Mathematical Biosciences and Engineering

MBE, 21(1): 346–368.
DOI: 10.3934/mbe.2024016
Received: 09 October 2023
Revised: 27 November 2023
Accepted: 28 November 2023
Published: 13 December 2023

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

Research article

Improved YOLOv7-based steel surface defect detection algorithm

Yinghong Xie[1], Biao Yin[1,*], Xiaowei Han[2] and Yan Hao[1]

[1] School of Information Engineering, Shenyang University, Shenyang 110003, China
[2] Institute for Science, Technology and Innovation, Shenyang University, Shenyang 110003, China

* **Correspondence:** Email: yin990228@163.com.

Abstract: In response to the limited detection ability and low model generalization ability of the YOLOv7 algorithm for small targets, this paper proposes a detection algorithm based on the improved YOLOv7 algorithm for steel surface defect detection. First, the Transformer-InceptionDWConvolution (TI) module is designed, which combines the Transformer module and InceptionDWConvolution to increase the network's ability to detect small objects. Second, the spatial pyramid pooling fast cross-stage partial channel (SPPFCSPC) structure is introduced to enhance the network training performance. Third, a global attention mechanism (GAM) attention mechanism is designed to optimize the network structure, weaken the irrelevant information in the defect image, and increase the algorithm's ability to detect small defects. Meanwhile, the Mish function is used as the activation function of the feature extraction network to improve the model's generalization ability and feature extraction ability. Finally, a minimum partial distance intersection over union (MPDIoU) loss function is designed to locate the loss and solve the mismatch problem between the complete intersection over union (CIoU) prediction box and the real box directions. The experimental results show that on the Northeastern University Defect Detection (NEU-DET) dataset, the improved YOLOv7 network model improves the mean Average precision (mAP) performance by 6% when compared to the original algorithm, while on the VOC2012 dataset, the mAP performance improves by 2.6%. These results indicate that the proposed algorithm can effectively improve the small defect detection performance on steel surface defects.

Keywords: YOLOv7; transformer; attention mechanism; SPPFCSPC; defect detection

1. Introduction

Steel is the foundation and support of modern construction. The steel production industry is a crucial foundational sector of the national economy, serving as a vital support for the construction of a modern and powerful nation. Moreover, it is a key area for achieving green and low-carbon development. In the process of steel production, various factors can lead to surface defects in steel materials. Steel surface defect detection can effectively screen out unqualified steel and prevent it from entering the market. It can help enterprises identify and solve problems in the production process in a timely manner and improve production processes and equipment. Therefore, the detection of surface defects in steel is a critical technology to ensure the quality of steel products, enhance production efficiency, reduce costs, ensure safety, and maintain credibility. In the steel production process, the detection of defects in steel is an indispensable step.

The traditional manual visual inspection method has a low efficiency, poor stability, high labor intensity, and a high cost, making it difficult to meet the needs of the modern manufacturing industry. To address the limitations of manual inspection, some researchers have explored the use of image processing techniques for steel surface defect detection. In recent years, many scholars have integrated deep learning frameworks with defect detection methods, thus achieving promising results. Shuang et al. [1] leveraged the strengths of convolutional neural networks (CNNs) and autoencoders (AEs) to learn the normal patterns in images and use reconstruction error to detect defects. He et al. [2] introduced a defect detection framework using regressions and classifications. Additionally, they presented a high-performance deep network architecture and a label generation algorithm to capture defect severity information in data annotations.

In recent years, numerous researchers have integrated deep learning frameworks with defect detection methods to develop end-to-end network models for defect detection, thereby achieving remarkable results. Luo et al. [3] proposed a decoupled two-stage object detection framework based on CNNs to address the challenges in detecting surface defects on flexible printed circuit boards (FPCB). They introduced a multilevel hierarchical aggregation (MHA) module as a feature enhancement module for precise defect localization and a local non-local (LNL) module for enhancing spatial encoding features (SEF) in defect classification tasks, thereby effectively localizing defects. Shao et al. [4] introduced a method for the pixel-wise, semi-supervised detection of textile defects by integrating a multi-task mean teacher (MT) framework. They established a multi-task detection network (ST-CNN) for detecting defect contours, defect areas, and defect distance maps, thereby aiding in defect segmentation. This model served both as a student network and a teacher network, thereby enabling the effective detection of textile defects with limited annotated samples. Chen et al. [5] proposed a genetic algorithm-based Gabor faster region-based CNN (Faster GG R-CNN), which incorporates Gabor kernels into Faster R-CNN. They devised a two-stage training methodology that combined a genetic algorithm (GA) and a backpropagation to train the Faster GG R-CNN model, thereby enabling the effective detection of textile defects under varying backgrounds, positions, and sizes.

In recent years, several YOLO-based algorithms [6–9] have been introduced, thus making significant progress in the domain of defect detection. For instance, Qian et al. [10] modified the YOLOv3 algorithm by replacing the original DarkNet53 with ShuffleNetv2. Moreover, they proposed the lightweight feature pyramid network (LFPN) network to enhance feature fusion for an improved efficiency in handling multi-scale features. Yang [11] integrated an improved YOLOv5 network into the domain of steel surface defects. They added a convolutional block attention module to the YOLOv5

network, thereby improving the detection accuracy by emphasizing crucial information. Wang et al. [12] proposed a defect detection algorithm based on an enhanced YOLOv7 model. They utilized a weight-dismantled weighted bi-directional feature pyramid network (BiFPN) structure to maximize the feature information fusion and to reduce the feature loss during the convolution process. Wang [13] introduced an improvement to the model based on You Only Look Once X (YOLOX). They embedded coordinate attention blocks within the backbone to enhance the modeling capabilities, employed zooming loss to address the foreground-background class imbalance, and predicted intersection over union-aware (IoU-aware) classification scores as a detection ranking criteria. Furthermore, they applied complete intersection over union (CIoU) loss to the regression branch to enhance the defect localization performance. Li et al. [14] introduced the YOLOv6 series, which incorporated replicated visual geometry group (ReVGG) for structural reparameterization and adopted the scale-invariant intersection over union (SIoU) loss function for a superior detection performance. Wang et al. [15] developed the YOLOv7 series, which leveraged an efficient long-range aggregation network and a cascaded model scaling strategy to effectively enhance the algorithm's detection capabilities.

The detection of small target defects is a common challenge in defect detection. Fityanul Akhyar [16] and others introduced a novel approach using RetinaNet to streamline defect detection from a two-stage to a more efficient single-stage process. Vikanksh Nath [17] proposed a hybrid model, S2D2Net, for an efficient and robust surface defect detection in steel materials during the manufacturing process. S2D2Net employed pre-trained ImageNet models as feature extractors and learned capsule networks on the extracted features.

The existing mainstream YOLOv7 series algorithms have faced persistent technical challenges in the detection of small objects, making them less suitable for steel inspection. To improve the detection accuracy of small defects in steel, this paper proposed an improved YOLOv7 algorithm. The proposed algorithm aims to overcome the limitations of current technologies in detecting small objects within the context of steel defect inspection. This paper's contributions include the following:

1) We propose the transformer-inception (TI) module, which is a novel fusion of the TransformerBlock [18] and the InceptionDWConvolution [19]. We seamlessly integrate the TI module into the YOLOv7 architecture, thus significantly improving the network's accuracy in detecting small target defects.

2) To optimize the network structure, we introduce the global attention mechanism (GAM) [20] module. This module is seamlessly fused with the YOLOv7 backbone network. It effectively extracts features at different scales while suppressing irrelevant information. The inclusion of the GAM attention mechanism substantially boosts the algorithm's proficiency in small target defect detection.

3) In contrast to the traditional spatial pyramid pooling cross-stage partial channel (SPPCSPC) module, we introduce the spatial pyramid pooling with feature context spatial pyramid convolution (SPPFCSPC) module. This innovative module not only maintains the algorithm's speed enhancements, but also ensures that the receptive field remains at the desired level. This adaptation significantly improves the algorithm's capacity.

The content overview of the subsequent sections is as follows. In Section 2, a detailed introduction to the YOLOv7 network is provided. Section 3 introduces the enhanced YOLOv7 network, thereby outlining the functions and characteristics of each module within the network. Section 4 offers an overview of the specific experimental design, encompassing the dataset introduction, the experimental environment and parameter settings, the evaluation criteria, the ablation experiments, and the comparative trials. Section 5 comprehensively summarizes the paper by emphasizing the algorithm's

strengths and weaknesses, along with any potential future solutions and directions.

2. YOLOv7 network structure

In this paper, we adopted the YOLOv7 network framework as the fundamental architecture for steel surface defect detection. YOLOv7 represents an evolution of the YOLOv5 model by incorporating several improvements and innovations in object detection. It was introduced in July 2022, and has demonstrated a superior performance across a wide range of frame rates, from 5 frames per second (FPS) to 160 FPS, thereby surpassing all known object detectors. The network structure of YOLOv7 is depicted in Figure 1. YOLOv7 consists of three main components:

1) Backbone: It is responsible for the feature extraction. It incorporates CSPNet and focus modules, thus enhancing multi-scale features and facilitating information flow between channels. These improvements contribute to an improved feature extraction.

2) Feature pyramid network (FPN): It handles feature fusion and utilizes SPP and PANet modules. These modules enhance the semantic understanding of features and spatial resolution, thus further refining the feature representation.

3) Head: It is responsible for generating predictions. YOLOv7 introduces various techniques, such as adaptive anchors and label smoothing, to enhance prediction accuracy.

Compared to YOLOv5, YOLOv7 introduces the efficient layer aggregation network (ELAN) and multilayer perceptron (MP) modules by incorporating them into both the backbone and FPN sections to enhance the feature extraction capabilities.

Overall, the adoption of YOLOv7 for our steel surface defect detection approach leverages its advanced architecture and performance improvements to achieve a more accurate and efficient defect detection.

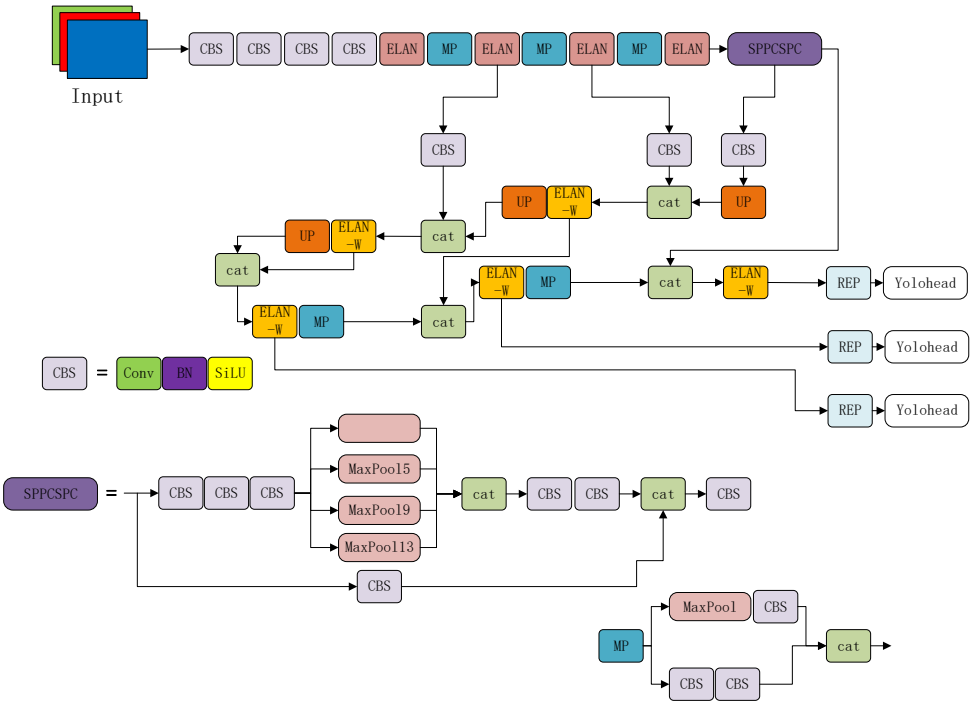


Figure 1. YOLOv7 network structure.

Upon establishing the aforementioned network architecture, YOLOv7 utilizes a binary cross-entropy loss function to compute the classification and confidence losses. The formulas are as follows:

$$\delta(x_n) = \frac{1}{1 + e^{x_n}}, \tag{1}$$

$$L_{conf} = L_{class} = -\frac{1}{n}\sum(y_n * \ln x_n + (1 - y_n) * \ln(1 - y_n)), \tag{2}$$

where y_n represents the probability of sample x_n being predicted as a positive instance, and n denotes the number of samples.

The localization loss is computed using the CIoU loss function, with the following formula:

$$CIoU = IoU - \left(\frac{\rho^2(b, b^{gt})}{c^2} + \alpha v\right). \tag{3}$$

In the provided context, α represents a weight function and v is used to measure the consistency of aspect ratios; these variable can be calculated as follows:

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{4}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w_{gt}}{h_{gt}} - \arctan\left(\frac{w}{h}\right)\right)^2 \tag{5}$$

The final definition of CIoU loss is as follows:

$$L_{loc} = L_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + av \tag{6}$$

Combined with the adaptive matching of positive samples to feature points using simplified optimal transport assignment (SimOTA), the final formula for the weighted sum of the three loss functions to compute the training loss is as follows:

$$L_{object} = L_{loc} + L_{conf} + L_{class} \tag{7}$$

3. The proposed YOLOv7 algorithm network structure

This paper presents improvements to the YOLOv7 network model, as illustrated in Figure 2. First, the custom-designed TI module is incorporated as a small target detection layer into the FPN layer, thereby improving the network's small defect detection capability. Additionally, the SPPFCSPC module replaces the original SPPCSPC module. This change maintains the speed improvement while keeping the receptive field at the same level, thus enhancing the algorithm's capability for steel surface defect detection. Furthermore, multiple GAM modules are introduced into the FPN layer to increase the network's focus on critical defect features. In this paper, two GAM modules and two TI modules are appended after the last two MP modules in the FPN layer. The GAM leverages the interaction between the channel and spatial dimensions to enhance the semantic and edge information in the feature map. The TI modules are added after the efficient layer aggregation network-wide (ELAN-W)

module to augment the YOLOv7 network with a small object detection layer without compromising the general image feature extraction capabilities of regular convolutions. This addition aims to enhance the target detection capabilities of the YOLOv7 network.

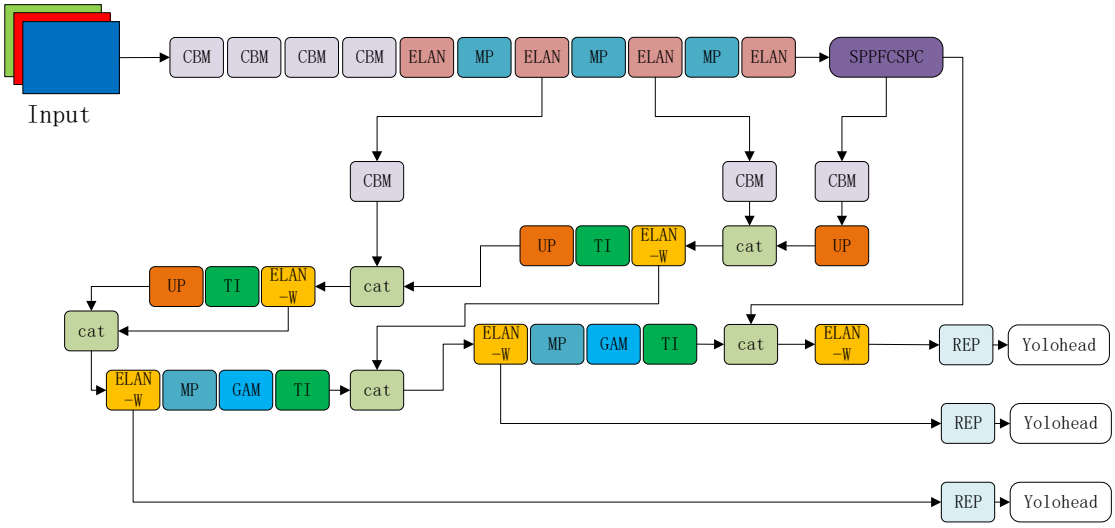


Figure 2. Improved YOLOv7 network structure diagram.

3.1. TI module

In the context of small target defect detection, we have introduced a transformer module into the FPN of YOLOv7. By utilizing the self-attention mechanism of the transformer module to extract the global dependencies and semantic relationships of feature maps, adjust the position weights of feature maps, and enhance the semantic expression ability of feature maps, the network can enhance its attention to small targets, thereby enhancing its ability to detect small targets.

While traditional transformer modules excel in handling sequential data, they may not effectively capture spatial information within image data, which exhibits inherent spatial structures. Given that YOLOv7 is an object detection model, it necessitates the spatial awareness and precise localization of objects within input images. Therefore, a lack of spatial information can adversely affect the model's accuracy.

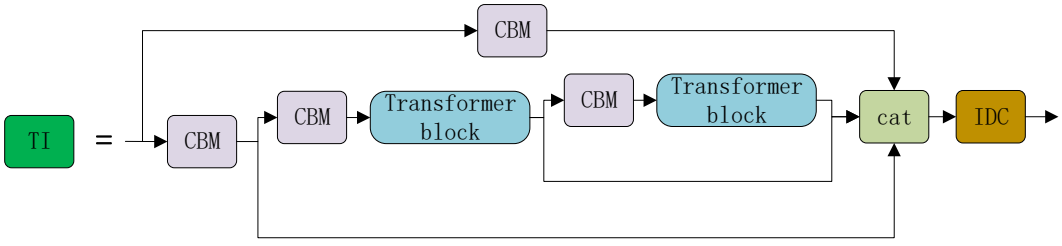


Figure 3. TI module structure diagram.

To address these limitations, this paper introduces the TI module as a dedicated layer for small

target detection within the network, as illustrated in Figure 3. In this design, the TransformerBlock is positioned after convolutional layers. By fusing features from the convolutional layers and the TransformerBlocks, we can use both the spatial awareness capabilities of convolutional layers and the modeling capabilities of TransformerBlocks. This enables a more effective capture of spatial information, thereby allowing the TransformerBlock to incorporate higher-level pixel information. Moreover, the combination of convolutional layers and TransformerBlock can improve the ability of TransformerBlock to extract information at different scales, thus enabling the model to learn richer feature representations and improve the detection accuracy. Additionally, to address potential shortcomings in spatial awareness and parameter efficiency when directly using TransformerBlocks within YOLOv7's FPN, we introduce an InceptionDWConvolution layer at the end of the module. This layer can receive more global features after passing through the TransformerBlock layers, thus resulting in a finer and more comprehensive feature map; this enhances the model's perceptual capabilities for objects of different scales and improves the generalization performance. By combining the module of TransformerBlock with the module of InceptionDWConvolution, the proposed algorithm effectively utilizes the global modeling from TransformerBlock and the spatial awareness of InceptionDWConvolution, which enables the model to have a more thorough understanding of images, adapt to objects of varying scales, and enhance the accuracy of small object detection.

In this paper, the TI module has been incorporated into the FPN network. The module is added after the ELAN-W module, with the ELAN-W module is retained to preserve the image feature extraction capabilities of regular convolution layers. This addition of the small target detection layer to the network has been performed without compromising the network's ability to extract features from images.

3.1.1. Transformer block

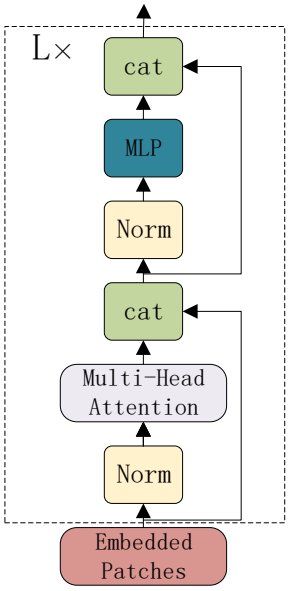


Figure 4. TransformerBlock structure diagram.

In this paper, the TI module incorporates a transformer Block. Its structure is illustrated in Figure 4. The primary components include a multi-head self attention (MHSA) mechanism and two fully

connected layers. Within the MHSA mechanism, each element of the input sequence is transformed into three matrices through different linear transformations: Q, K, and V. The attention weights are computed by taking the dot product of Q and K, followed by scaling and normalization. Then, these weights are used to weight the elements in V for each attention head. Finally, all head outputs are combined and linearly transformed to yield the final output. The MHSA outputs are concatenated and processed through fully connected layers to achieve the desired dimensionality.

3.1.2. InceptionDWConvolution

The structure of InceptionDWConvolution is depicted in Figure 5. InceptionDWConvolution has the capability to increase the network depth and width, as well as the adaptive learning ability, all without adding extra parameters or computational complexity. This enhancement allows for the improved differentiation between different types of defects during defect detection, thereby facilitating a more rapid and efficient feature extraction. As a result, the network effectively improves the precision, speed, and stability of steel defect detection.

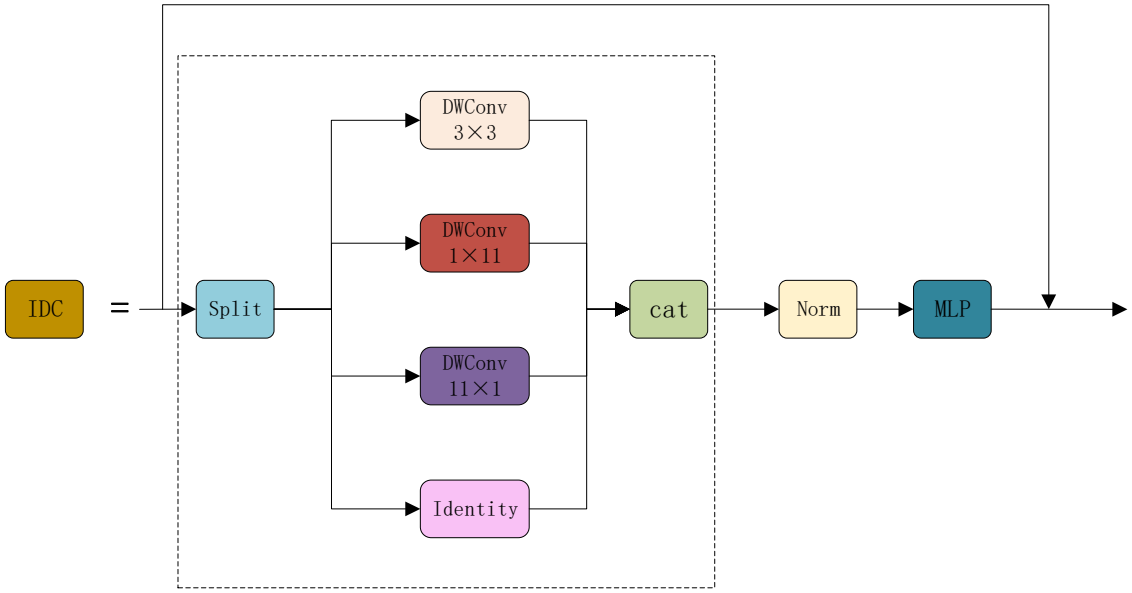


Figure 5. InceptionDWCevolution structure diagram.

In this paper, InceptionDWConvolution is used as a novel convolution operation, which combines ideas from both Inception and ConvNeXt. Its structure is depicted in Figure 5. InceptionDWConvolution divides the input channels into multiple groups and concatenates the outputs into three separate depth-wise separable convolutions: 3 × 3, 1 × 11, and 11 × 1, along with an identity path. By using different kernel sizes for convolution within various receptive field ranges, it improves the model's multi-scale object perception. This is particularly effective for object detection tasks, where the precise localization of objects of various sizes is crucial. Compared to the TransformerBlock, InceptionDWConvolution places a stronger emphasis on spatial awareness when processing image features, thereby compensating for the TransformerBlock's limitations in capturing spatial information. The TI module uses the InceptionDWConvolution module after the TransformerBlock,

thereby enriching the model's ability to express features at different scales and improving its adaptability and generalization.

3.2. SPPFCSPC

In this paper, the SPPFCSPC structure used is based on YOLOv6 3.0, and its diagram is depicted in Figure 6. This structure incorporates the spatial pyramid pooling with feature (SPPF) module into the original YOLOv7's SPPCSPC module. Through multiscale max pooling, the SPPF module increases the receptive field, thereby enhancing the detection capabilities for low-resolution images and small objects. A portion of the SPPCSPC module connects directly to the output, while another part passes through convolutional and pooling layers before connecting to the output. This architecture is designed to reduce parameters and computational load and aims to enhance feature diversity. SPPFCSPC removes the maximum pooling layer between the first and second convolutional layers, then halves the number of output channels in the first convolutional layer and doubles the number of input channels in the second convolutional layer. Moreover, while executing the three maximum pooling layers in parallel, it sets all three maximum pooling kernels to nine. This method effectively expands the receptive field to improve detection capabilities for low-resolution images and small objects while preserving the speed enhancements. Furthermore, it enhances the feature diversity, which enables the detector to capture more defect information and improves the detection accuracy.

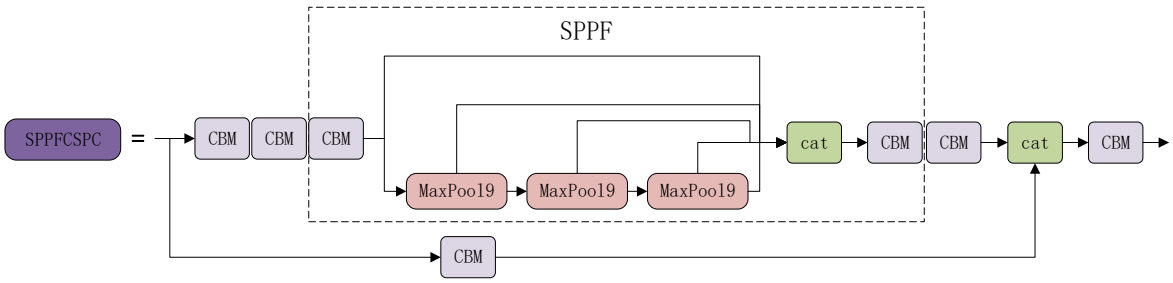


Figure 6. Structure diagram of SPPFCSPC.

3.3. GAM global attention mechanism

The structure of GAM is illustrated in Figure 7. The core idea behind GAM is to use a global pooling layer to capture the global information and then merge this global information with features from each spatial position. By leveraging the spatial and channel attention maps it generates, GAM applies weighting to the input feature maps. This process diminishes the significance of non-essential spatial and channel regions while elevating the importance of other critical areas. Consequently, any irrelevant information in the image is attenuated. Moreover, GAM augments its proficiency in detecting small defects by capturing cross-dimensional interaction features, thereby enhancing both its feature representation and its discriminative ability.

This module is applied in the FPN section of the network, thus allowing it to downplay any irrelevant information in the images during defect detection. It strengthens the network's ability to detect small target defects on the surface of steel by focusing on the relevant regions of interest.

In this context, M_C represents the channel attention feature map, and M_S represents the spatial

attention feature map. Given an input feature map (i.e., Input Feature1), it is multiplied with M_C to obtain the Input Feature2 feature map. Subsequently, InputFeature2 is multiplied with M_S to yield the output feature map (i.e., Output Feature). This process illustrates how channel and spatial attention feature maps are utilized to adjust the input feature map, thereby resulting in the generation of the final output feature map. This operation assists the network in better focusing on essential channels and spatial locations, thereby enhancing the model's performance and detection accuracy.

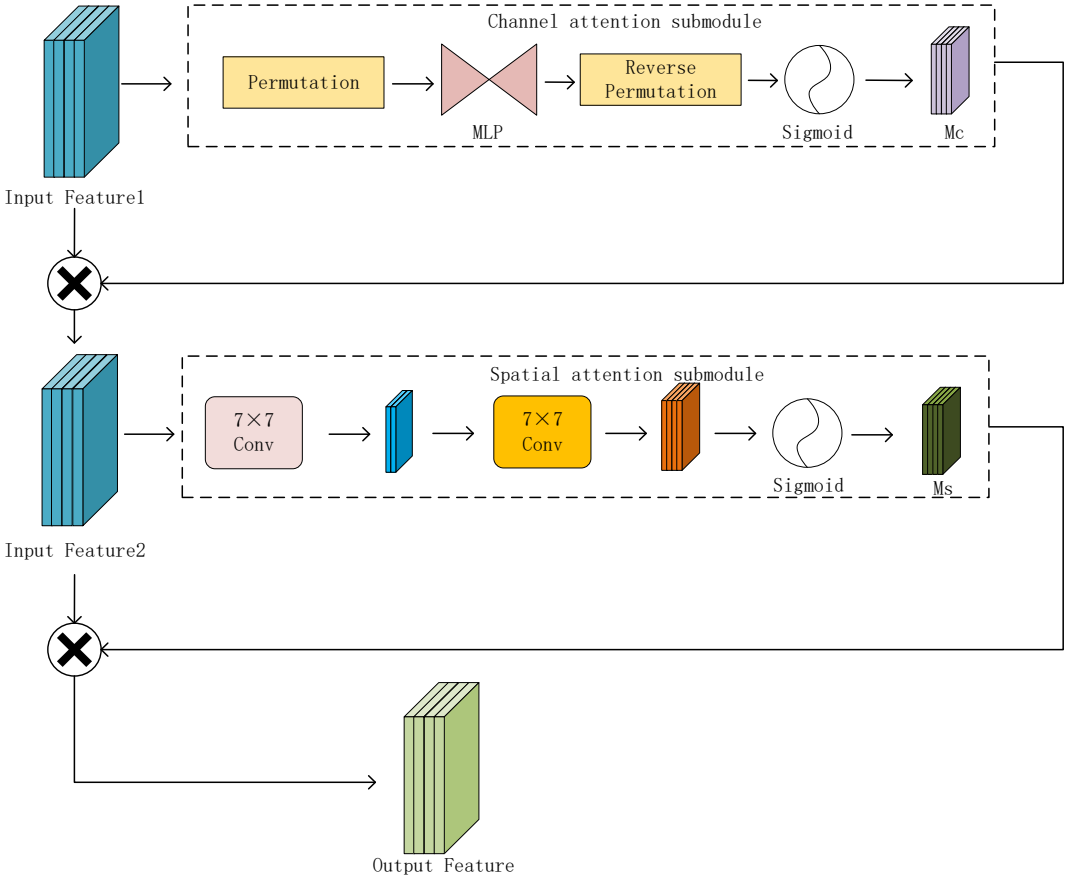


Figure 7. Structure diagram of GAM.

3.4. Mish

This paper replaces the originally used sigmoid linear unit (SiLU) activation function in YOLOv7 with Mish, which offers superior smoothness, self-stabilization, and generalization capabilities. Mish possesses characteristics such as an unbounded range, non-saturation, and self-regularization, thus leading to smoother gradients and functions. Being a non-monotonic function, Mish enhances the expressive power of neural networks, thus promoting feature diversity and complexity. This, in turn, improves the model's accuracy and generalization. The formula for Mish is as follows:

$$f(x) = x * \tanh(\ln(1 + e^x)), \tag{8}$$

where $\ln(1 + e^x)$ represents the normalized exponential function, and x represents the output.

3.5. MPDIoU

The positioning loss is achieved using the CIoU function in YOLOv7, which can't handle the mismatch between the predicted box and the real box in the direction. Therefore, this paper applies the minimum partial distance intersection over union (MPDIoU) loss function for positioning loss. The MPDIoU loss function reduces the minimum point distance between the predicted box and the ground truth box, thereby promoting the predicted box to closely approach the ground truth box. This loss function also maximizes the IoU, thereby aligning the position and shape of the predicted box with the ground truth box; this enhances the accuracy of the bounding box regression. First, MPDIoU calculates the minimum distance between the predicted box and the ground truth box; then, it utilizes this minimum distance to compute the similarity between the predicted box and the ground truth box. The formula is as follows:

$$d_1^2 = (x_1^B - x_1^A)^2 + (y_1^B - y_1^A)^2, \tag{9}$$

$$d_2^2 = (x_2^B - x_2^A)^2 + (y_2^B - y_2^A)^2, \tag{10}$$

$$MPDIoU = \frac{A \cap B}{A \cup B} - \frac{d_1^2}{w^2 + h^2} - \frac{d_2^2}{w^2 + h^2}. \tag{11}$$

In this paper, MPDIoU is used for localization loss and is calculated using the minimum distance between the predicted box B and the ground truth box A. The parameters d_1 and d_2 are utilized in computing this minimum distance.

(x_1^B, y_1^B) and (x_2^B, y_2^B) represent the coordinates of the top-left and bottom-right corners of the predicted bounding box B.

(x_1^A, y_1^A) and (x_2^A, y_2^A) represent the coordinates of the top-left and bottom-right corners of the ground truth bounding box A.

The MPDIoU loss function helps measure the similarity between the predicted and ground truth boxes by considering their minimum distance, which is determined using the d_1 and d_2 parameters.

The final definition of MPDIoU loss is as follows:

$$\mathcal{L}_{MPDIoU} = 1 - MPDIoU. \tag{12}$$

Utilizing the MPDIoU loss function for localization not only addresses the issue of misalignment between the predicted and ground truth bounding box in the direction, but it also streamlines the calculation process. This simplification leads to an improved efficiency and accuracy in defect detection. The MPDIoU loss function offers a practical solution for enhancing both the precision and the computational efficiency of the detection process, ultimately contributing to the overall effectiveness of the model.

4. Experiment and result analysis

4.1. Datasets

In this paper, we conducted experiments on the openly available NEU-DET dataset provided by Northeastern University. Some images from the dataset are shown in Figure 8. It consists of 1800 grayscale images, with 300 samples for each defect type on steel surfaces. It includes six types of

defects: crazing (Cr), inclusion (In), patches (Pa), pitted-surface (Pi), rolled-in-scale (Rs), and scratches (Sc). The images have a size of 200 × 200 pixels. In this paper, 1296 images were used as the training set, 144 images were used as the validation set and 360 images were used as the testing set. These sets were used for conducting ablation experiments and comparative studies.

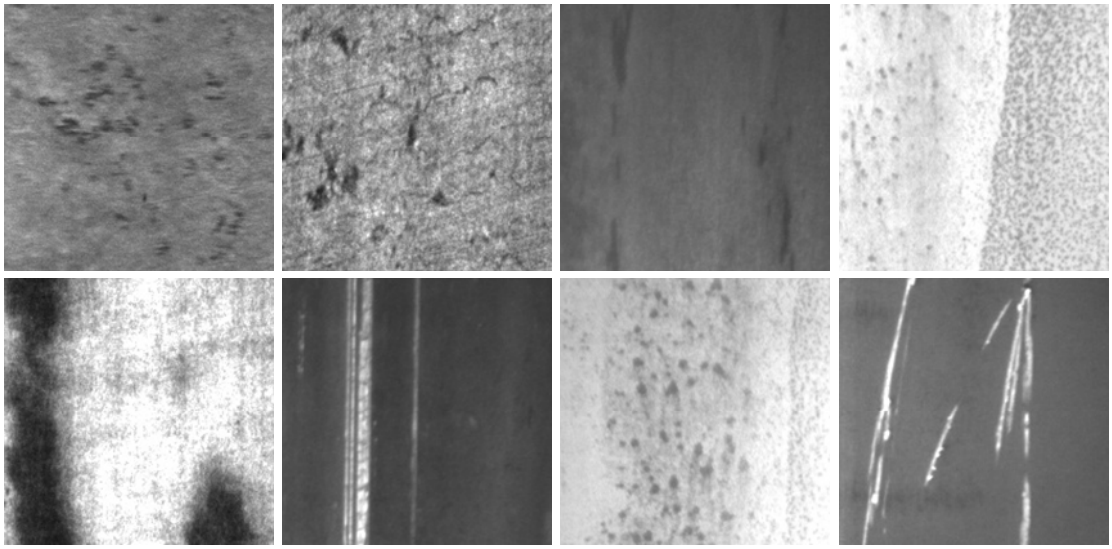


Figure 8. NEU-DET dataset.

Figure 9. VOC2012 dataset.

To assess the algorithm's generalizability, the paper also employed the PASCAL VOC2012 dataset for our experimentation. Some images from the dataset are shown in Figure 9. The VOC2012

dataset consists of 17,125 images and encompasses four major categories and 20 subcategories; 13,700 images were used as the training set, 1713 images were used as the validation set, and 1712 images were used as the testing set. These sets were used for conducting comparative studies. the VOC2012 dataset contains a greater number of categories compared to the NEU-DET dataset, thus providing a more comprehensive evaluation of the algorithm's versatility.

4.2. Settings

The experimental environment for this study included the Ubuntu 20.04 operating system, 50 GB of memory, Python version 3.9, PyTorch version 1.7.0, CUDA version 11.3, a single GPU with the NVIDIA Tesla V100 model, and PyCharm as the chosen development software. During the model training process, the parameters were set as indicated in Table 1.

Table 1. Hyperparameters setting.

Hyper parameters	Value
Learning rate	0.001
Momentum	0.937
Weight decay	0.0005
Optimizer	SGD
Batch size	16
Total epochs	300
Frozen epochs	50
Box	0.05
Cls	0.5
Cls_pw	1.0
Obj	1.0
Obj_pw	1.0

These hyperparameters provide the necessary settings for training and optimizing the object detection model on the specified hardware and dataset.

4.3. Evaluation metric

This paper employed precision, recall, average precision (AP), mean average precision (mAP) and FPS as evaluation metrics for the experiments. The formulas for these metrics are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

$$AP = \int_0^1 P(R)dR \tag{15}$$

$$mAP = \frac{\sum_{i=1}^{K} AP_i}{K} \tag{16}$$

$$FPS = FrameNum/ElapsedTime \tag{17}$$

TP represents the number of positive samples correctly predicted by the model, while FP signifies the count of negative samples that the model incorrectly identifies as positive, thus constituting false positives. On the other hand, FN represents the number of true positive samples erroneously classified as negative, thus giving rise to false negatives. Precision is the ratio of the number of true positive (TP) samples to the total count of positively identified images, while recall is the proportion of correctly identified positive samples to the total number of positive samples in the test set.

The AP is calculated for a single class, while the mAP is calculated across all classes. These metrics are employed to evaluate the overall detection performance across all target classes. Here, K represents the number of a specific class, and AP_i denotes the precision for class i.

FPS quantifies the rate at which the model detects images per second. A higher FPS value indicates that the model can process more images per second. FrameNum signifies the total number of images in the detection dataset, and ElapsedTime represents the overall runtime of the model's detection process.

4.4. Ablation

In the context of ablation experiments, the objective is to assess the practical value of the proposed improvements in steel surface defect detection. Each improvement or modification to the network model is experimentally evaluated to compare its impact. In this paper, the YOLOv7 model serves as the baseline, and a series of ablation experiments were conducted on the NEU-DET dataset. The results of these ablation experiments are summarized in Table 2.

Table 2. Results of ablation experiment.

YOLOv7	Mish	SPPFCSPC	GAM	TI	MPDIoU	mAP/%	AP/%					
							Cr	Rs	Sc	In	Pa	Pi
√						76.4	50.1	75.0	83.9	78.5	94.0	77.7
√	√					76.8	50.3	75.0	85.6	79.5	94.0	78.2
√	√	√				77.5	50.3	74.8	87.4	82.1	95.3	77.6
√	√	√	√			79.8	52.6	76.5	91.2	82.3	97.2	81.1
√	√	√	√	√		81.1	53.5	79.6	92.0	87.6	97.7	81.1
√	√	√	√	√	√	82.4	56.1	80.3	92.0	89.7	97.7	81.1

The results of the ablation experiments indicate the following findings:

1) When YOLOv7 was modified by solely replacing the activation function, there was a modest improvement in the detection performance for some types of defects, thus resulting in a 0.42% increase in the mAP.
2) The addition of the SPPFCSPC module to YOLOv7 enhanced the model's ability to perceive defects by improving feature point detection, thus leading to an improvement in the AP values for the Sc and In defects. However, there is a slight decrease in AP values for some defects,

thus resulting in a 1.03% increase in the mAP.

3) The incorporation of the GAM attention module resulted in a reduction of irrelevant information in images during defect detection. As a result, there was a significant improvement in the detection performance for all types of defects, with increased AP values for the Sc and Pa defects, this resulting in a 3.42% increase in the mAP.
4) The inclusion of the TI module improved the model's feature extraction capabilities, thus leading to a substantial 4.71% increase in the mAP. Notably, this improvement was particularly beneficial for detecting small target defects such as the SC, In, and Pa defects. Hence, the experimental results indicate that by including the TI module within the algorithm, there is a notably improved capability in detecting small targets compared to the original algorithm.
5) After incorporating the MDPIoU loss function, the algorithm showed an improvement in both the efficiency and accuracy of defect detection. Specifically, the algorithm achieved an increased AP value for the Cr defect, thus resulting in a 6.0% improvement in the mAP.
6) The proposed algorithm exhibited significant improvements in the AP values for small target defects during detection.

These findings collectively indicate that the modifications and enhancements proposed in this paper have a positive impact on the algorithm's ability to detect various types of defects, especially small target defects.

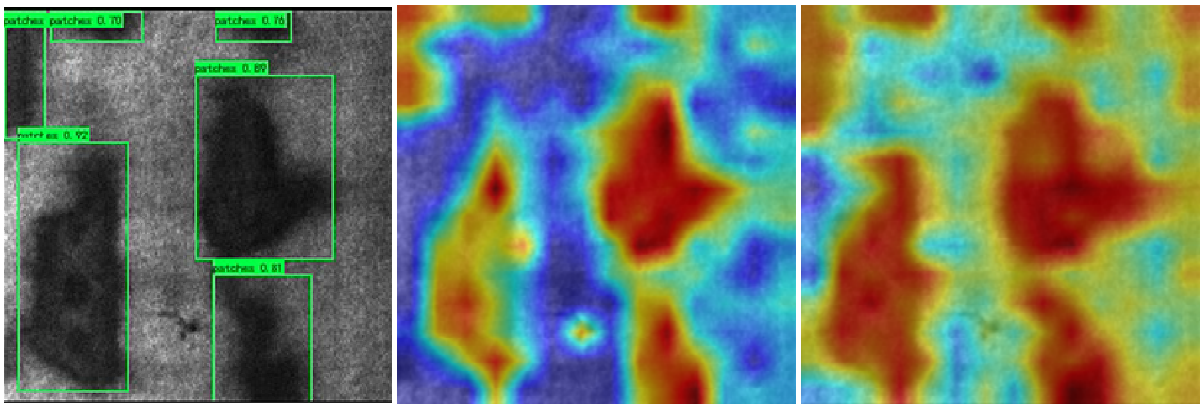


Figure 10. Visualization of TI module feature maps: ((a) Detection bounding box; (b) Feature map without TI module; (c) Feature map with TI module).

To illustrate the efficacy of the TI module, we employed heatmaps to visualize the feature maps before and after its integration, as depicted in Figure 10. For this investigation, we isolated a defect image of the Pa category. It is apparent that the network without the TI module predominantly directed its focus towards larger defects, while neglecting the smaller target defects. However, following the integration of the TI module, the network shifted its attention to prioritize the smaller target defects, thereby showcasing a more equitable distribution of attention across all defect types. The visualization outcomes effectively validate the efficacy of the TI module, thus affirming its role in augmenting feature extraction and ultimately enhancing the detection performance.

4.5. Contrast test

In the paper, under the condition of keeping all parameter settings unchanged, the algorithms [8–11] were initially trained alongside the proposed algorithm. Figures 11–15 show the training results for each algorithm on the NEU-DET dataset.

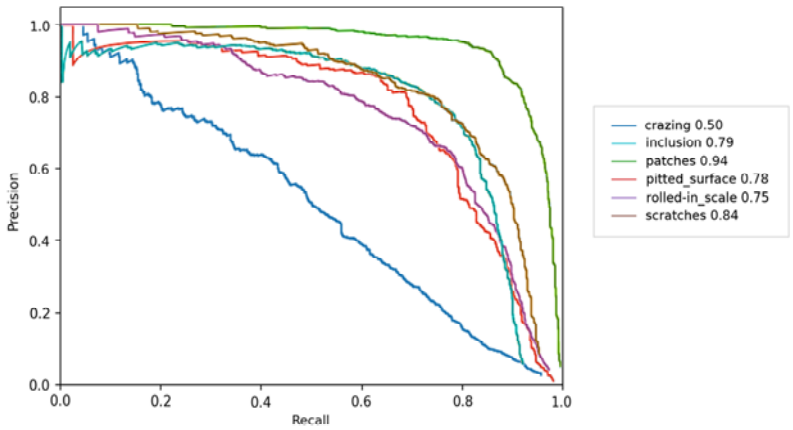


Figure 11. Training results of the original YOLOv7 algorithm.

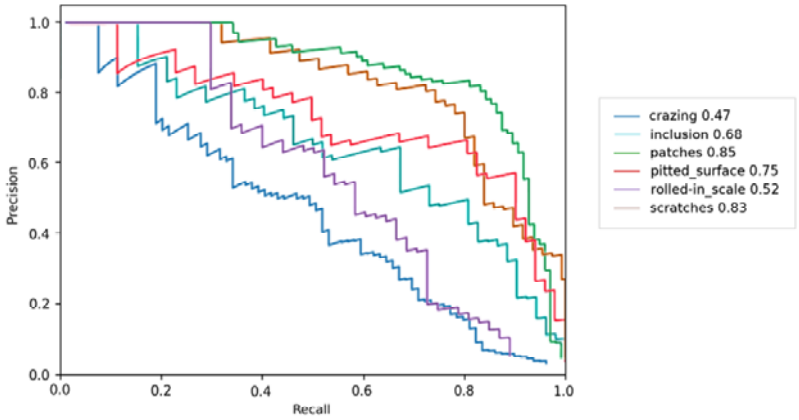


Figure 12. Training results of SSD algorithm.

Figure 16 displays partial detection results for defect samples, with confidence scores indicated in the upper-right corner of the bounding boxes. From Figure 16, it can be observed that defects, especially small target defects such as the Sc, In, and Pa defects are identified in the improved YOLOv7 network, while they can't be detected in YOLOv7, YOLOv5, and YOLOX. Furthermore, the confidence scores for these detections in the improved model are consistently higher compared to the original network model.

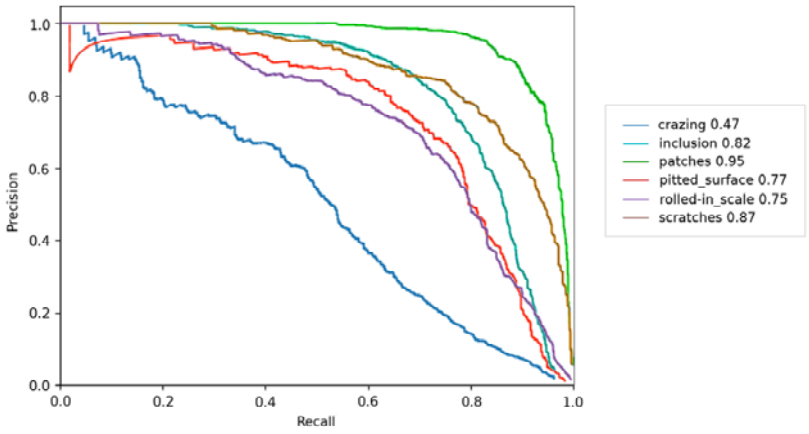


Figure 13. Training results of YOLOv5 algorithm.

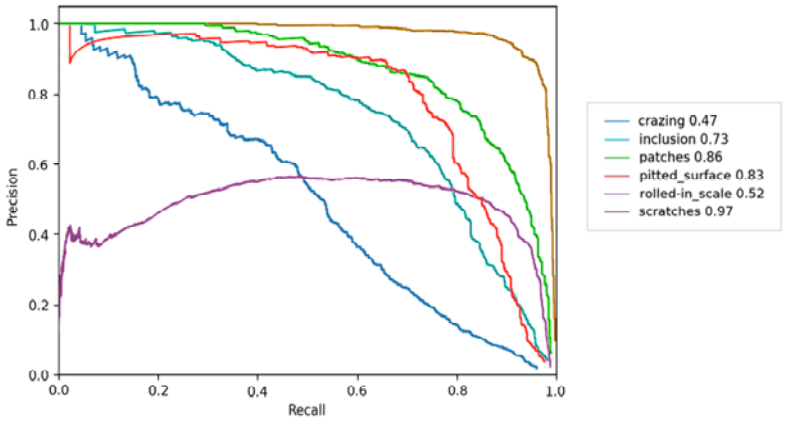


Figure 14. Training results of YOLOX algorithm.

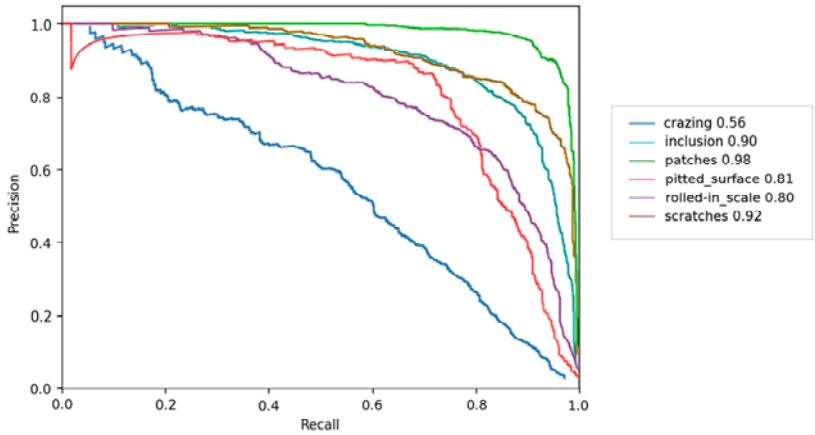
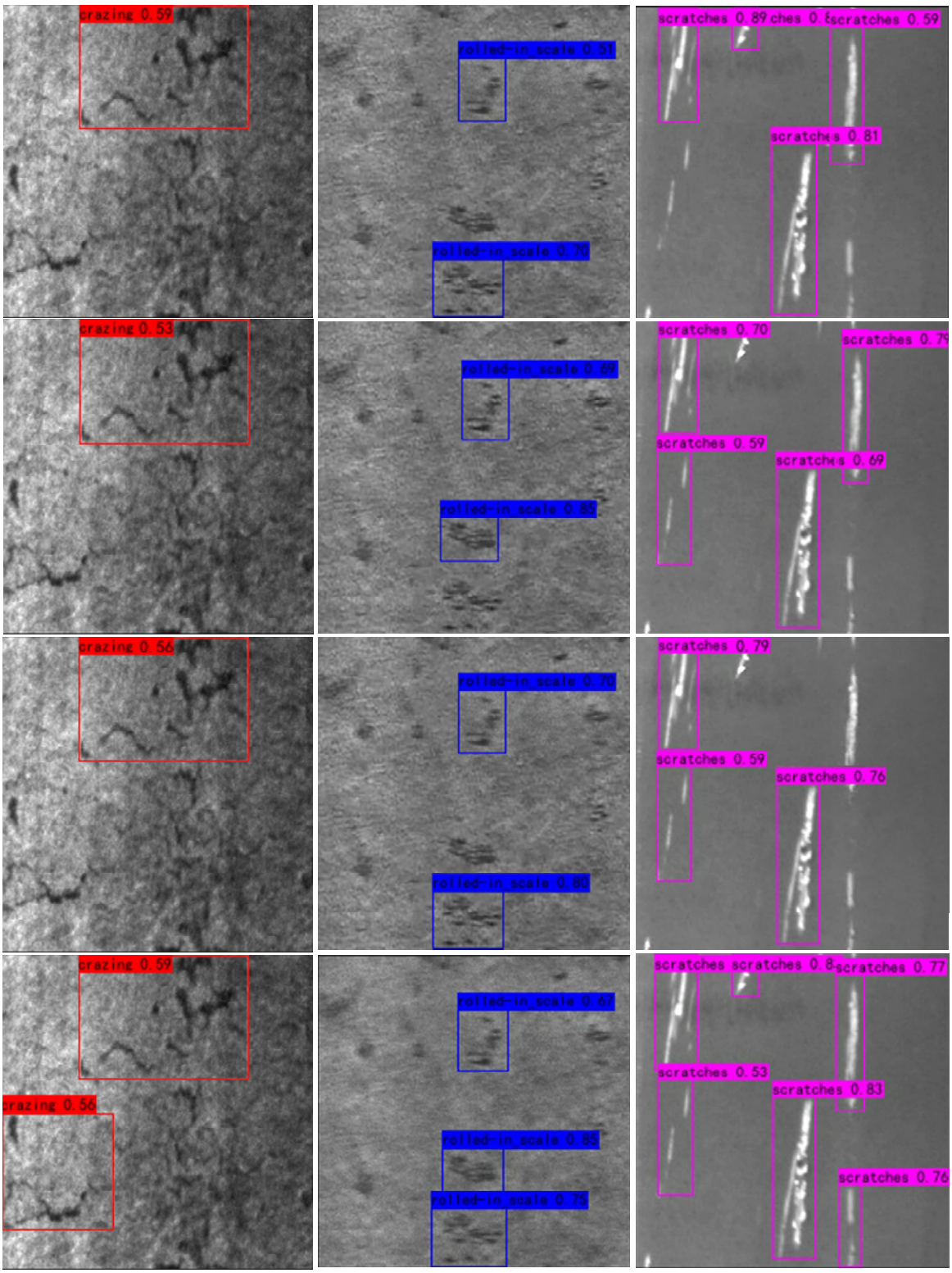
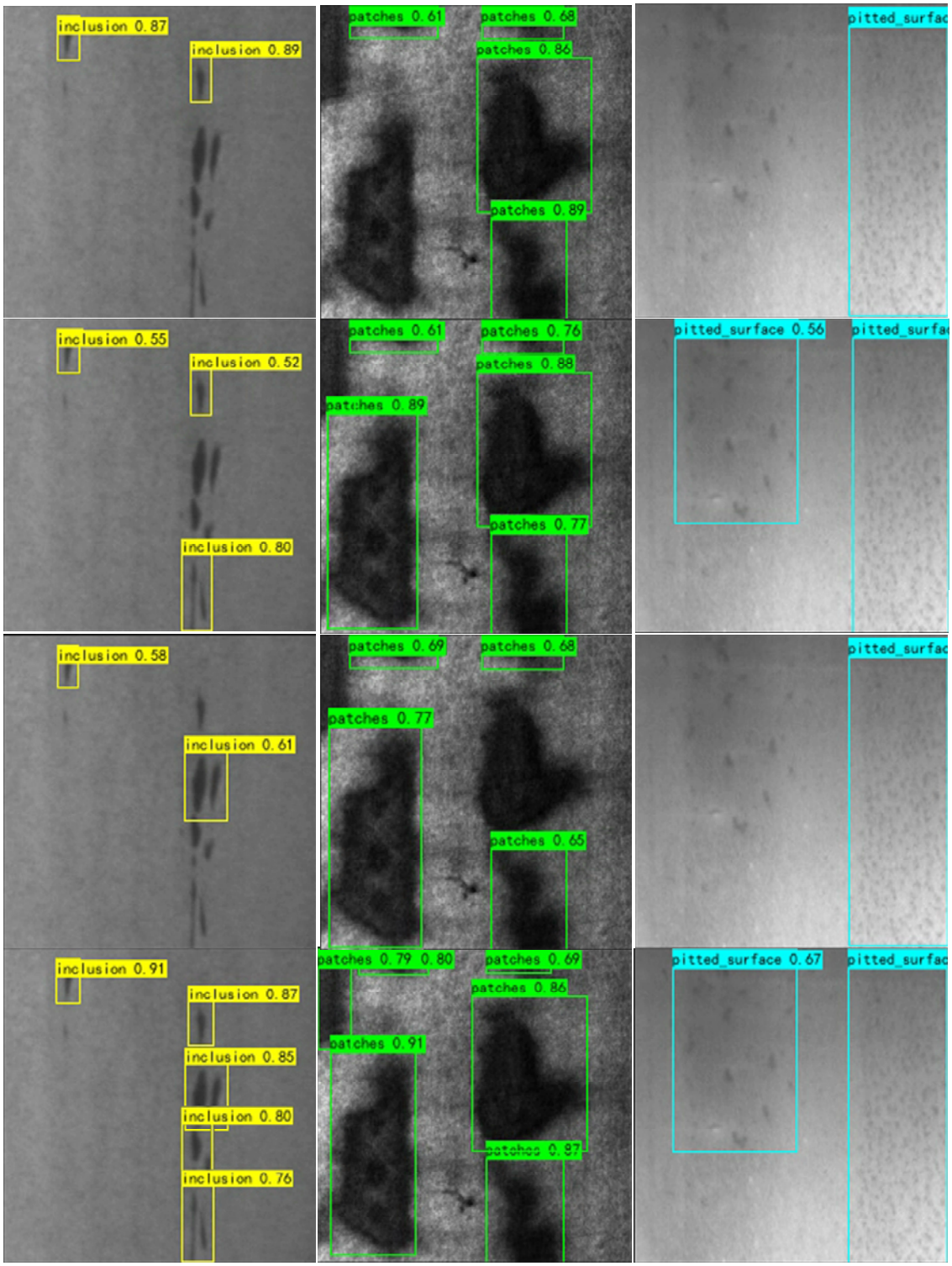


Figure 15. Training results of the proposed YOLOv7 algorithm.



(a) (b) (c)

Continued on next page



(d) (e) (f)

Figure 16. Visualization of the detection results of NEU-DET dataset, belonging to the categories: a) Cr; b) Rs; c) Sc; d) In; e) Pa; f) Pi, each column consists of 4 images, from top to bottom, displaying the results of 4 different algorithms: YOLOv7, YOLOv5, YOLOX, and Proposed YOLOv7.

We conducted rigorous comparative experiments between our improved YOLOv7 algorithm and other mainstream detection algorithms on the NEU-DET dataset, the results of which are shown in Table 3.

The experiments indicate that the proposed YOLOv7 algorithm performs favorably as compared to current mainstream detection models on the NEU-DET dataset. The results demonstrate that our method outperforms other detection models in AP for Cr, Rs, In, and Pa defects. However, AP values for the Sc and Pi defects are slightly lower than YOLOX. Despite not reaching an optimal performance across all defect types, our method has achieved a comparable level. Looking at the mAP, the improved YOLOv7 achieves the highest precision at 82.42%. This represents a 6% improvement as compared to the original YOLOv7 detection algorithm, thus highlighting the superior comprehensive detection accuracy of this network compared to the other methods mentioned. From the results, it can be concluded that the algorithm introduced in this paper experiences a decrease in speed due to the incorporation of the transformer module. However, it exhibits a certain degree of superiority in terms of the mAP and AP values for various defects.

Table 3. Comparison of experimental results on the NEU-DET dataset.

	mAP/%	AP/%						FPS
		Cr	Rs	Sc	In	Pa	Pi	
YOLOv7	76.4	50.1	75.0	83.9	78.5	94.0	77.7	49.3
SSD	68.6	47.3	52.6	83.2	68.7	85.1	75.0	46.1
YOLOv5	77.5	47.6	75.0	88.0	82.3	94.8	77.7	48.5
YOLOX	75.6	46.1	52.3	97.2	73.5	86.6	83.4	50.3
proposed YOLOv7	82.4	56.1	80.3	92.0	89.7	97.7	81.1	25.6

Table 4 displays the training results for both the YOLOv7 algorithm and the improved version on the VOC2012 dataset, using the same parameter settings (i.e., epoch set to 100).

Table 4. Comparison of experimental results on the VOC2012 dataset.

	mAP/%	Precision/%	FPS
YOLOv7	66.7	65.2	47.6
YOLOv5	67.2	67.5	46.3
YOLOX	67.5	67.2	49.6
proposed YOLOv7	69.3	68.9	26.8

When compared to the original YOLOv7, the improved algorithm has shown a significant increase in performance. The mAP value has risen from 66.7% to 69.3%, marking a 2.6% improvement, and the precision has increased from 64.2% to 68.9%, reflecting a 3.7% enhancement. In comparison to YOLOv5, there is also notable progress, with a 2.1% increase in the mAP and a 1.4% improvement in the precision. Compared to YOLOX, there is a 1.9% increase in the mAP.

4.6. Result analysis

We conducted disintegrative experiments on the enhanced algorithm, thus revealing that the TI

module outlined in this paper significantly amplifies the model's performance. This augmentation stems from the fusion of TransformerBlock's global modeling and InceptionDWConvolution's spatial perception capabilities. Notably, this improvement greatly enhances the precision of detecting minute defects in the Sc, In, and Pa categories within the imagery of these flaws. Consequently, when compared to the original algorithm, the experimental results underscore a substantial enhancement in the algorithm's ability to detect small targets following the integration of the TI module.

In our comparative analysis with other mainstream algorithms, our experiments demonstrate that our enhanced algorithm outperforms others notably in terms of the AP value for detecting the Cr defect, owing to the efficacy and accuracy enhancement attributed to the MDPIoU loss function in defect detection. This superiority arises from the augmented capacity of our algorithm in pinpointing small targets, thus resulting in elevated AP values for defects such as Rs, In, and Pa compared to alternative detection models. However, the inclusion of the TransformerBlock within the TI module leads to the construction of an attention matrix that yields a computational complexity of $O(n^2)$. Conversely, the InceptionDWConvolution employs grouped convolutions, thereby approximating a computational complexity of O(n). When amalgamating operations of differing computational complexities, the overall complexity tends to be dictated by the highest-complexity operation. In this context, the overall computational complexity may predominantly align with that of the TransformerBlock, thus resulting in an $O(n^2)$ complexity. This higher complexity has the potential to hamper operational speed, thus signifying considerable scope for enhancing efficiency.

Even though the proposed algorithm exhibits a lower FPS performance compared to other algorithms, experimental validation has affirmed its effectiveness, not only on the NEU-DET dataset, but also in its general adaptability.

5. Conclusions

In this paper, we have addressed the issue of steel surface defect detection by making several improvements to the YOLOv7 algorithm. These improvements include the following: the use of the Mish activation function to enhance the model's generalization and stability; the design of the TI module, which combines the Transformer module with InceptionDWConvolution to improve small object defect detection; the incorporation of the GAM attention mechanism to optimize the network structure and enhance the detection of small target defects; the introduction of the SPPFCSPC structure to maintain speed while improving network performance; and the utilization of the MPDIoU loss function for accurate localization. The experimental results of the enhancement algorithm demonstrate its superiority. Compared to other algorithms, the improved YOLOv7 network model exhibits significant performance improvements. On the neu-det dataset, it achieves a 6% increase in the mAP compared to the original algorithm. Additionally, on the VOC2012 dataset, there is a 2.6% improvement in the mAP. Notably, it excels in detection accuracy, particularly in the detection of small target defects. Meanwhile, through comparative experiments, it has been proven that although the method proposed in this article has improved the detection accuracy of some defect types compared to its comparative algorithms, there is still some room for improvement in detection accuracy, more specifically in terms of computing resources, computational complexity, and timeliness. The next step will be to further optimize the network structure, improve the detection accuracy of some defects in the network, and make lightweight improvements to the network. This will not only maintain the detection ability of the network, but also reduce its computational complexity and improve its

timeliness. Research will deploy the model on the mobile end to improve it in practical defect detection applications, which can be better applied in the steel production inspection process.

Use of AI tools declaration

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

Acknowledgments

This work was supported by the Liaoning Provincial Science and Technology Plan Project 2023JH2/101300205 and the Shenyang Science and Technology Plan Project 23-407-3-33.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. S. Mei, Y. D. Wang, G. J. Wen, Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model, *Sensors*, **18** (2018), 1064. http://doi.org/10.3390/S18041064
2. Z. Q. He, Q. F. Liu, Deep regression neural network for industrial surface defect detection, *IEEE Access*, **8** (2020), 35583–35591. http://doi.org/10.1109/ACCESS.2020.2975030
3. J. X. Luo, Z. Y. Yang, S. P. Li, Y. Wu, FPCB surface defect detection: a decoupled two-stage object detection framework, *IEEE Trans. Instrum. Meas.*, **70** (2021). http://doi.org/10.1109/TIM.2021.3092510
4. L. H. Shao, E. R. Zhang, Q. R. Ma, M. Li, Pixel-wise semisupervised fabric defect detection method combined with multitask mean teacher, *IEEE Trans. Instrum. Meas.*, **71** (2022). http://doi.org/10.1109/TIM.2022.3162286
5. M. Q. Chen, L. J. Yu, C. Zhi, R. Sun, S. Zhu, Z. Gao, et al., Improved faster R-CNN for fabric defect detection based on Gabor filter with genetic algorithm optimization, *Comput. Ind.*, **134** (2022). http://doi.org/10.1016/j.compind.2021.103551
6. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 27–30. http://doi.org/10.1109/CVPR.2016.91
7. J. Redmon, A. Farhadi, YOLO9000: Better, faster, stronger, in *30th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017), 21–26. http://doi.org/10.1109/CVPR.2017.690
8. J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, preprint, arXiv:180402767.
9. A. Bochkovskiy, C. Y. Wang, H. Y. M. Liao, YOLOv4: Optimal speed and accuracy of object detection, preprint, arXiv:200410934.
10. X. H. Qian, X. Wang, S. Y. Yang, J. Lei, LFF-YOLO: A YOLO algorithm with lightweight feature fusion network for multi-scale defect detection, *IEEE Access*, **10** (2022), 130339–130349. http://doi.org/10.1109/ACCESS.2022.3227205

11. N. Yang, W. Guo, Application of improved YOLOv5 model for strip surface defect detection, in *2022 Global Reliability and Prognostics and Health Management (PHM-Yantai)*, (2022), 1–5. http://doi.org/10.1109/PHM-Yantai55411.2022.9942194
12. Y. Wan, H. Y. Wang, Z. H. Xin, Efficient detection model of steel strip surface defects based on YOLO-V7, *IEEE Access*, **10** (2022), 133936–133944. http://doi.org/10.1109/ACCESS.2022.3230894
13. X. Wang, K. Zhuang, An improved YOLOX method for surface defect detection of steel strips, in *2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA)*, (2022), 152–157. http://doi.org/10.1109/ICPECA56706.2023.10075827
14. C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, et al., YOLOv6: A single-stage object detection framework for industrial applications, preprint, arXiv:220902976.
15. C. Y. Wang, A. Bochkovskiy, H. Y. M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2023), 7464–7475. http://doi.org/10.48550/arXiv.2207.02696
16. F. Akhyar, C. Y. Lin, K. Muchtar, T. Y. Wu, H. F. Ng, High efficient single-stage steel surface defect detection, in *16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, (2019), 18–21. http://doi.org/10.1109/AVSS.2019.8909834
17. V. Nath, C. Chattopadhyay, S2D2Net: An improved approach for robust steel surface defects diagnosis with small sample learning, in *IEEE International Conference on Image Processing (ICIP)*, (2021), 1199–1203. http://doi.org/10.26599/TST.2018.9010090
18. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, in *Advances in Neural Information Processing Systems*, **30** (2017). http://doi.org/10.1109/ICIP42928.2021.9506405
19. W. Yu, P. Zhou, S. Yan, X. Wang, Inceptionnext: When inception meets convnext, preprint, arXiv:230316900.
20. Y. Liu, Z. Shao, N. Hoffmann, Global attention mechanism: Retain information to enhance channel-spatial interactions, preprint, arXiv:211205561.

AIMS Press