76.8
Paper [14]	43.9	85.4	93.7	89.7	66.6	93.8	78.9
Paper [15]	55.7	83.2	95	88.6	61.9	93.6	79.7
Paper [16]	45	84.9	93.8	84.9	65.5	94.3	78.13
Ours	57.2	84.6	94.5	77.3	74.7	92.8	80.2

5.5.3. Comparison of generalization experiments

To comprehensively evaluate the generalization capability of the algorithm, this study selected the public dataset GC10-DET. As shown in Fig7, the defect categories from left to right are: crescent gap (Cg), silk spot (Ss), water spot (Ws), crease (Cr), weld line (Wl), oil stain (Os), inclusion (In), waist fold (Wf), rolling pit (Rp), and punched hole (Pu). The dataset was divided into training and validation sets in 8:2 ratio, with experimental settings consistent with those described in Section 4.2. The experimental results are presented in Table8.

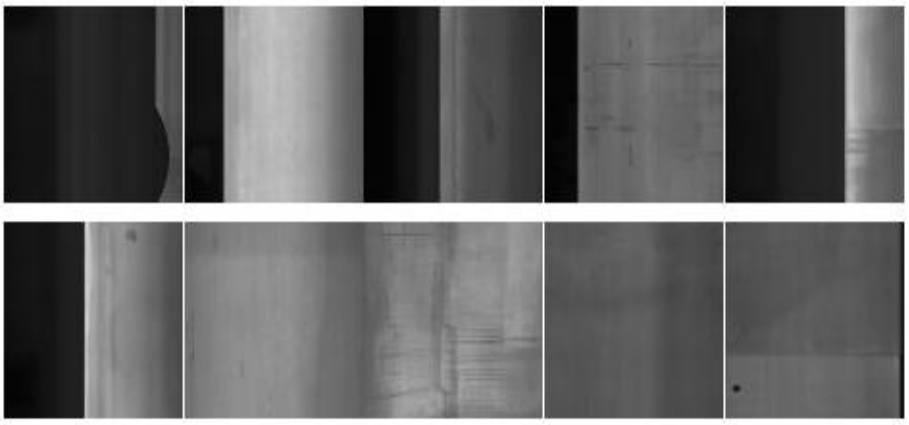

Figure 8. Partial images of GC10 dataset

According to the experimental results in Table 8, on the GC10 dataset, although the improved algorithm exhibits a 1% decrease in recall compared to the original YOLOv5s algorithm, its precision improves by 1.1%, while the mean average precision (mAP) increases by 2.2%. These findings demonstrate that the improved algorithm still performs effectively on the GC10-DET dataset, particularly in multi-scale object detection, maintaining strong robustness and generalization capability.

Table 8. Performance Comparison of GC10 Dataset

Model	R	P	MAP
YOLOV5s	58.8	60	56.9
Ours	57.8	61.1	59.1

5.5.4. Detection result

To evaluate the effectiveness of the improved algorithm, six different defect images were randomly selected for detection verification. The comparison of detection results before and after the improvement is shown in Fig9. It can be seen from the results that the improved algorithm in this paper has addressed the issue of missed detections and successfully identified defects that the original algorithm failed to detect.

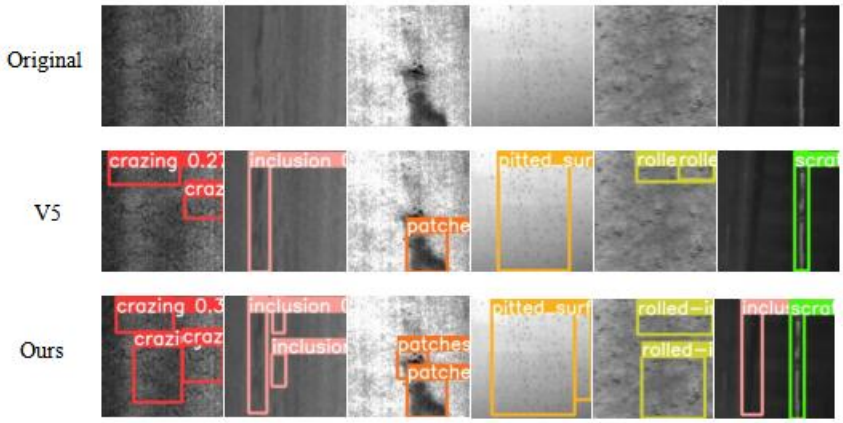

Figure 9. Comparison diagram of the test results

Fig10 shows a comparison of confusion matrices, with the original YOLOV5 algorithm detection results on the left and the improved algorithm detection results on the right. Through the detection results, it can be seen that the proposed algorithm has improved in the detection of three types of defects: cracking, rolled-in-scale, and scratch. At the same time, the false detection rate of 0.01 for identifying patches as inclusions has been eliminated.

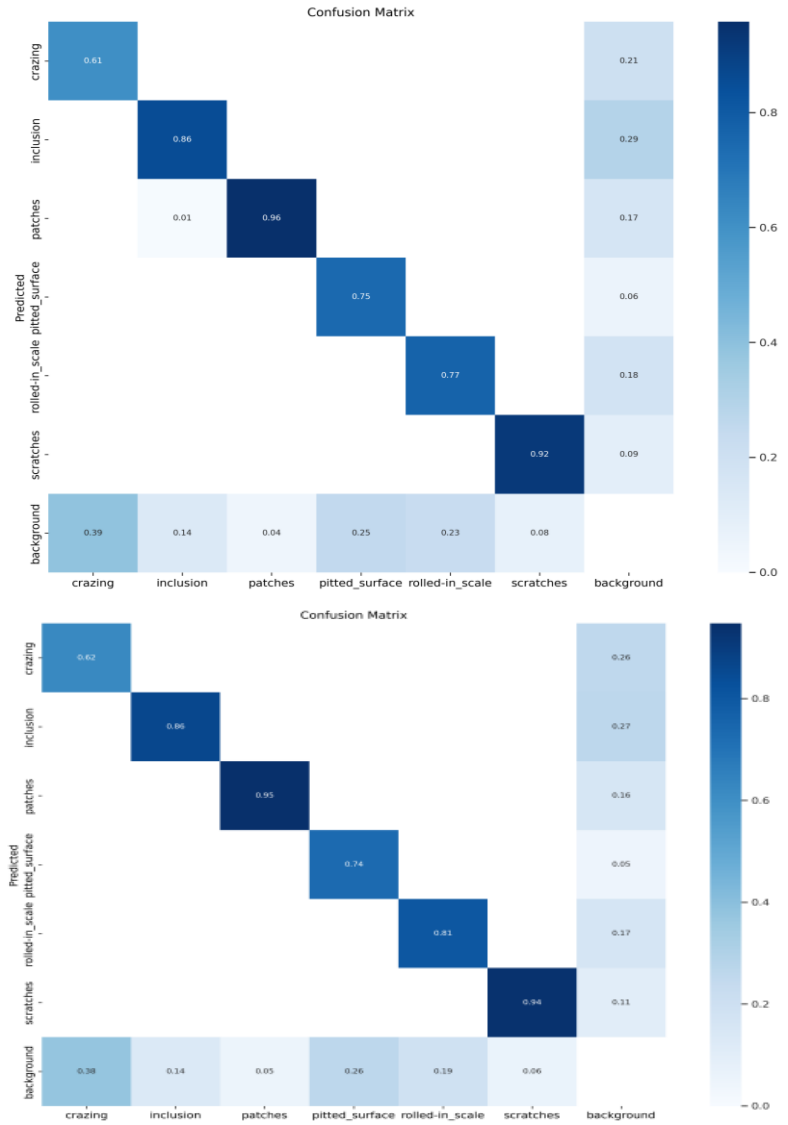

Figure 10. Shows a comparison of confusion matrices

6. CONCLUSION

This paper proposes a steel surface defect detection algorithm based on YOLOv5s to address the low detection accuracy and issues of missed and false detections in steel surface defect detection. Firstly, the Ghost Bottleneck module is introduced to improve the C3 module in the backbone network. The Ghost module generates diverse feature maps to obtain more detailed information, thereby enhancing the detection accuracy of the model. Secondly, a spatial and channel reconstruction convolution SCConv is added to simplify the feature information generated by the C3 module, making the output feature information of the C3 module more accurate and further improving the detection accuracy of the algorithm. Lastly, a multi-dimensional collaborative attention MCA is added at the bottom of the backbone network. Its three-branch structure fuses spatial and channel feature information, enhancing the model's attention to multi-scale target feature information and thus reducing missed and false detections. The experimental results show that the proposed algorithm achieves a recall rate (R) of 74.4, precision (P) of 78.1, and mean average precision (mAP) of 80.2 in the NEU-DET dataset, which are 2.8%, 5.3%, and 5.2% higher than those of the original network, respectively. This effectively enhances the ability to detect steel surface defects.

REFERENCES

[1] Li, J., Li, H., Hu, X. K., et al. Research Progress on Deep Learning-Based Surface Defect Detection Technology. Computer Integrated Manufacturing Systems, 2024, 30(3): 774-790. DOI:10.13196/j.cims.2023.IM28.

[2] Luo Q, Sun Y, Li P, et al. Generalized completed local binary patterns for time-efficient steel surface defect classification [J]. IEEE Transactions on Instrumentation and Measurement, 201 8, 68(3): 667-679.

[3] Kumar J, Srivastava S, Anand R S, et al. GLCM and ANN based approach for classification of radiographics weld images [C]. International Conference on Industrial and Information Systems, 201 8: 168-172.

[4] Cortes C. Support-Vector Networks [J]. Machine Learning, 1995.

[5] Rätsch G, Onoda T, Müller K R. Soft margins for AdaBoost [J]. Machine learning, 2001, 42: 287-320.

[6] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks [J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 39(6): 1137-1149.

[7] Xia K, Lv Z, Zhou C, et al. Mixed receptive fields augmented YOLO with multi-path spatial pyramid pooling for steel surface defect detection [J]. Sensors, 2023, 23(11): 5114.

[8] Wang L, Liu X, Ma J, et al. Real-time steel surface defect detection with improved multi-scale YOLO-v5 [J]. Processes, 2023, 11(5): 1357.

[9] Zhou, Y. L., Wu, X. C., Liu, W. G., et al. (2023). STCS-YOLO based algorithm for strip steel surface defect detection. China Metallurgy, 33(12), 128-138.

[10] Han K, Wang Y, Tian Q, et al. Ghostnet: More features from cheap operations [C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 1580-1589.

[11] Li J, Wen Y, He L. Scconv: Spatial and channel reconstruction convolution for feature redundancy [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 6153-6162.

[12] Yu Y, Zhang Y, Cheng Z, et al. MCA: Multidimensional collaborative attention in deep convolutional neural networks for image recognition [J]. Engineering Applications of Artificial Intelligence, 2023, 126: 107079.

[13] Xin H, Song J. YOLOv5-ACCOF Steel Surface Defect Detection Algorithm [J]. IEEE Access, 2024

[14] Fan J, Wang M, Li B, et al. ACD-YOLO: Improved YOLOv5-based method for steel surface defects detection [J]. IET Image Processing, 2024, 18(3): 761-771.

[15] Zhang, S. Q., Shi, W. Y., Zhang, S. W., et al. (2023). Steel surface defect detection based on improved YOLOv5 algorithm. Science Technology and Engineering, 23(35), 15148-15157.

[16] Lv, X. L., Lu, H. B., Hou, C. G., et al. (2024). Improved YOLOv5s-based algorithm for steel surface defect detection. Chemical Industry Automation and Instrumentation, 51(2), 301-309.

[17] Gao, C. Y., Qin, S., Li, M. H., et al. (2024). Research on steel surface defect detection using improved YOLOv7 algorithm. Computer Engineering and Applications, 60(7), 282-291.

[18] Liang, L. M., Long, P. W., Lu, B. H., et al. (2024). Improved GBS-YOLOv7t for steel surface defect detection. Opto-Electronic Engineering, 51(5), 61-73.

[19] Song X, Cao S, Zhang J, et al. Steel Surface Defect Detection Algorithm Based on YOLOv8 [J]. Electronics, 2024, 13(5): 988.

[20] Zhang X, Wang Y, Fang H. Steel Surface Defect Detection Algorithm Based on ESI-YOLOv8 [J]. 2024.

[21] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection [J]. arxiv preprint arxiv:2004.10934, 2020.

[22] Yang L, Zhang R Y, Li L, et al. Simam: A simple, parameter-free attention module for convolutional neural networks [C]//International conference on machine learning. PMLR, 2021: 11863-11874.

[23] Goyal A, Bochkovskiy A, Deng J, et al. Non-deep networks [J]. Advances in neural information processing systems, 2022, 35: 6789-6801.

[24] Lee Y, Park J. Centermask: Real-time anchor-free instance segmentation [C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 13906-13915.

[25] HOU Q B, ZHOU D Q, FENG J S. Coordinate attention for efficient mobile network design [C]. Washington: Proceedings of the 2021 IEEE Conference on Computer Vision and Pattern Recognition, 2021: 13713-13722. DOI: 10.1109/CVPR46437.2021.01350.

[26] WOO S, PARK J, LEE J Y, KWEON I S. Cbam: convolutional block attention module [C]. Munich: Proceedings of the European conference on computer vision, 2018:3-19.

[27] WANG Q L, WU B G, ZHU P F, LI P H, ZUO W M, HU Q H. ECA-Net: efficient channel attention for deep convolutional neural networks [C]. Nanjing: Proceedings of the lEEE/CVF conference on computer vision and pattern recognition, 2020:11534-11542. DOI:10.1109/CVPR42600.2020.01155.

[28] Liu Y, Shao Z, Hoffmann N. Global attention mechanism: Retain information to enhance channel-spatial interactions [J]. arxiv preprint arxiv: 2112.05561, 2021.

[29] Li C, Zhou A, Yao A. Omni-dimensional dynamic convolution [J]. arxiv preprint arxiv: 2209.07947, 2022.

[30] Zhang X, Liu C, Yang D, et al. Rfaconv: Innovating spatital attention and standard convolutional operation [J]. Arxiv Preprint Arxiv, 2023, 2304.03198:1-12.

[31] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector [C]//Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21-37.

[32] Zhao Y, Lv W, Xu S, et al. Detrs beat yolos on real-time object detection [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 16965-16974.

[33] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors [C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 7464-7475.

[34] Varghese R, Sambath M. YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness [C]//2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS). IEEE, 2024: 1-6.

[35] Wang C Y, Yeh I H, Liao H Y M. Yolov9: Learning what you want to learn using programmable gradient information [J]. arxiv preprint arxiv: 2402.13616, 2024.

[36] Wang A, Chen H, Liu L, et al. YOLOv10: Real-Time End-to-End Object Detection. arXiv 2024 [J]. arXiv preprint arXiv:2405.14458.

[37] Khanam R, Hussain M. Yolov11: An overview of the key architectural enhancements [J]. arXiv preprint arXiv: 2410.17725, 2024.

[38] Tian Y, Ye Q, Doermann D. Yolov12: Attention-centric real-time object detectors [J]. arXiv preprint arXiv: 2502.12524, 2025.