Computing and Informatics, Vol. 43, 2024, 1352–1371, doi: 10.31577/cai_2024_6_1352

IMPROVED SWIN TRANSFORMER-BASED MODEL FOR HOT-ROLLED STRIP DEFECT DETECTING

Shenglong Hou, Hua He*, Kang Peng

College of Mathematics and Statistics
Shandong University of Technology
Zibo 255000, China
e-mail: h1227914623@163.com, huahe@sdut.edu.cn, kangpeng1219@163.com

Sibo Qiao

The College of Software
Tiangong University
Tianjin 300387, China
e-mail: siboqiao@126.com

Abstract. Hot-rolled steel strip plays an important role in the field of industrial manufacturing. In addition, defects on its surface affect the aesthetics of the subsequent products and their corrosion resistance, wear resistance, and fatigue strength. However, the existing methods are difficult to learn or capture discriminative feature representations, resulting in poor detection performance. Therefore, its surface defect detection faces two main challenges: one is the insufficient ability to extract local features, and the other is the limited ability to detect multi-scale targets. To address the above issues, we propose a Residual Deformable Convolution and Double LayerNorm Swin Transformer and Channel Expansion Feature Pyramid Networks (RTCN) multi-scale hot-rolled strip surface defect detection model, which adopts Double LayerNorm Swin Transformer (DLST) and as Residual Deformable Convolution Block (RDCB) its backbone network to increase the sensitivity of the model's detection of small and irregular defects. In addition, we adopt Channel Expansion Feature Pyramid Networks (CEFPN) to introduce more feature dimensions to better capture the structure and semantic image information. Ultimately, we assess the proposed model using the publicly available NEU-DET dataset. Our

* Corresponding author

comprehensive testing shows that the model developed in this paper beats the most advanced approach by 1.1 % to 7.2 % in mAP.

Keywords: Object detection, Swin transformer, NEU-DET, multi-scale targets

1 INTRODUCTION

Hot-rolled strip generally refers to coiled steel with a thickness of 1 mm to 20 mm and a width of 600 mm to 2 000 mm, which is widely used in industries such as automotive, electric motors, chemicals, and shipbuilding. However, due to factors such as processing methods, design inadequacies, equipment failures, and harsh operating conditions, hot-rolled strips and their products are prone to surface defects. Defect categories include rolled scale (RS), patches (Pa), cracks (Cr), pitted surface (PS), inclusions (In), and scratches (Sc), etc. Hot-rolled strip surface defect detection techniques now in use can be categorized into two categories: computer vision-based techniques and conventional techniques. Traditional surface defect detection methods for hot-rolled steel include manual detection [1], eddy current testing [2], infrared testing [3, 4], and magnetic flux leakage testing [5], etc. The manual detection method refers to the judgment of surface defects on hot-rolled strip made by detectors based on personal experience or evaluation criteria. Eddy current testing detects the presence of defects within a test object by utilizing the principle of electromagnetic induction, which involves detecting changes in the induced eddy currents. Infrared testing is a method based on the principle of infrared radiation, which scans and records temperature changes on the target surface to detect defects. Magnetic flux leakage testing uses the high magnetic conductivity of the strip surface to detect surface defects. However, these detection methods have disadvantages such as being susceptible to subjective influences, strict environmental requirements, low detection efficiency and false detection rate. In recent years, with the application of Charge Coupled Device (CCD) cameras and the development of computer vision technology, deep learning-based methods have emerged as preferred alternatives for defect detection in hot-rolled strip due to its high accuracy and fast speed [6, 7].

Because of their inductive bias, current mainstream deep learning target detection algorithms usually use Convolutional Neural Networks (CNNs) for feature extraction, which excel at capturing local patterns. But convolutional neural networks often fail to effectively extract global information, by introducing global information, the model can locate and identify the target more accurately. The self-attention mechanism introduced in Transformer [8] has been widely adopted by computer vision tasks to address the limitations of convolutional neural networks. It effectively captures long-range pixel relationships, emphasizes interconnections among different image regions, and integrates global image information [9, 10, 11]. Transformer-based methods have significantly enhanced object detection performance, but challenges in computer vision remain, including:

1. The performance of detecting minor defects is suboptimal, and the ability to collect local information is weak.
2. Transformer-based methods are mainly used for image classification and have lower accuracy in multi-scale object detection.

To address the above issues, Liu et al. [11] proposed the Swin Transformer, a versatile backbone network for computer vision. It restricts self-attention computation to non-overlapping local windows through sliding windows while also allowing cross-window connections, effectively improving detection efficiency and reducing computational complexity. Although the Swin Transformer has performed well in the area of computer vision, it still has limitations in capturing features of smaller targets, which is the problem that needs to be solved to detect defects in hot-rolled strip. Therefore, in this study the Swin transformer is used as the backbone network for improvement.

Hence, we propose a multi-scale RTCN surface defect detection model for hot-rolled strip. By incorporating the Residual Deformable Convolution Block into the Swin Transformer model, the improved RDCDLST model can extract a substantial amount of local information in the picture, better adapt to the target shape changes in the picture, and avoid information loss. Furthermore, to improve the performance of the model to detect defects of multi-scale, this paper also designs CEFPN, which increases the number of channels and introduces more dimension of features to better capture both structural and semantic features in the picture.

The most important contributions of this research are as follows:

1. In this paper, we propose the Double LayerNorm Swin Transformer (DLST), which normalizes the input features by using LayerNorm (LN) layers before the (S)W-MSA module and the MLP module, so that the input distributions in each layer have similar means and standard deviations. Reducing the bias of the input distribution reduces the risk of overfitting the model.
2. The backbone network proposed in this paper, called Residual Deformable Convolution and Double LayerNorm Swin Transformer (RDCDLST), cleverly integrates the advantages of deformable convolution and Swin Transformer, and combines RDCB with DLST to increase the adaptability and flexibility of the model, which is able to better capture the contextual information of the image, adapt to the geometric changes, and reduce the spatial bias.
3. This paper proposes the CEFPN network framework, which improves the expressive ability of the network by increasing the dimension of the feature map. By using cross-layer connections, it combines low-level detailed features with high-level semantic attributes to improve the detection accuracy of the model for targets at different scales.

2 RELATED WORK

2.1 Object Detection Method Based on Convolutional Neural Network

The methods for the detection of objects on the basis of the CNN can be classified into two types [12]. One is the two-stage detection method, and the R-CNN family of models exemplifies the advantages of this approach. These models first use a Region Proposal Network (RPN) to generate candidate regions, and subsequently classify and recognize these candidate regions [13]. The other one is the one-stage detection method represented by YOLO [14] and SSD [15], which avoids preliminary candidate regions and only needs one feature extraction to achieve object detection. Although the detection speed of the one-stage detection method is higher than that of the two-stage counterparts, the detection accuracy has decreased.

Wu et al. [16] used an improved Faster RCNN network to detect defects. Through the introduction of deformable convolution module, FPN multi-scale feature fusion module and CBAM attention module, the detection accuracy was effectively improved. Ye et al. [17] used the ChostNet network to replace the original feature extraction network in YOLOv4, which improves the feature extraction capability and reduces the model complexity at the same time. Chen et al. [18] used an improved SSD network to detect targets in different layers, demonstrating commendable robustness and adaptability. He et al. [19] proposed DDN (Defect Detection Network), which integrates multiple layers of features to determine defect class and position, enabling complete surface defect recognition on steel strip. Ding et al. [20] designed suitable anchor boxes through K-means clustering and introduced Feature Pyramid Network to enhance the fusion of low-level structural information, improving the detection accuracy of small defects. Dai et al. [21] proposed Deformable Convolutional Networks (DCN), which address the limitations of fixed shape sampling by adding an offset to the position of each sampling point in the convolutional kernel.

2.2 Transformer-Based Object Detection Method

Vaswani et al. [8] introduced the Transformer network model in 2017, based on the self-attention mechanism, which was subsequently implemented in natural language processing. Through its unique self-attention mechanism, the Transformer can effectively establish connections between distant targets, thus extracting more efficient feature information. End-to-end object detection with transformers proposed by Carion et al. [10] pioneered the application of Transformers in object detection. By inferring the relationship between pixels and combining with global image information, it directly and parallelly outputs the predictions. The Deformable DETR model proposed by Zhu et al. [22] combines DETR with deformable convolutions to make its self-attention module focus only on key sampling points, alleviating the slow convergence speed and high complexity issues of DETR. Vision Transformer (ViT) proposed by Dosovitskiy et al. [9] applies Transformer to image classification tasks successfully by dividing the image into multiple non-overlapping 16×16 patches and inputting them into the Transformer while trying to adhere to the original Transformer architecture as much as possible. However, this self-attentive mechanism, which is based within a fixed window, lacks information communication between different windows and can only be computed within a single window, severely limiting its performance. To solve this problem, the Swin Transformer [11] presented by Liu et al. segments the input image according to a fixed-size window, and performs the self-attention computation only in the divided local window, which effectively reduces the model complexity. In addition, Swin Transformer introduces a unique mechanism for computing self-attention in different windows, which establishes connections between windows and enhances the model's ability to perceive.

3 METHODS

The RTCN model consists of Residual Deformable Convolution and Double LayerNorm Swin Transformer (RDCDLST) and Channel Expansion Feature Pyramid Networks (CEFPN). The model inputs the image to the RDCDLST backbone network for feature extraction. After generating feature maps, they are channeled into the CEFPN for feature fusion. Finally, the model classifies the feature maps and then performs bounding box regression.

3.1 Residual Deformable Convolution and Double LayerNorm Swin Transformer

This paper proposes the RDCDLST network, which adds the residual deformable convolution module in each Swin Transformer Block and adds LayerNorm layers after W-MSA, SW-MSA, and MLP to increase the convergence speed during model training and to improve the accuracy of the feature extraction network for irregularly shaped defects. The RDCDLST network consists of Patch Merging layer, Residual Deformable Convolutional Module, and DLST module. The overall structure is shown in Figure 1.

The Swin Transformer starts by partitioning an input image of size of $H \times W \times 3$ RGB into multiple non-overlapping patches of equal size through the Patch Partition module, where each adjacent 4×4 pixels form a patch. The partitioned patches are flattened along the channel dimension. This operation results in a feature dimension of 48 for each patch ($4 \times 4 \times 3 = 48$). Therefore, after passing through the Patch Partition module, the image size changes from $H \times W \times 3$ to $H/4 \times W/4 \times 48$. The Linear Embedding module then projects the dimension to any dimension C. The Swin Transformer Block then receives the processed data. The first Block keeps the input size unchanged and forms Stage 1 together with the Linear Embedding layer. To generate hierarchical representations, as the depth of the network increases, each group of adjacent patches of size 2×2 is concatenated through the Patch Merging

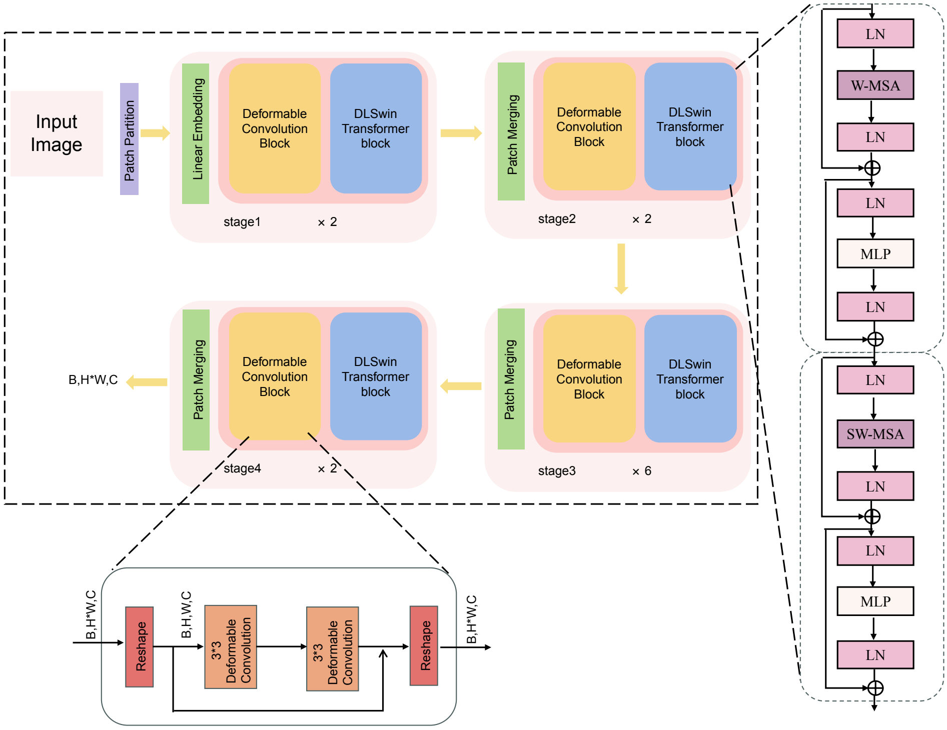


Figure 1. RDCDLST network model

module. This operation halves the resolution of each patch, while the dimension of each patch becomes four times larger. Thereafter, the dimension is reduced to 2C through a fully connected layer. The first Patch Merging module and Swin Transformer Block jointly form Stage 2. Repeating the process of Stage 2 multiple times yields Stage 3 and Stage 4. Each stage changes the dimension of the output, forming a hierarchical expression.

3.1.1 Residual Deformable Convolution Block

The Swin Transformer adopts a unique sliding window mechanism, but there is still a problem of missing defect information when extracting irregular-shaped features. Therefore, this paper introduces the Residual Deformable Convolution Block (RDCB) to adaptively handle the shape changes of target objects through deformable convolutions, thereby enhancing the model's perception of irregular objects and improving the network's capacity to learn from local correlations and irregular defects. In this study, the recognition accuracy of irregular defects is improved by inserting the RDCB into the Swin Transformer Block. The design of the RDCB is demonstrated in Figure 1.

The input image to the convolutional neural network is a three-dimensional tensor with a fixed size, i.e., the input size is (H, W, C). Where W and H denote the width and height of the image, and C indicates channel counts. In contrast, the Swin Transformer takes a two-dimensional tensor as in $(H \times W, C)$. Therefore, before inputting to RDCB for feature extraction, the output of the previous stage needs to be transformed into a three-dimensional tensor. Assuming that the input data has a two-dimensional tensor of dimensions $(H \times W, C)$, it is first converted to a three-dimensional feature map of dimensions (H, W, C). It is then fed into two 3×3 deformable convolution layers. To achieve fusion, the extracted features are combined with the input feature map via residual connections. Finally, the fused 3D feature maps are adjusted to a 2D tensor and fed into the next stage of the feature extraction module.

Deformable convolution includes two steps: 1) sampling on the feature map x using a fixed-size grid $\mathfrak{R}$, 2) performing a weighted sum of the sampled values. The grid $\mathfrak{R}$ defines the size of the receptive field. For each position y_0 on the output feature map p_0:

$$y(p_0) = \sum_{p_n \in \Re} w(p_n) \cdot x(p_0 + p_n), \tag{1}$$

where p_n represents $\mathfrak{R}$ the sampling position on the feature map.

Deformable convolution adjusts the sampling position by adding a position offset vector $\triangle p_n$ to the calculation formula of standard convolution $\{\triangle P_n | n = 1, 2, \ldots, N; N = |\Re|\}$. Substituting into formula (1), it is obtained that:

$$y(p_0) = \sum_{p_n \in \Re} w(p_n) \cdot x(p_0 + p_n + \triangle p_n). \tag{2}$$

The structure of deformable convolution is shown in Figure 2, where the input feature map extracts the offset of the deformable convolution through a convolutional layer. To elaborate, the number of input feature map channels is changed from N to $2N$ by a convolution operation, where the magnitude of the offset is denoted by N, and $2N$ denotes the offsets in both x and y directions needed to perform a translation in the plane. During the training process, the convolutional kernels that generate output features and those that generate offsets are learned concurrently, while the offsets are obtained through the backward propagation of the linear interpolation algorithm.

3.1.2 DL Swin Transformer Block

RDCB is added before the Swin Transformer Block in this paper, but RDCB has a notably high computational complexity. Therefore, in this paper, an LN layer is added after the W-MSA/SW-MSA and MLP layers to enhance the convergence rate of the model. In addition, the improved module ensures a consistent distribution of input features, reduces differences between different samples, making it easier for the model to perform effective feature extraction, improve the model's generalization

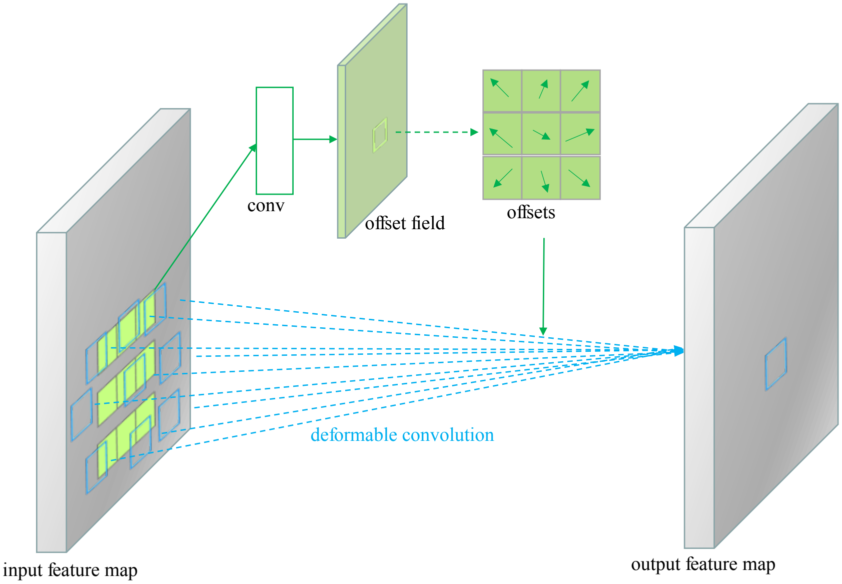


Figure 2. Structure of deformable convolution

ability, and prevalent training challenges such as gradient vanishing and exploding. Its formula is as follows:

$$\hat{Z}^l = \mathrm{LN}\left(W - \mathrm{MSA}\left(\mathrm{LN}\left(\hat{Z}^{l-1}\right)\right)\right) + Z^{l-1}, \tag{3}$$

$$Z^l = \mathrm{LN}\left(\mathrm{MLP}\left(\mathrm{LN}\left(\hat{Z}^{l}\right)\right)\right) + \hat{Z}^{l}, \tag{4}$$

$$\hat{Z}^{l+1} = \mathrm{LN}\left(\mathrm{SW} - \mathrm{MSA}\left(\mathrm{LN}\left(\hat{Z}^{l-1}\right)\right)\right) + Z^{l}, \tag{5}$$

$$Z^{l+1} = \mathrm{LN}\left(\mathrm{MLP}\left(\mathrm{LN}\left(\hat{Z}^{l+1}\right)\right)\right) + \hat{Z}^{l+1}, \tag{6}$$

where Z^l and Z^{l+1} represent the outputs of W-MSA/SW-MSA and MLP in the layer l.

3.2 Channel Expansion Feature Pyramid Networks (CEFPN)

In neural networks, deep feature maps usually contain richer global semantic information, commonly useful for detecting large targets, while shallow feature maps contain more local texture and structural information, pivotal for detecting smaller targets. Due to the receptive field of each convolutional layer in a convolutional neural network has a fixed size, detecting objects of different sizes effectively is difficult. Therefore, effective processing of multi-scale targets is a challenging problem in defect detection. A common practice to address this is to add a neck structure between the backbone network and the prediction layer to help integrate the flow of information [23]. Feature Pyramid Network (FPN) [24] is the most commonly used neck structure in the field of object detection. FPN constructs a top-down multi-scale feature pyramid, providing rich semantic information to capture target information at different scales. However, FPN introduces a certain degree of blurring or resolution reduction during the sampling operation, resulting in the loss of details and edge clarity. To address the issue of information loss, the CEFPN designed in this paper expands the dimensions of features at different levels, making the output feature maps of each level have the same number of channels to improve the utilization of shallow features. It also uses a bottom-up structure to fuse features of different scales to avoid information loss caused by upsampling.

Figure 3 shows the detailed structure of CEFPN, where C_{ij} represents the j^{th} dimension expansion of the input O_i. For input O_i with size $[H/4, W/4, N]$, after dimension expansion through C_{11}, C_{12}, C_{13} and C_{14}, the feature map channel count expands to 8 times the input channel count, and the final output size is $[H/4, W/4, 8N]$. For input O_2 with a size of $[H/8, W/8, 2N]$, first, the four outputs of O_2 and O_1 through C_{11}, C_{12}, C_{13} and C_{14} are fused, and then they are sent to C_{21}, C_{22} and C_{23} to expand the dimensions to 8N. The final output size is $[H/4, W/4, 8N]$. O_3, O_4 is similar to O_2. The output of the previous stage is fused with O_3 and O_4 and separately fed into C_{31}, C_{32} and C_{41} for dimension expansion, yielding output sizes of O_3 and O_4 are $[H/16, W/16, 8N]$ and $[H/32, W/32, 8N]$, respectively.

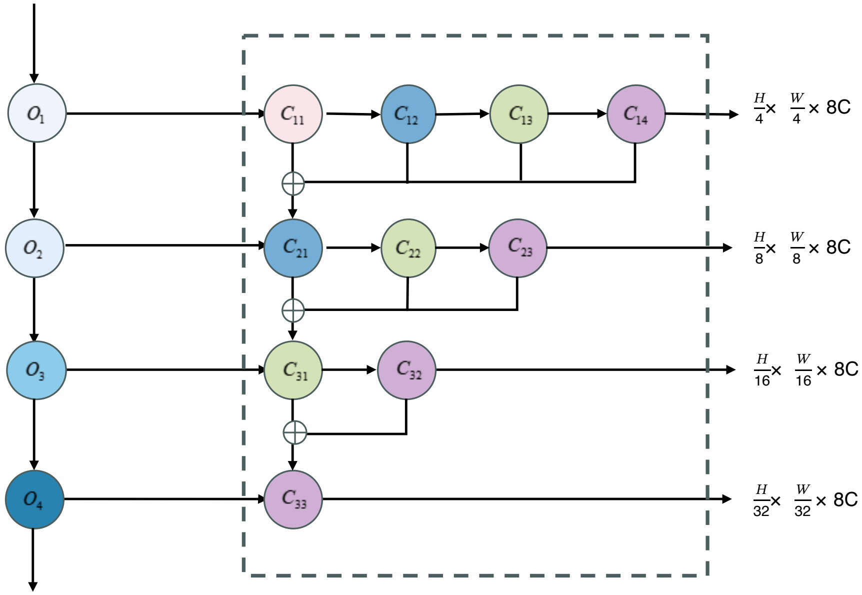

O_1
C_{11}
C_{12}
C_{13}
C_{14}
$\frac{H}{4}\times\ \frac{W}{4}\times 8C$
O_2
C_{21}
C_{22}
C_{23}
$\frac{H}{8}\times\ \frac{W}{8}\times 8C$
O_3
C_{31}
C_{32}
$\frac{H}{16}\times\ \frac{W}{16}\times 8C$
O_4
C_{33}
$\frac{H}{32}\times\ \frac{W}{32}\times 8C$

Figure 3. Structure of CEFPN

3.3 Residual Deformable Convolution and Double LayerNorm Swin Transformer and Channel Expansion Feature Pyramid Networks (RTCN)

In this paper, the combination of RDCDLST and CEFPN is proposed as the RTCN model for the detection of defects on the surface of hot-rolled steel strip, and the structure of the model is shown in Figure 1. The input of this model is an RGB image with dimensions $H \times W \times 3$, where H is the height of the image, W is the width, and the input image encompasses 3 channels. First, the image is reduced to 1/2 its original size by block partitioning using Patch Partition. Subsequently, the Linear Embeding layer is used to linearly transform the channel data of each pixel, and the number of channels changes from 48 to C. Then the feature map is input into Block1, which includes a residual deformable convolution module and a DL Swin Transformer module. Before inputting into the residual deformable convolution module, the $[H/4 \times W/4, C]$ 2D tensor needs to be reshaped into $[H/4, W/4, C]$ feature map. The feature map is then input into the residual deformable convolution module, and the output feature map is restored to its initial 2D tensor size before progressing to the DL Swin Transformer module. After the feature maps pass through the DLST and the residual deformable convolution module proposed in this paper, the size of the feature maps does not change, so the final output size of Block1 is $[H/4, W/4, C]$. In contrast to Block1, Block2 module just replaces Linear Embedding layer in Block1 to Patch Merging layer. Patch Merging takes pixels at the same position in each 2×2 neighbour image and gets four outputs. These four outputs are then concatenated in the channel direction and the outputs are then processed using a LayerNorm. Finally, the depth of the feature map is changed from C to $C/2$ by making a linear change in the depth direction through a fully connected layer. Therefore, after the Patch Merging layer, the height and width of the feature map are halved, while the depth becomes twice as before, resulting in an output size of $[H/8, W/8, 2C]$ for Block2. Block3 and Block4 have the same structure as Block2, except that Block3 contains six consecutive DL Swin Transformer modules. The four output features of Block1 to Block4 are obtained separately. Through the neck structure CEFPN proposed in this paper, the output features are fused at multiple scales to obtain four outputs. The sizes of the output feature maps are $[H/4, W/4, 8C]$, $[H/8, W/8, 8C]$, $[H/16, W/16, 8C]$, $[H/16, W/16, 8C]$. Finally, these four feature maps are directed to four detection heads for the detection process.

4 EXPERIMENTS AND RESULTS ANALYSIS

4.1 Dataset Introduction

The experimental data for surface defect detection in hot-rolled strip used in this paper are taken from the NEU surface defect database (NEU-DET) dataset [25, 26, 19], which was provided by Professor Song Kechen of Northeastern University.

Six different types of surface defects are included in this dataset: RS, Pa, Cr, PS, In, and Sc. There are a total of 1 800 defect images, as there are 300 images in each defect category, each 200 by 200 in size. There is an 8:1:1 split between the training, validation, and test sets. In addition, data quality techniques such as random cropping and horizontal mirroring are employed to increase the sample size. Some of the data are shown in Figure 4.

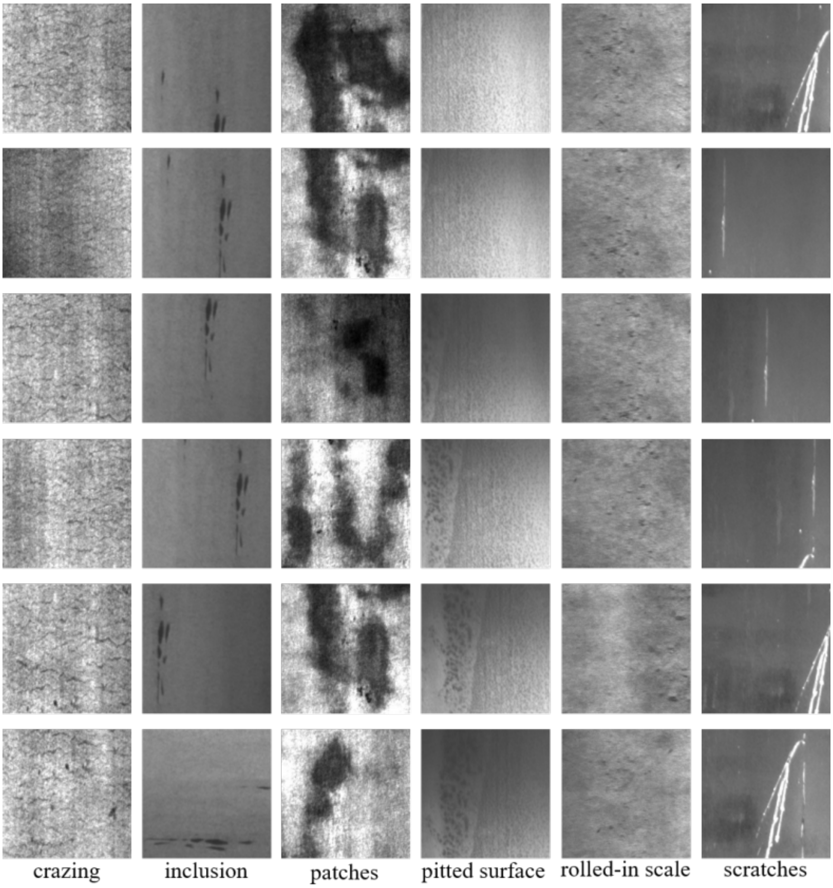


Figure 4. Structure of CEFPN

4.2 Evaluation Indicators

The primary evaluation indicator for network model performance in this paper is mean average precision (mAP). The relevant indicators that affect the mAP value are defined and calculated as follows.

As shown in formula (7), IoU or the intersection over union ratio between the predicted box and the ground truth box indicates the degree of overlap between the predicted box and the actual ground truth bounding box.

$$IOU = \frac{A \bigcap B}{A \bigcup B}. \tag{7}$$

TP (True Positive) refers to the number of positive samples predicted as positive. FP (False Positive) represents the number of negatives predicted as positive. If the IoU between the ground truth box and the predicted box is greater than 0.5, it is considered as TP; otherwise, it is considered as FP. FN (False Negative) represents the number of positives predicted as negatives. TN represents the number of negatives predicted as negative class.

P (Precision) represents the proportion of accurately predicted bounding boxes out of all the predicted boxes. It can be calculated using the formula (8) as follows:

$$P = \frac{TP}{TP + FP}. \tag{8}$$

R (Recall) is defined as the ratio of correctly predicted bounding boxes to all ground truth bounding boxes, and can be expressed using formula (9):

$$R = \frac{TP}{TP + FN}. \tag{9}$$

By plotting precision on the y-axis and recall on the x-axis, we can obtain the precision-recall curve, abbreviated as P-R curve. AP is the area bounded by the P-R curve and the axis of coordinates. Its calculation formula is shown as (10):

$$AP = \int_0^1 P(R)dR. \tag{10}$$

mAP represents the average value of all category APs, and a higher value for the mAP indicates a better detection performance of the model. Its formula is shown as (11):

$$mAP = \frac{1}{n}\sum_{i=1}^{n}(AP)_i. \tag{11}$$

4.3 Experimental Settings

Throughout the experiment, a computer equipped with an Intel i5-13490F CPU and an NVIDIA GeForce RTX 4070 (12 G) GPU serves as the hardware platform. PyTorch is employed as the deep learning framework, within a compilation environment consisting of Python 3.9.16 and PyTorch 1.13.1. In this paper, several classical object detection methods are used, including Faster R-CNN, YOLO V3, SSD, RetinaNet, etc. In addition, Resnet-50, the Swin Transformer and RDCDLST backbone network are used.

4.4 Ablation Experiment

To verify the impact of each component of the proposed method on the model performance, ablation experiments are conducted in this paper. This paper uses the NEU-DET dataset as the training data. The specific experimental content is as follows.

Indeed, the optimizer plays a crucial role in the experiments. This paper first conducts the experiments to investigate the choice of optimization parameters and learning rate. We consider common optimizers in computer vision, such as SGD [27] and AdamW [28]. The framework for this experiment is SSD, the backbone network of SSD is replaced by Swin-T, and the SGD and AdamW optimizers were used for the tests to find out the optimizer parameters that would lead to the highest performance of the model. To investigate the effect of the learning rate parameter set at the beginning of training the optimizer on the experiments, in this paper the same learning rate parameter is set for both the SGD and AdamW to validate the performance of the optimizers. To investigate the impact of parameters on the experiment, the learning rate before starting training is set as 0.0020, 0.0025, and 0.0030 for comparative experiments. The specific results of the experiments are described detail in Table 1.

Method	Opt	LR	mAP	mAP_{50}	mAP_{75}
Swin-T	SGD	0.0020	19.8	48.8	11.3
		0.0025	29.0	61.7	23.8
		0.0030	26.9	59.7	19.0
	AdamM	0.0020	24.6	56.5	15.6
		0.0025	24.2	55.2	16.3
		0.0030	13.8	37.1	7.2

Table 1. Opt and LR ablation experiment results

Table 1 shows that SGD has better performance than AdamW. By changing the optimizer, mAP can be improved by more than 4.4 %. The model achieved its highest level of detection accuracy when SGD was used for optimization and the initial learning rate was adjusted to 0.0025. Therefore, in this experiment, the SGD optimizer is used for model training with an initial learning rate of 0.0025 to ensure that Swin Transformer achieves better performance in detecting surface defects on hot-rolled strip.

In this work, the SSD serves as the main structure into which Swin-T and RDCDLST are inserted. Comparative experiments are then performed on the data set to confirm the effectiveness of the model in detecting hot-rolled steel strip.

Method	Backbone	Neck Structure	mAP	mAP_{50}	mAP_{75}
SSD	ResNet50	–	23.2	54.6	13.3
	Swin-T	–	29.0	61.7	23.8
	Swin-T	CEFPN	31.4	65.3	25.7
	RDCDLST	CEFPN	32.8	67.6	25.9

Table 2. mAP for comparative experiments with different strategies (%)

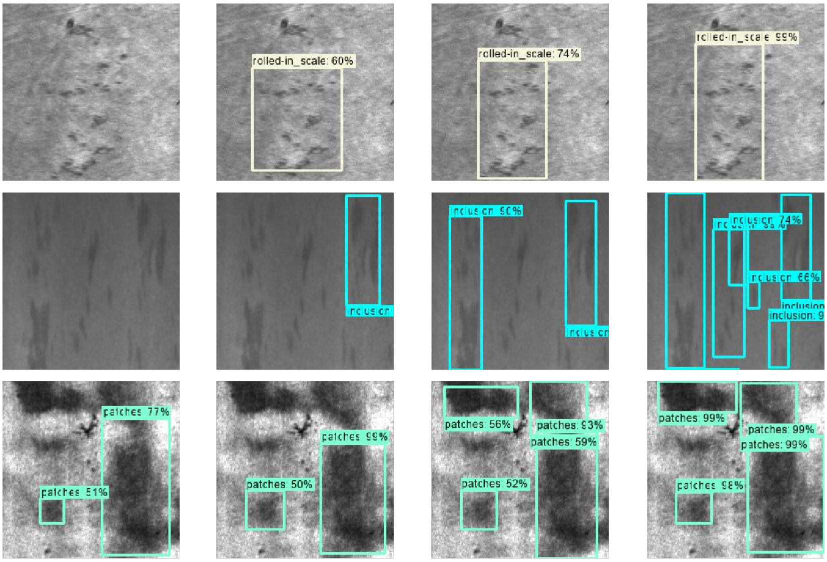
Table 2 shows that within the SSD framework, using Swin Transformer and RDCDLST as backbone networks has significantly improved the experimental results on various indicators compared to traditional convolutional neural network models. In particular, when compared with ResNet50, the experimental results using Swin-T as the backbone show a remarkable 5.8 % increase in mAP, highlighting the effectiveness of Swin Transformer. In addition, this paper introduces and compares the CEFPN neck structure to RDCDLST and Swin-T, respectively. Experimental results show that RDCDLST improves the ability to detect small defects: mAP increases by 0.8 %, mAP_{50} increases by 2.3 %, mAP_{75} increases by 0.2 %.

Figure 5 shows partial results of the recognition of superficial defects on hot-rolled strip using three backbones: ResNet50, Swin-T, and RDCDLST. Compared with traditional CNN networks, Swin Transformer focuses more on learning global features, especially the detection ability of large-scale defects has been significantly improved. Experimental results show that ResNet50 has issues such as low confidence and ineffective defect detection when detecting surface defects on hot-rolled strips. Swin Transformer demonstrates effective defect detection within the same detection area, resulting in increased confidence in detecting large-scale defects. After introducing the neck structure CEFPN proposed in this paper, the number of detected defects has increased, and the confidence in detecting defects of different scales has simultaneously improved. The proposed RDCDLST emphasizes the local features while at the same time giving due consideration to the global features, and differs from the Swin Transformer in this respect. Figures 11 c) and d) show that RDCDLST improves the ability to detect small defects, increases the number of defects detected, and significantly improves the confidence level of defect detection compared to the other method.

4.5 Experimental Model Evaluation

By comparing the model designed in this paper with commonly used object detection methods in the same dataset and experimental environment, Table 3 and Table 4 show the comparative experiment data.

To evaluate the detection performances of the method proposed in this paper, the RTCN model is evaluated against other widely used object detection methods (Faster RCNN, YOLOv3, SSD, and RetinaNet) using the NEU-DET dataset, and the experimental results are shown in Table 3. The proposed RTCN model achieves



(a)ResNet50 (b)Swin-T (c)Swin-T+CEFPN (d)本文

Figure 5. Prediction results

Model	Backbone	mAP	mAP$_{50}$	mAP$_{75}$
Faster RCNN	ResNet50	25.6	54.9	21.8
YOLOv3	Darknet53	26.8	57.2	21.3
SSD	ResNet50	23.2	54.6	13.3
RetinaNet	ResNet50	26.7	61.0	19.4
This Paper	EDCDLST	32.8	67.6	25.9

Table 3. Comparison of Mean Average Precision (mAP) of different models (%)

good detection results on the NEU-DET dataset, with mAP improved by 6.0 % to 9.6 %.

In addition, this paper also compares the mAP of five models under six defects, and the experimental results of different models under six types of defects are displayed in Table 4. In contrast to the other methods of object detection that are used in this paper, the proposed RTCN model achieves good performance improvement in multiple defect detection tasks, especially in the detection task of crazing, with a mAP accuracy improvement of 11.3 %.

5 CONCLUSIONS

The surface defect detection task for hot-rolled strip is an important research area for industrial development and it has great potential for use in a wide range of

Model	crazing	inclusion	patches	pitted_surface	rolled-in_scale	scratches
Faster RCNN	7.6	66.3	80.7	68.7	28.6	77.7
YOLOv3	15.3	54.5	79.3	68.7	43.7	82.0
SSD	22.2	57.4	79.2	72.3	33.4	62.9
RetinaNet	29.7	68.1	87.4	75.5	51.5	53.6
This Paper	41.0	69.7	87.8	72.4	55.1	79.5

Table 4. Comparison of Mean Average Precision (mAP) for comparative experiments with different strategies (%)

industrial scenarios. Firstly, we introduce Swin Transformer for detecting surface flaw on hot-rolled steel strip. Secondly, this paper improves swin transformers in accordance with the strengths and weaknesses of transformers and convolutional neural networks. Finally, this study designs the CEFPN network for the fusion of features of different scales for the improvement of the recognition accuracy. On the basis of the NEU-DET dataset, the following experimental conclusions may be drawn:

1. According to the experiment, the model with RDCDLST as the backbone has improved by 3.8 % in mAP, 5.9 % in mAP_{50}, and 1.9 % in mAP_{75}.
2. By innovative combination of the strengths of convolutional neural networks (CNN) and transformers in the local and global acquisition of information, the accuracy of the detection of small targets is significantly increased.

The research results show that the RTCN model provides a surface defect detection solution for hot-rolled strip that precisely meets industrial needs and has considerable application prospects for surface defect detection. However, the research that has been done in this paper to improve and increase the speed at which the model infers is not comprehensive. In the research work ahead, our focus will be on the elimination of the above limit.

REFERENCES

[1] Ye, X.: Research on Surface Defect Detection Algorithm of Hot-Rolled Strip Based on Deep Learning. Master Thesis. Wuhan University of Science and Technology, 2021, doi: 10.27380/d.cnki.gwkju.2021.000382 (in Chinese).

[2] Ahmed, R.—Sutcliffe, M. P. F.: Identification of Surface Features on Cold-Rolled Stainless Steel Strip. Wear, Vol. 244, 2000, No. 1-2, pp. 60–70, doi: 10.1016/S0043-1648(00)00442-7.

[3] Wei, J.—Liu, J.—He, L.—Wang, Y.—He, Y.: Recent Progress in Infrared Thermal Imaging Nondestructive Testing Technology. Journal of Harbin University of Science and Technology, Vol. 25, 2020, No. 2, pp. 64–72, doi: 10.15938/j.jhust.2020.02.009 (in Chinese).

[4] Arjun, V.—Sasi, B.—Rao, B. P. C.—Mukhopadhyay, C. K.—Jayakumar, T.: Optimisation of Pulsed Eddy Current Probe for Detection of Sub-Surface Defects in Stainless Steel Plates. Sensors and Actuators A: Physical, Vol. 226, 2015, pp. 69–75, doi: 10.1016/j.sna.2015.02.018.

[5] Tsukada, K.—Majima, Y.—Nakamura, Y.—Yasugi, T.—Song, N.—Sakai, K.—Kiwa, T.: Detection of Inner Cracks in Thick Steel Plates Using Unsaturated AC Magnetic Flux Leakage Testing with a Magnetic Resistance Gradiometer. IEEE Transactions on Magnetics, Vol. 53, 2017, No. 11, pp. 1–5, doi: 10.1109/INTMAG.2017.8007857.

[6] He, D.: Application of Deep Learning Method in Detecting Steel Surface Defects and Chars. Ph.D. Thesis. University of Science and Technology Beijing, 2021, doi: 10.26945/d.cnki.gbjku.2021.000018 (in Chinese).

[7] Chen, Y.: Research on Surface Defects Detection Theory and Recognition Algorithm of Steel Strip by Images. Ph.D. Thesis. China University of Mining and Technology, 2014 (in Chinese).

[8] Vaswani, A.—Shazeer, N.—Parmar, N.—Uszkoreit, J.—Jones, L.—Gomez, A. N.—Kaiser, Ł.—Polosukhin, I.: Attention Is All You Need. 2017, pp. 5998–6008, doi: 10.48550/arXiv.1706.03762.

[9] Dosovitskiy, A.—Beyer, L.—Kolesnikov, A.—Weissenborn, D.—Zhai, X.—Unterthiner, T.—Dehghani, M.—Minderer, M.—Heigold, G.—Gelly, S.—Uszkoreit, J.—Houlsby, N.: An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. 2021, doi: 10.48550/arXiv.2010.11929.

[10] Carion, N.—Massa, F.—Synnaeve, G.—Usunier, N.—Kirillov, A.—Zagoruyko, S.: End-to-End Object Detection with Transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J. M. (Eds.): Computer Vision – ECCV 2020. Springer, Cham, Lecture Notes in Computer Science, Vol. 12346, 2020, pp. 213–229, doi: 10.1007/978-3-030-58452-8_13.

[11] Liu, Z.—Lin, Y.—Cao, Y.—Hu, H.—Wei, Y.—Zhang, Z.—Lin, S.—Guo, B.: Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 9992–10002, doi: 10.1109/ICCV48922.2021.00986.

[12] Liang, T.—Chu, X.—Liu, Y.—Wang, Y.—Tang, Z.—Chu, W.—Chen, J.—Ling, H.: CBNet: A Composite Backbone Network Architecture for Object Detection. IEEE Transactions on Image Processing, Vol. 31, 2022, pp. 6893–6906, doi: 10.1109/TIP.2022.3216771.

[13] Ren, S.—He, K.—Girshick, R.—Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 39, 2017, No. 6, pp. 1137–1149, doi: 10.1109/TPAMI.2016.2577031.

[14] Redmon, J.—Divvala, S.—Girshick, R.—Farhadi, A.: You Only Look Once: Unified, Real-Time Object Detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[15] Liu, W.—Anguelov, D.—Erhan, D.—Szegedy, C.—Reed, S.—Fu, C. Y.—

Berg, A. C.: SSD: Single Shot MultiBox Detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.): Computer Vision – ECCV 2016. Springer, Cham, Lecture Notes in Computer Science, Vol. 9905, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.

[16] Wu, J.—Wang, J.—Fu, M.—Wang, Z.—Lu, Y.: Determination of Defect-Testing for Steel Strips Based on Improved Faster-RCNN Algorithm. Angang Technology, Vol. 56, 2022, No. 6, pp. 23–28 (in Chinese).

[17] Ye, Z.—Liu, M.—Zhang, F.: Small-Scale Defect Detection in Industrial Environments Based on Lightweight Deep Learning Networks. Control and Decision, Vol. 38, 2023, No. 5, pp. 1231–1238, doi: 10.13195/j.kzyjc.2022.1893 (in Chinese).

[18] Chen, J.—Liu, Z.—Wang, H.—Núñez, A.—Han, Z.: Automatic Defect Detection of Fasteners on the Catenary Support Device Using Deep Convolutional Neural Network. IEEE Transactions on Instrumentation and Measurement, Vol. 67, 2018, No. 2, pp. 257–269, doi: 10.1109/TIM.2017.2775345.

[19] He, Y.—Song, K.—Meng, Q.—Yan, Y.: An End-to-End Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features. IEEE Transactions on Instrumentation and Measurement, Vol. 69, 2020, No. 4, pp. 1493–1504, doi: 10.1109/TIM.2019.2915404.

[20] Ding, R.—Dai, L.—Li, G.—Liu, H.: TDD-Net: A Tiny Defect Detection Network for Printed Circuit Boards. CAAI Transactions on Intelligence Technology, Vol. 4, 2019, No. 2, pp. 110–116, doi: 10.1049/trit.2019.0019.

[21] Dai, J.—Qi, H.—Xiong, Y.—Li, Y.—Zhang, G.—Hu, H.—Wei, Y.: Deformable Convolutional Networks. 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 764–773, doi: 10.1109/ICCV.2017.89.

[22] Zhu, X.—Su, W.—Lu, L.—Li, B.—Wang, X.—Dai, J.: Deformable DETR: Deformable Transformers for End-to-End Object Detection. CoRR, 2020, doi: 10.48550/arXiv.2010.04159.

[23] Zhou, P.—Ni, B.—Geng, C.—Hu, J.—Xu, Y.: Scale-Transferrable Object Detection. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 528–537, doi: 10.1109/CVPR.2018.00062.

[24] Lin, T. Y.—Dollár, P.—Girshick, R.—He, K.—Hariharan, B.—Belongie, S.: Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944, doi: 10.1109/CVPR.2017.106.

[25] Song, K.—Yan, Y.: A Noise Robust Method Based on Completed Local Binary Patterns for Hot-Rolled Steel Strip Surface Defects. Applied Surface Science, Vol. 285, 2013, pp. 858–864, doi: 10.1016/j.apsusc.2013.09.002.

[26] Bao, Y.—Song, K.—Liu, J.—Wang, Y.—Yan, Y.—Yu, H.—Li, X.: Triplet-Graph Reasoning Network for Few-Shot Metal Generic Surface Defect Segmentation. IEEE Transactions on Instrumentation and Measurement, Vol. 70, 2021, pp. 1–11, doi: 10.1109/TIM.2021.3083561.

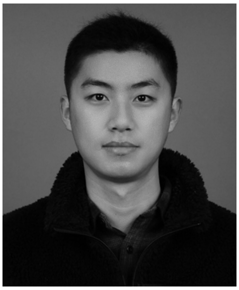
[27] Sinha, N. K.—Griscik, M. P.: A Stochastic Approximation Method. IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-1, 1971, No. 4, pp. 338–344, doi: 10.1109/TSMC.1971.4308316.

[28] Loshchilov, I.—Hutter, F.: Decoupled Weight Decay Regularization. In-

ternational Conference on Learning Representations (ICLR 2019), 2019, doi: 10.48550/arXiv.1711.05101.

Shenglong Hou graduated from the Shandong University of Technology in 2021 with a Bachelor's degree in statistics. He is now a graduate student of statistics at the School of Mathematics and Statistics, Shandong University of Technology. His research interests include machine learning and statistics.

Hua He is Associate Professor at the School of Mathematics and Statistics, Shandong University of Technology, Shandong, China. She received her Ph.D. in computer science from the Tianjin University. Her research interests include big data analytics, artificial intelligence, and Petri nets.



Kang Peng graduated from the Shandong University of Technology in 2021 with a Bachelor's degree in statistics. He is now a graduate student of statistics at the School of Mathematics and Statistics, Shandong University of Technology. His research interests include machine learning and statistics.

Sibo Qiao is working in the College of Software at Tiangong University. He received his Master's and Ph.D. degrees at the China University of Petroleum, Qingdao, China, in 2020 and 2023, respectively. His research interests include federated learning, deep learning, and image processing.