

Article

Surface Defect Detection of Steel Strip with Double Pyramid Network

Xinwen Zhou ^{1,*}, Mengen Wei ¹, Qianglong Li ¹, Yinghua Fu ² , Yangzhou Gan ³, Hao Liu ¹, Jing Ruan ⁴ and Jiuzhen Liang ¹

¹ School of Computer Science and Artificial Intelligence, Changzhou University, Changzhou 213164, China

² School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

³ Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China

⁴ Department of Mathematics, The Hong Kong University of Science and Technology, Hong Kong, China

* Correspondence: zhouxw@cczu.edu.cn

Abstract: Defect detection on the surface of the steel strip is essential for the quality assurance of the steel strip. Precise localization and classification, the two significant tasks of defect detection, still need to be completed due to the diversity of defect scales. In this paper, a residual atrous spatial pyramid pooling (RASPP) module is first designed to enrich the multi-scale information of the feature maps and increase the receptive field of the feature maps. Secondly, a double pyramid network (DPN) that combines RASPP and feature pyramid is proposed to fuse multi-scale features further so that similar semantic features are shared among the features of each layer. Finally, DPN-Detector, an automatic surface defects detection network, is proposed, which embeds the DPN module into Faster R-CNN and replaces the original detection head with a designed double head. Experiments are carried out on the steel strip surface defect dataset (NEU-DET), and the results show that the mAP of DPN-Detector is as high as 80.93%, which is 3.52% higher than that of the baseline network Faster R-CNN. The classification accuracy is 74.64%, and the detection speed reaches 18.62 FPS. The proposed method performs better robustness, classification and regression capability than other steel strip defect detection methods.

Keywords: defect detection; deep learning; residual atrous spatial pyramid pooling; feature fusion; attention mechanism; steel strip



Citation: Zhou, X.; Wei, M.; Li, Q.; Fu, Y.; Gan, Y.; Liu, H.; Ruan, J.; Liang, J. Surface Defect Detection of Steel Strip with Double Pyramid Network. *Appl. Sci.* **2023**, *13*, 1054. <https://doi.org/10.3390/app13021054>

Academic Editor: Xianpeng Wang

Received: 16 November 2022

Revised: 9 January 2023

Accepted: 11 January 2023

Published: 12 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Steel strips are an important industrial material in production and life. The defects on steel strips such as patch, scratch, rolled in scale, etc., are observed in the production process due to equipment, technology and production environment. The presence of these defects can seriously affect the quality and value of the steel strips.

The defects on the steel strip surface are very complex. As shown in Figure 1a, the contrast between some types of defects and the background is quite low in the grayscale image of the steel strip, and there are defects of multiple scales and shapes. Furthermore, there is a high degree of similarity between certain types of defects, and there is also a phenomenon of defect overlap, as shown in Figure 1b,c. Therefore, defect detection of steel strip surface is very challenging. In the early days, surface defects of steel strips were usually detected through human eyes. This method is quite costly and inefficient. Consequently, the automatic defect detection on the steel strip surface is essential to reduce costs and improve production efficiency [1].

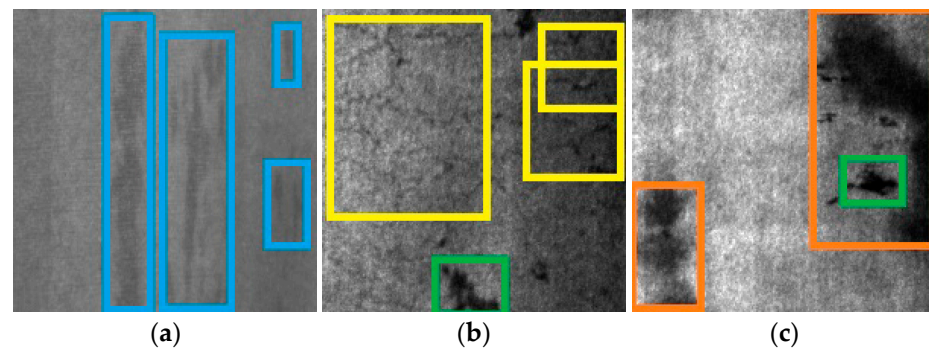


Figure 1. Complex defects on the steel strip surface, boxes of the same color represent the same defect type. (a) multiple defects; (b) multiclass defects; (c) overlapping defects.

The automatic defect detection methods based on computer vision have shown great advantages in the surface defect detection of steel strips [2,3]. Early automatic defect detection methods usually use feature extractors, such as histogram of oriented gradient (HOG) [4], local binary patterns (LBP) [5], etc. to extract image features. Classifiers such as support vector machines (SVM) [6] and random forest [7] are then used to classify and identify defects. However, those methods mainly rely on ingenious hand-crafted feature design. Additionally, they require a lot of professional knowledge and only work on specific scenes. Therefore, the generalization and robustness of those methods is poor.

Deep learning-based methods have been widely used in industrial inspection [8,9]. As one of the most successful algorithms of deep learning, the convolutional neural network (CNN) has achieved great progress in the computer vision area. Features with high abstraction and high representational capabilities can be extracted by CNN through learning the feature distribution of datasets with lots of samples. Compared with earlier feature extractors, the high-level features extracted by CNN have richer semantic information, which helps better classify and detect, and enhances the generalization and robustness of the model. However, some feature information, especially for minor defects, will be lost in the feature extraction process due to convolution and pooling operations by CNN. Therefore, maintaining the feature information of minor defects is very important for multi-scale defect detection during extracting high-level semantic feature information [10]. At present, only a single high-level feature is used for defections in many networks, resulting in poor detection for multi-scale and minor defects. The Feature Pyramid Network (FPN) [11] provides an idea to solve this problem. It fuses high-level semantic features and low-level coarse-grained features by adding a top-down path. The use of multi-level features in detection is beneficial for the detection of multi-scale and minor defects. In order to further shorten the information flow path from low-level features to high-level features, a bottom-up path augmentation [12] is proposed. It improves the utilization of rich location information in low-level features by reducing the number of convolutional layers that need to be traversed between low-level and high-level features. In addition, the atrous spatial pyramid pooling (ASPP) [13] also has a similar pyramid structure by arranging multiple convolutional layers with different dilation rates in parallel. It can learn more contextual information and expand the receptive field.

Even though there are a lot of excellent networks, it is still hard for the neural network models to learn defective features of a steel strip for the following two issues. The steel strip images are generally grayscale images captured by CCD cameras, and those grayscale images contain less information than color images. Furthermore, the defect positive samples are far less than the background parts. The attention mechanism in neural network models has the potential to solve this problem. The idea is to make CNN-imitated human functions to focus on the information of interest but ignore the irrelevant information when processing an image. Lots of research studies have been conducted and good results were achieved in this area. For example, SE-Net [14], CBAM [15], Non-local [16].

Inspired by the above-mentioned excellent work, a DPN module is presented in this paper to refine the feature maps of different scales generated by the backbone and fuse features of different scales. The problem of missing minor defect features as the network deepens and the detection difficulty caused by the diversity of defect scales will be resolved. Since in the proposed DPN module, the receptive field is expanded and the multi-scale information of the feature maps is enriched. Finally, an object detection network for surface defect detection of steel strips is proposed. The main contributions of this paper are as follows:

- (1) A novel residual atrous spatial pyramid pooling (RASPP) module is developed by exploiting multiple scale dilated convolutions to capture feature information and introduce residual connection. This module can enrich the multi-scale information of feature maps and expand the receptive field, while speeding up information dissemination.
- (2) A double pyramid network (DPN) is proposed to speed up the transfer of rich location information in the low-level features by embedding the RASPP module into top-down and bottom-up feature pyramid networks. DPN improves the robustness of the model to defect scales by integrating feature information of different scales multiple times and expanding the receptive field of the feature maps.
- (3) A steel strip surface defect detection network, DPN-Detector, is proposed, which is formed by embedding DPN into Faster R-CNN and replacing the original detection head with a designed double head. The double head contains a fully connected detection head biased towards classification and a convolution detection head focused on bounding box regression. Its embedded non-local attention blocks can enhance the defective features and provide high-quality localization information. DPN-Detector significantly improves the detection capability of multi-scale defects and can achieve the high-precision detection of steel strips.

The rest of this paper is organized as follows: Section 2 outlines related work. Section 3 details the proposed method. Section 4 presents extensive experiments and results analysis. Finally, the conclusion and prospect of this paper are given in Section 5.

2. Related Work

2.1. Automatic Defect Detection Method

In the traditional detection methods, manual features are usually designed to extract the texture, shape, edge, and other information of the images, and then classifiers are trained for defect detection. Song et al. [17] proposed enhanced local binary patterns (LBP) to resist noise, and utilized the nearest neighbor classifier (NNC) and the SVM for defect identification. Chu et al. [18] used four kinds of statistical features to extract the feature information of the steel strip images and used the enhanced SVM to detect defects in the samples and achieved multi-classification. Bai et al. [19] divided the detection process into two stages. The phase-only Fourier transform is used to generate saliency maps in the first stage, and the defects are extracted by using the saliency maps in the next stage. Although these methods have achieved good performance, they limit the application to specific situations and rely mostly on expertise. Therefore, they generally lack generalization and robustness.

CNN has created a craze in the computer vision area and achieved great success. Through the learning of a large amount of data, the feature information with deeper levels and stronger performance capabilities can be extracted by CNN. The good portability and generalization of CNN shows great application prospects in industrial detection. He et al. [20] proposed a multi-level feature fusion method to detect surface defects of steel strips. The method captures multi-scale information by fusing multi-level features, which improves the detection effect of multi-scale defects. However, they did not consider the importance of different levels of features when performing feature fusion. Cheng et al. [21] published a surface defect detection method for steel strips based on attention mechanism and feature fusion. This method uses an attention mechanism to reduce the loss of feature information and applies an ASFF module to fuse high-level features and low-level features.

However, this method has certain limitations when detecting dense defects. Bao et al. [22] proposed a few-shot segmentation method for metal surface defects segmentation. In this method, the defect segmentation problem is transformed into semantic segmentation of defect parts and background parts by using a triplet encoder. Multi-graph reasoning is then used to infer the similarity between different images for few-shot surface defect segmentation. Gao et al. [23] developed a deep learning-based detection method for surface defects of steel strips. Feature collection and compression modules are applied to enrich multi-scale information firstly in the method, and then a novel Gaussian weighted pooling method is employed to extract ROI features for classification and localization. Song et al. [24] used a saliency detection method for steel strip defects by combining a decoder and an encoder with an attention mechanism. This method can eliminate background noise and speed up the convergence of the model. Liu et al. [25] improved Faster R-CNN by using deformable convolution, balanced pyramid, and cascaded detection head to improve the defect detection of steel strips. However, most current CNN-based steel strip defect detection methods perform unsatisfactorily in multi-scale defect detection. Therefore, a novel method for multi-scale feature capture and fusion is proposed to address the above-mentioned problem in this paper.

2.2. Object Detector

The current object detectors can be divided into two-stage detectors and one-stage detectors according to different structures. The two-stage detectors generate region proposals in the first stage and then the corresponding features of the region proposals are used for classification and regression in the second stage. Generally, the two-stage detectors use a coarse-to-fine detection process. R-CNN [26] applies a selective search algorithm [27] to extract region proposals, and then the corresponding features of region proposals are extracted by CNN for identification and classification. Fast R-CNN [28] firstly extracts the features of the entire image, and then maps the region proposals obtained by the selective search algorithm with the extracted feature maps. It can avoid repeated feature extraction and significantly reduce processing time. However, Fast R-CNN still needs an additional search algorithm module to generate region proposals and cannot be trained end-to-end. Therefore, Faster R-CNN [29] employs a region proposal network (RPN) to directly generate region proposals, which can perform end-to-end training and detection. After the above improvements, the detection speed and accuracy of Faster R-CNN are greatly enhanced.

The one-stage detectors can be considered as regression analysis models, which directly predict the category and location of the objects. As a typical one-stage detector, YOLO [30] has good performance in detection accuracy and speed. SSD [31] improves the multi-scale objects detection ability of the model by using feature maps of different scales. RetinaNet [32] is proposed with a new loss function to address the imbalance of positive and negative samples during training.

One-stage detectors can detect fast, but the accuracy is relatively unsatisfactory. Comparatively, two-stage detectors exhibit high detection accuracy, but the amount of computation is slightly large. In the manufacture of steel strips, the high detection accuracy of defects can effectively prevent defective steel strips from entering the next manufacturing process. Therefore, a two-stage defect detection method is proposed in this paper to achieve high-precision detection of steel strip defects.

2.3. Attention Mechanism

When viewing an image, humans can quickly focus on important areas and ignore useless information. Based on this principle, attention mechanism is introduced into computer vision, which greatly improves the efficiency and accuracy of information processing. Wang et al. [16] designed a non-local attention module to compute the interaction between any two locations rather than adjacent locations. This module can capture long-term dependencies to maintain more information and increase the receptive field of the network. Tang et al. [33] employed a method based on attention mechanism and multi-scale pooling

for steel strip surface defect detection. Attention module was added to the backbone network to make the model focus on defect regions and suppress background regions. Additionally, a multi-scale pooling module is used to expand the receptive field. Su et al. [34] proposed a complementary attention mechanism to suppress the large amount of background noise in solar cell images and improve the network's attention to the spatial location of defects. Cheng et al. [21] brought up a method for surface defect detection of steel strips by applying differential channel attention and adaptive spatial feature fusion module.

In this paper, a non-local attention block is embedded in the proposed DPN-Detector to reassign the weights of defect and background regions. In this way, the features of the defective parts are highlighted, and the background is suppressed.

3. Methodology

The overall framework of DPN-Detector, the RASPP and the DPN will be introduced in detail in this section.

3.1. Defect Detection Framework

The overall framework of the DPN-Detector proposed in this paper is shown in Figure 2. The Faster R-CNN is used as the baseline detector for defect detection on steel strip surfaces. The DPN-Detector consists of four parts: feature extraction, feature refinement, region proposal, and classification and regression. The steel strip images input to DPN-Detector will go through the above four parts in turn, and finally the predicted defect category and location are obtained.

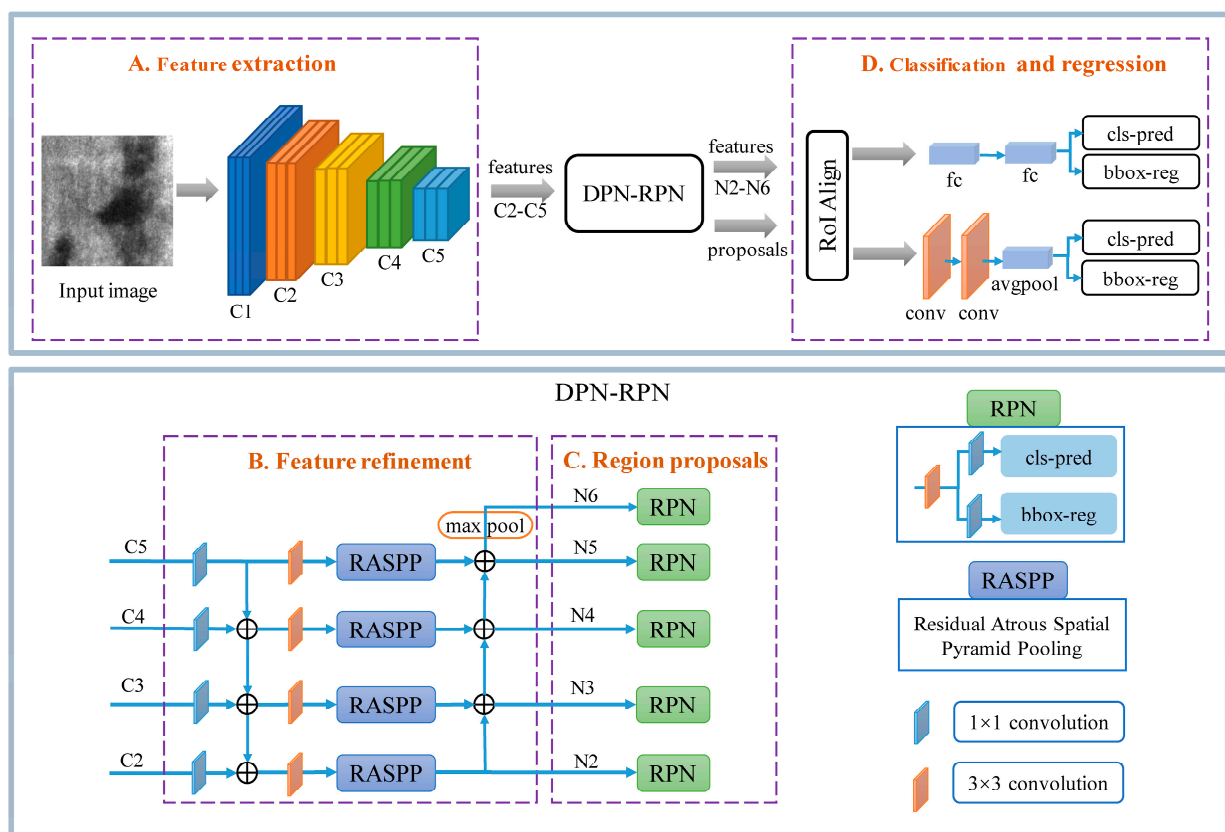


Figure 2. The overall framework of the proposed steel strip defect detection network (DPN-Detector). cls-pred: class prediction, bbox-reg: bounding-box regression. fc: fully connected layer, conv: convolutional layer.

In the feature extraction part, the superior performance ResNet50 [35] is used as backbone to extract the features of the steel strip images. The parameters are initialized

with the weights pre-trained in ImageNet, which can reduce the training time and make the overall model converge fast. In the feature refinement part, the features of different levels extracted by the backbone firstly pass the semantic information level by level through the top–down feature fusion path. The RASPP module is then used to enrich the multi-scale information of features and expand the receptive field. Finally, the bottom–up path is used to transfer the rich location information of the low-level features to the high-level features. In the region proposal part, the refined features of the DPN are fed into RPN to generate a large number of region proposals that may contain defects using sliding windows. In the classification and regression part, ROIALign firstly takes the refined feature maps and regional proposals as input and maps the regional proposals to the corresponding features of different levels. Then, the features are intercepted in the region candidate boxes and resized into fixed-size ROI feature vectors. Finally, the double head [36] is used as the detection head to perform defect category classification and bounding box regression according to the ROI feature vectors.

3.2. Residual Atrous Spatial Pyramid Pooling (RASPP)

We modified the ASPP module and propose the RASPP module to enlarge the receptive field and enrich the multi-scale information of feature maps. The relationship between the input features and output features can be adaptively learned by RASPP. In addition, it shortens the path of information propagation to reduce the risk of model overfitting.

The ROI features intercepted by ROIALign only contain feature information in the region proposals, and the information outside will be discarded regardless of whether it contains the object feature or not. However, the region proposals generated by RPN are not accurate, resulting in inaccurate ROI features, which affects subsequent classification and regression. In the case of a large receptive field, the limited defect features intercepted by ROIALign can still contain discarded feature information outside the region proposals to ensure the subsequent accurate defect detection. Moreover, the larger receptive field also helps the RPN to capture large defects. Therefore, a RASPP module is designed to expand the receptive field and enrich multi-scale information that lays the foundation for obtaining more accurate ROI features.

As shown in Figure 3, the input of RASPP passes through four parallel branches. The resulting outputs are then concatenated together along the channel dimension and finally fused with the input to form the output of RASPP. Convolution layers are used in three branches, and the size of the convolution kernel is (1, 3, 3). The input channel and the output channel are consistent, and the ReLU function [37] is used to activate the output. The dilation rates of the convolutional layers are set to (1, 3, 6), respectively, to capture multi-scale feature information. The stride of all dilated convolutions is 1. The padding is set to (0, 3, 6), respectively, to keep the feature map size unchanged. Global average pooling is adopted to extract global semantic information in the pooling branch. The feature vector generated after pooling is used to further extract useful information by a 1×1 convolution and ReLU, then the vector is concatenated with the feature maps of the other three branches in the channel dimension after resizing. At this time, the channels of the concatenated feature maps are four times that of the RASPP input feature maps. A 1×1 convolution is then used to reduce the channel dimension of the output feature maps to be consistent with the input feature maps. Finally, the refined features are linked with the input feature through a skip connect to form the final output of the RASPP module. The calculation process of RASPP is defined as follows:

$$\begin{aligned} F_{c1} &= \text{conv}_1^1(R_{in}), F_{c2} = \text{conv}_3^3(R_{in}) \\ F_{c3} &= \text{conv}_3^6(R_{in}), F_p = \text{conv}_1^1(\text{avgpool}(R_{in})) \end{aligned} \quad (1)$$

$$F_{concat} = \text{conv}_1^1(\text{concat}(F_{c1}, F_{c2}, F_{c3}, F_p)) \quad (2)$$

where R_{in} represents the inputs features of the RASPP module, F_{c*} represents the feature map after convolution, and F_p is the feature map after pooling. The i in con_j^i is the atrous rate and the j is the convolution kernel size.

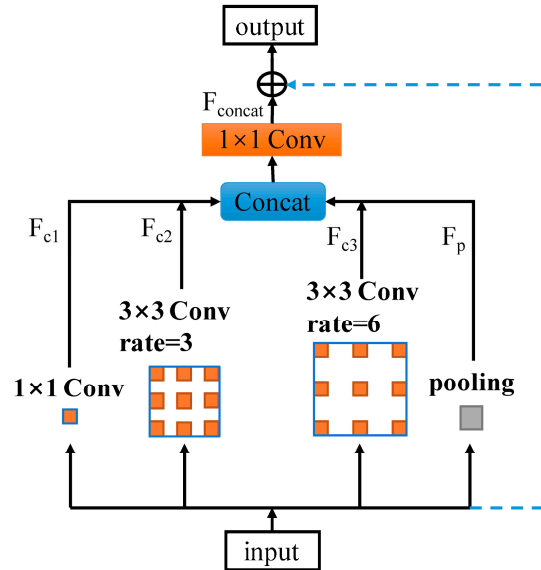


Figure 3. The structure of the proposed residual atrous spatial pyramid pooling (RASPP) module. 1×1 Conv: Convolution with a kernel size of 1, 3×3 Conv rate = 3: Dilated convolution with a convolution kernel size of 3 and an expansion rate of 3, other similar.

The output of the RASPP module is defined as:

$$R_{out} = \alpha \cdot F_{concat} + R_{in} \quad (3)$$

where R_{out} represents the output features of the RASPP module. It is noted that we multiply the concatenated feature map F_{concat} with a scale parameter α . Additionally, α is initialized to 0 and gradually increases during the learning process to assign more weights. The RASPP module can enrich the multi-scale information of the features and increase the receptive field of the features, and also shorten the path of information flow, and enhance the robustness of the model.

3.3. Double Pyramid Network (DPN)

DPN module is designed to obtain more informative ROI features, solve the problem of diverse defect scales and ineffective detection by single-layer features. It is formed by the combination of RASPP and top-down and bottom-up feature pyramid networks (PA-Net). The module can expand the receptive field of the model and fuse feature information of different scales. It is essential for ROI feature capture.

The novel DPN module designed in this paper is shown in Figure 4. The feature maps of C2, C3, C4, and C5 levels extracted by the backbone is the input of DPN module. Firstly, a 1×1 convolutional layer is employed to reduce the channel dimension of the feature maps extracted by the backbone to 256. This can reduce the amount of computation and facilitates the fusion of features at different levels. Secondly, the top-down path is used to transfer the high-level semantic information to the low-level features, and a 3×3 convolutional layer also is utilized to reduce confusion caused by feature fusion. Thirdly, the RASPP module is used to expand the receptive field and enrich the multi-scale information of the feature maps. Finally, the bottom-up path is used to transfer rich location information of the low-level feature to the high-level feature. The DPN algorithm is shown in Algorithm 1. It is worth noting that the RASPP module is applied in all layers of DPN for more efficient extraction of multi-scale feature information. However, some extra features are useless and

counterproductive. Therefore, the weights of the RASPP module are shared to reduce the parameters and complexity of the model. DPN enriches the multi-scale information of the features and expands the receptive field and improves the utilization of the rich location information in the low-level features by combining RASPP with top-down and bottom-up feature pyramids. In addition, the features of each layer have similar semantic features in DPN, which is beneficial for the model robustness to multi-scale defect detection.

Algorithm 1 DPN in DPN-Detector.

Input: Feature map $C_i, i \in (2, \dots, 5)$

Output: Feature map $N_i, i \in (2, \dots, 6)$

```

1:                                      $C_i \leftarrow \text{Conv1}(C_i)$  for  $i = 2, \dots, 5$ 
2:                                     last  $\leftarrow C_5$ 
3:                                     for  $i = 4 \rightarrow 2$  do
4:                                       lateral  $\leftarrow C_i$ 
5:                                       top-down  $\leftarrow \text{Upsample}(\text{last})$ 
6:                                       last  $\leftarrow \text{lateral} + \text{top-down}$ 
7:                                        $P_i \leftarrow \text{Conv3}(\text{last})$ 
8:                                     end for
9:                                     for  $j = 2 \rightarrow 4$  do
10:                                      if  $j == 2$  then
11:                                         $N_j \leftarrow \text{RASPP}(P_j)$ 
12:                                      else
13:                                         $N_j \leftarrow \text{Conv3}(\text{RASPP}(P_j)) + \text{Downsample}(N_{j-1})$ 
14:                                      end if
15:                                    end for
16:                                      $N_6 \leftarrow \text{Maxpool}(N_5)$ 
17:                                     return  $N_i, i \in (2, \dots, 6)$ 

```

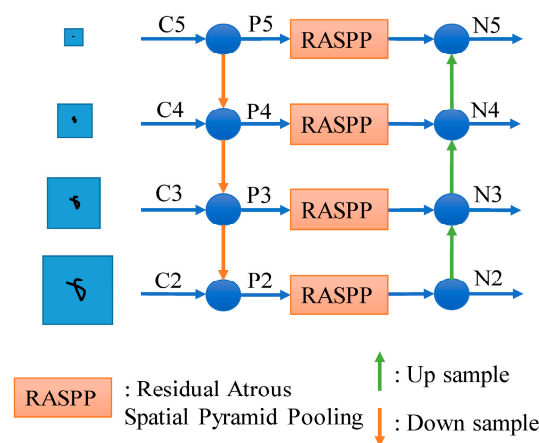


Figure 4. The structure of the proposed double pyramid network (DPN). $C_i/P_i/N_i$ represents features of different scales.

3.4. Double Head Module

Previous work shows that the fully connected detection head (fc-head) and the convolutional detection head (conv-head) have different functional focus [36]. Therefore, applying different detection heads for different tasks is beneficial for obtaining more accurate classification and regression results. In our double head module, the fc-head is used for the classification task while the conv-head is used for the regression task.

The structure of the detection head we adopt is shown in Figure 5. The ROI features extracted by ROIAlign is the input of the double head module. The ROI features are fed into two branches for classification and regression. In the classification branch, we follow the design of Faster R-CNN, in which two fully connected layers with 1024 dimension

are shared before the two output layers of classification and regression. In the regression branch, a residual block is used to increase the feature map channel from 256 to 1024, and then two bottleneck blocks are used to extract location information. Each bottleneck block is followed by a non-local attention block for enhancing defective features and suppressing background noise. The introduction of non-local attention block can cope the detection challenge caused by the high similarities between the steel strip defects and the background. Finally, average pooling is used to generate 1024-dimensional feature vectors for class prediction and bounding box regression.

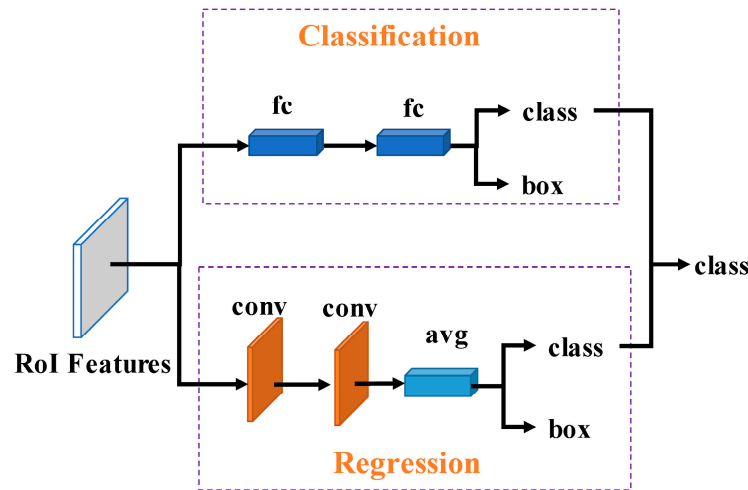


Figure 5. The structure of the double head module. fc: fully connected layer, conv: convolution layer.

Notably, classification and regression are performed in both branches. While giving full play to the advantages of the two detection heads, additional supervision information is added, and they complement each other to achieve better performance. It has been reported that stacking more non-local attention blocks can achieve better performance [36]. We stack two non-local attention blocks, which are stacked alternately with bottleneck blocks to balance the accuracy and speed. Other settings are referred to the previous publication [36].

After the classification and regression tasks are divided and conquered, the loss function also includes two parts, namely fc-head loss, and conv-head loss. Additionally, each detection head includes both classification loss and bounding box regression loss. The mathematical expressions are:

$$L^{fc} = \lambda^{fc} \cdot L_{cls}^{fc} + (1 - \lambda^{fc}) \cdot L_{reg}^{fc} \quad (4)$$

$$L^{conv} = (1 - \lambda^{conv}) \cdot L_{cls}^{conv} + \lambda^{conv} \cdot L_{cls}^{conv} \quad (5)$$

where L^{fc} is the loss of fc-head, and L^{conv} is the loss of conv-head. λ^{fc} is the weight coefficient for balancing the classification loss and bounding box regression loss in L^{fc} . Similarly, λ^{conv} is the weight coefficient for balancing the classification loss and bounding box regression loss in L^{conv} .

The total loss function is:

$$L = \omega^{fc} \cdot L^{fc} + \omega^{conv} \cdot L^{conv} + L^{rpn} \quad (6)$$

where ω^{fc} and ω^{conv} are the weight coefficients of L^{fc} and L^{conv} , respectively. L^{rpn} is the loss of RPN. In both fc-head and conv-head, the cross-entropy function [38] is used in the classification loss, and the smooth L1 function is used in the bounding box regression loss.

Different detection heads in the double head can capture different information, and the function of the classifiers of the two detection heads can achieve better defect classification. Mathematical expression is shown as:

$$S = S^{fc} + S^{conv} \cdot (1 - S^{fc}) = S^{conv} + S^{fc} \cdot (1 - S^{conv}) \quad (7)$$

where S^{fc} and S^{conv} are the classification scores of fc-head and conv-head, respectively.

4. Experiments

Extensive experiments are conducted to verify the performance of the proposed DPN-Detector. The dataset, implementation details, evaluation metrics, experimental evaluation, and ablation experiments are described in detail below.

4.1. Dataset

In this paper, the steel strip dataset NEU-DET [20] is used to verify the performance of the proposed DPN-Detector. The annotation type of the NEU-DET dataset is bounding box. Each annotation including a category label and the coordinates of the two points of the box (top left and bottom right), is stored in XML. There are six types of defects, namely crazing (Cr), inclusion (In), patches (Pa), pitted surface (PS), rolled in scale (RS), and scratches (Sc). There are a total of 1800 images, 300 images for each defect. The resolution of each image is 200×200 . We divide the dataset into training and testing sets, where the training and the testing sets contain 1260 images and 540 images, respectively. The distribution instances of the six defect types in the training and testing sets are visible in Table 1. Furthermore, in order to prevent overfitting, the training dataset is expanded by six times through different operations such as random horizontal or vertical flipping, cropping, zooming, and translation. Figure 6 shows some typical defect annotations.

Table 1. Defect instance distribution of six defect types in train and test sets.

Type	Cr	In	Pa	PS	RS	Sc
Test	197	306	252	136	189	161
Train	2947	4219	3762	1776	2633	2319
Total	3144	4525	4014	1912	2822	2480

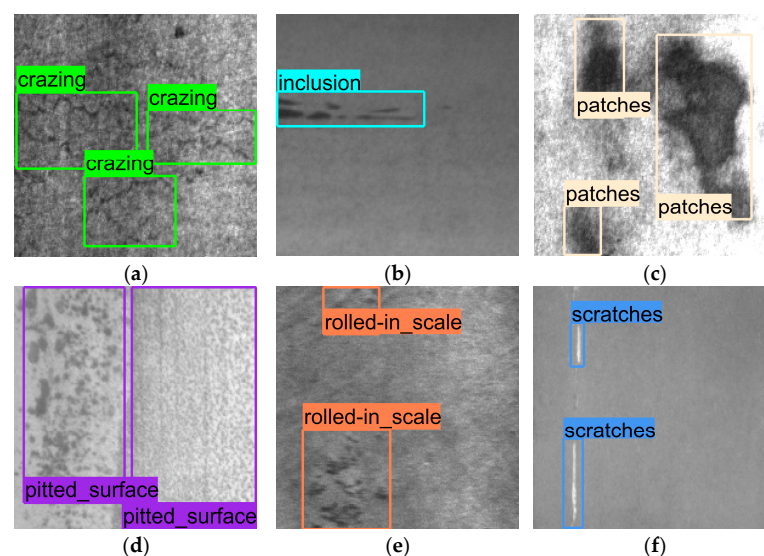


Figure 6. Examples of defective images with ground-truth annotations in the NEU-DET dataset: (a) Cr; (b) In; (c) Pa; (d) PS; (e) RS and (f) Sc.

4.2. Evaluation Metrics

Both classification accuracy and predicted location accuracy need to be evaluated in defect detection. Therefore, the $F1$ and average precision (AP) are used to evaluate the classification and location performance of model. In addition, mean average precision (mAP) and FPS are used to evaluate the comprehensive detection ability and detection speed of the model, where mAP is the average of all categories of AP . The above metrics are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (10)$$

$$AP = S_{P-R} \quad (11)$$

where TP , FP , and FN represent the number of true positives, false positives, and false negatives, respectively. S_{P-R} represents the area under the P - R curve.

4.3. Implementation Details

We perform all experiments on a workstation with one INTEL XEON CPU and four TESLA V100 GPUs (32 GB memory), CUDA 11.4, and CUDNN 8.0.5 on an ubuntu system. A single GPU is used for both training and testing. All codes are developed based on the deep learning framework pytorch 1.10.0. We uniformly scale the image size to 224×224 and use the stochastic gradient descent (SGD) optimizer to optimize the parameters during the training process. The weight decay is fixed to 0.0001 and the momentum is 0.9. The batch size for all experiments is eight. The initial learning rate is 0.001 and the maximum number of iterations is 17,000. The learning rate is reduced to 10% at 11,500 and 15,000 iter. The backbone is initialized with the pre-trained weight parameters in ImageNet to speed up the convergence of the model, and the rest is initialized with 'Xavier' [39]. In this study, the backbone of all models is ResNet50, unless we specifically mark. For the generation of anchors, we adopt the same settings as Faster R-CNN. The base anchor ratios of features in each level are (0.5, 1, 2), and the base anchor sizes are (32×32 , 64×64 , 128×128 , 256×256 , 512×512). The anchor boxes with different scales are used at different levels of feature maps for prediction.

The fc-head and conv-head play complementary roles to perform category classification and bounding box regression. We follow the setting in [36], classification tasks and regression tasks are assigned in fc-head and conv-head, respectively, and λ^{fc} is set to 0.7 and λ^{conv} is 0.8. In the total loss, ω^{fc} and ω^{conv} are set to 2.0 and 2.5, respectively, which means that it is more inclined to obtain accurate regression positions. In addition, the number of ROIs is reduced from 1000 to 500 to further improve the inference speed of the model.

4.4. The Result of Defect Detection

We perform extensive experiments on the NEU-DET dataset to evaluate the performance of DPN-Detector and compare it with RetinaNet [32], Cascade R-CNN [40], DETR [41] and YOLO-V5.

(1) The $F1$ is used to evaluate the classification performance of different detectors for defect categories. DPN-Detector has excellent classification ability due to the complementary fusion of the two detection heads. As shown in Table 2, DPN-Detector has the highest classification performance of 74.64%, significantly outperforming all one-stage detectors in the classification of all defect categories. Moreover, it shows improvement compared to the two-stage detectors. On the whole, DPN-Detector has better classification performance for steel strip defects. Furthermore, better classification performance contributes to better defect detection, which is demonstrated in the following experiments.

Table 2. The F1 comparison of different detectors.

Method	Cr	In	Pa	PS	RS	Sc	Overall
RetinaNet [32]	48.22%	73.33%	84.61%	73.46%	55.63%	71.13%	67.73%
YOLO-V5	49.70%	75.49%	84.44%	75.98%	61.20%	85.61%	72.07%
Faster R-CNN [29]	51.90%	72.55%	86.02%	74.34%	62.70%	83.97%	71.91%
Faster R-CNN+FPN [11]	50.95%	77.01%	84.79%	74.33%	63.08%	86.52%	72.78%
Cascade R-CNN [40]	51.94%	80.70%	85.04%	74.63%	60.85%	87.97%	73.52%
DPN-Detector	52.25%	80.85%	88.37%	76.51%	61.50%	88.34%	74.64%

(2) The mAP and FPS are used to evaluate the detection performance and detection speed of the detectors, respectively. As shown in Table 3, the DPN-Detector embedded with DPN module and double head outperforms other detectors. It achieves mAP of 80.93%, which is 3.52 points higher than the baseline Faster R-CNN. It shows that the multi-level feature detection method is beneficial to the detection of multi-scale defects. The mAP is also 4.11 points and 3.99 points higher than DETR and YOLO-V5, respectively. The gap of 4.48 points above RetinaNet is the most significant. At similar FPS, the mAP of DPN-Detector is 2.22 points higher than Cascade R-CNN, which also uses a heavy detection head. After embedding the DPN module in Faster R-CNN, the mAP is improved from 77.41% to 79.67%. The mAP is further improved to 80.93% after double head module is added. These results demonstrate the effectiveness of the DPN module and double head for improving the detection of steel strip defects. It is worth noting that the DPN module can bring greater performance improvement compared to double head.

Table 3. Defect detection results on NEU-DET.

Method	mAP	Cr	In	Pa	Ps	RS	Sc	FPS
RetinaNet [32]	76.45%	48.98%	80.41%	91.39%	84.87%	64.91%	88.15%	30.00
DETR [41]	76.82%	44.64%	86.19%	91.67%	80.90%	66.54%	90.99%	28.42
YOLO-V5	76.94%	45.80%	82.04%	91.93%	83.93%	65.67%	92.28%	54.00
Faster R-CNN [29]	77.41%	53.68%	79.34%	91.16%	79.42%	70.94%	89.89%	16.88
Faster R-CNN + FPN [11]	78.32%	53.38%	83.98%	90.91%	80.65%	70.11%	90.88%	23.48
Cascade R-CNN [40]	78.71%	54.28%	83.02%	90.93%	81.86%	69.58%	92.61%	19.29
Faster R-CNN + DPN	79.67%	53.47%	84.06%	92.06%	85.91%	71.56%	90.98%	20.77
DPN-Detector	80.93%	55.33%	86.45%	92.09%	85.03%	73.54%	93.16%	18.62

In addition, the performance of two-stage detectors is generally better than that of one-stage detectors. Additionally, high detection accuracy is extremely important for strip quality assurance, that is why we chose a two-stage detector as the baseline. The detection results of DPN-Detector for each defect type shows that DPN-Detector has better robustness than other detection methods with the same evaluation metric.

(3) The PR curves are used to evaluate the detection performance of DPN-Detector for each defect type. As shown in Figure 7, the area enclosed by each curve and the coordinate axis is the AP value of each defect. It can be seen obviously that the detection performance of Cr and RS is worse than that of other defects. The reason lies in the high similarity of the two types of defects to the background, and their complex and varied shapes. Furthermore, the two types of defects are very scattered. Other types of defects have a strong contrast to the background, and appear in the form of lumps, so the detection effect is relatively good.

(4) We use a single GPU to perform inference on the model with a total of 540 test images. The FPS of each detector is shown in Table 3. The FPS of DPN-Detector reaches 18.62, which can meet the needs of practical industrial applications. The one-stage object detector YOLO-V5 is famous for its fast detection speed with a maximum 54.00 FPS, but it does not perform well in the detection results of defects. Compared with Cascade R-CNN, the addition of DPN module shows good cost performance in terms of balancing the detection performance and speed of Faster R-CNN. Although DPN-Detector is not the fastest in de-

tection speed, it can still meet actual production needs. In terms of detection performance, it is superior to other detectors, which is very important for industrial applications.

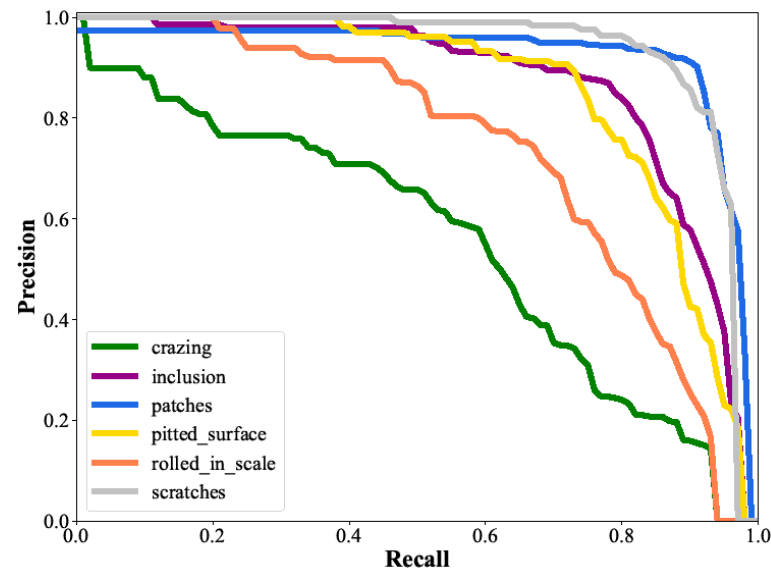


Figure 7. The PR curves for six defect types.

From the above evaluation results, the proposed DPN-Detector has better performance in the classification and detection tasks of the defect detection on the steel strip surface. The reason is that the proposed DPN module can enrich multi-scale information, expand the receptive field, and give good guidance to feature fusion. At the same time, double head takes full advantage of different inspection heads and high-lights defect features. It is very beneficial for the detection of surface defects of steel strips with various scales and strong background noise interference. Figure 8 shows some detection results sample of DPN-Detector on NEU-DET.

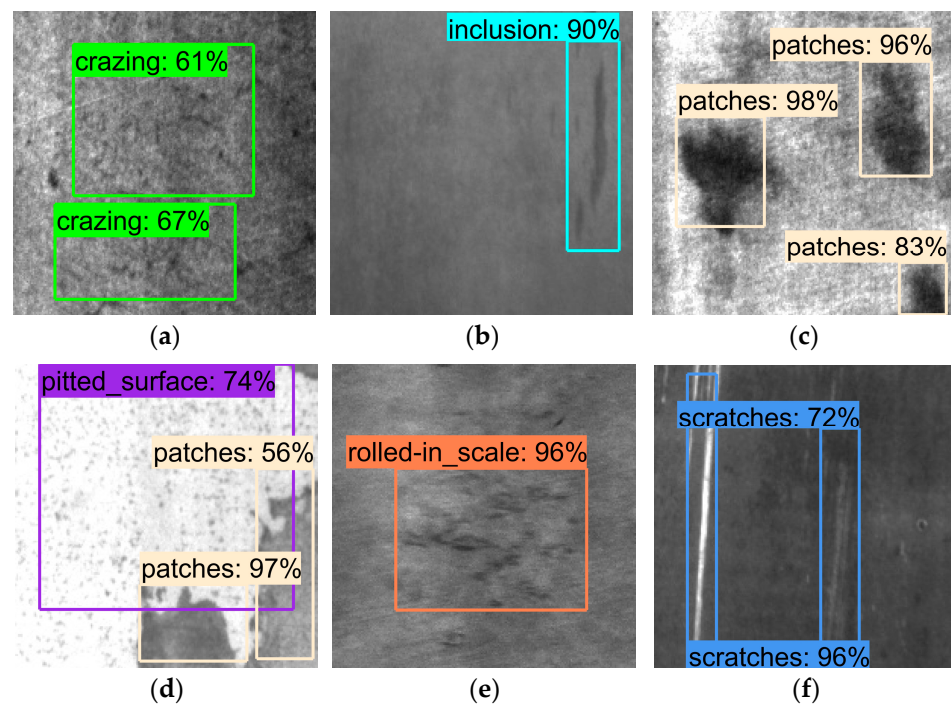


Figure 8. Example of DPN-Detector detection results on NEU-DET: (a) Cr; (b) In; (c) Pa; (d) PS; (e) RS and (f) Sc.

4.5. The Effect of Different Modules on the Detection Results

We perform ablation experiments on different components of the proposed DPN-Detector, RASPP, DPN, and double head, to verify the effectiveness of those components.

4.5.1. The Effect of RASPP

We conduct ablation experiments with the RASPP module as a variable. The results are shown in Table 4. Thanks to the multi-scale information captured by RASPP and the enlarged receptive field, the detection performance can be improved after embedding the RASPP module, whether it is a top-down feature pyramid or a top-down and bottom-up feature pyramid. It shows that the RASPP module is effective for multi-scale defect detection of steel strips. The mAP showed a larger drop after replacing the RASPP module with ASPP. The reason is that ASPP can capture a wider range of feature information than RASPP. The extra information does not always work but interferes with the useful feature information and causes model confusion, which leads to a decrease in detection performance. It also shows that RASPP is sufficient to cover defective feature information at all scales. The adaptive residual learning added in RASPP can independently judge whether the learned features are effective, so as to avoid the confusion caused by ASPP. It also helps to shorten the information flow path and enhance the robustness of the model. Nevertheless, RASPP requires fewer parameters than ASPP, so it has a faster detection speed.

Table 4. The results of DPN-Detector with different components on the NEU-DET dataset. (√: use this module, /: not use this module).

FPN	RASPP	Double Head	mAP	FPS
Top-down	/	/	78.32%	23.48
Top-down	ASPP	/	70.05%	20.00
Top-down	RASPP	/	78.84%	20.77
Top-down	RASPP	√	80.07%	18.62
Top-down + Bottom-up	/	/	78.77%	23.48
Top-down + Bottom-up	ASPP	/	71.47%	20.00
Top-down + Bottom-up	RASPP	/	79.67%	20.77
Top-down + Bottom-up	RASPP	√	80.93%	18.62

4.5.2. The Effect of DPN

We performed ablation experiments on the proposed DPN module by changing the structure of the feature pyramid. From the penultimate row in Table 4, the top-down and bottom-up feature pyramids with the RASPP module achieve an mAP of 79.67%, which is better than the feature pyramids with other structures. It proves the effectiveness of the DPN module for multi-scale defect detection and model robustness. When the ASPP modules are consistent, the detection performance of top-down and bottom-up feature pyramids is better than that of top-down feature pyramids, as shown in the comparison of the first four rows and the last four rows in Table 4. It proves that the bottom-up path can enhance the flow of information, outperforming the FPN module with only one top-down path. In addition, it can be seen from the FPS that the top-down and bottom-up feature pyramids do not affect the overall detection speed of the model compared to the top-down feature pyramid, which proves its excellent cost performance.

4.5.3. The Effect of Double Head

Ablation experiments are also performed to verify the effect of double head on model performance. From the last two rows of Table 4, the mAP of 80.93% is obtained after further applying double head. Similarly, the mAP is also improved by 1.23% after adding double head to the top-down FPN + RASPP. It proves the effectiveness of the double head using different detection heads to handle different tasks, and fully utilizes the advantages of detection heads with different functional focus. Although the detection speed is slightly re-

duced, it can still meet the needs of practical detection after replacing the original detection head with a double head.

4.5.4. The Impact of ROI Quantity

The number of ROIs affects the inference speed and detection performance of the model. A large number of regional proposals generated in RPN stage will be sorted according to their confidence levels, and only a few high-confidence region proposals will be used to extract ROI features. More ROI features will gain better detection results, but it will also slow down the inference speed of the model. As shown in Table 5, we performed ablation experiments with the number of ROIs as a variable. It can be seen that as the number of ROIs increases, the detection performance of the model shows an upward trend, while the detection speed gradually decreases. When the number of ROIs is increased from 500 to 1000, the detection speed will be damaged to a certain extent and the performance improvement will be limited. Additionally, when the number of ROIs is reduced to 50, the best detection performance is still 96.7%, which proves the robustness of our proposed model. In actual industrial production, the steel strip defect detection model needs to achieve a balance between detection performance and inference speed. Therefore, we set the number of ROIs to 500.

Table 5. Detection results of different ROI numbers.

ROI Number	mAP	Cr	In	Pa	Ps	RS	Sc	FPS
50	77.67%	47.57%	83.50%	90.35%	85.93%	68.03%	90.66%	28.42
100	78.22%	54.86%	83.30%	90.98%	82.82%	65.79%	91.59%	27.00
300	79.15%	55.51%	84.73%	91.89%	83.80%	66.20%	92.79%	24.55
500	80.93%	55.33%	86.45%	92.09%	85.03%	73.54%	93.16%	18.62
1000	80.97%	57.57%	84.48%	91.27%	85.36%	73.07%	94.08%	16.88

5. Conclusions

In this paper, a RASPP module is designed to expand the receptive field of the model so that the region proposals generated by the RPN network contain a broader region of feature information. Moreover, we combine RASPP with top-down and bottom-up pyramid networks to form DPN. DPN can enrich multi-scale feature information, shorten the information flow of the features among different levels, maintain more feature information, facilitate the detection of minor defects, and improve the robustness of the model to defect scales. In addition, we use a double head to perform the classification and regression tasks, respectively. We add a non-local attention mechanism in the double head to enhance defect features and suppress background features, as some defect types are highly similar to the background. Finally, with the proposed DPN-Detector, we obtain an mAP of 80.93% on the steel strip dataset NEU-DET, with 3.52% improvement compared to the baseline. The classification accuracy is 74.64%, and the detection speed reaches 18.62 FPS. The results show that our DPN-Detector has excellent advantages in detecting the defects on the steel strip surface. Our research will contribute to the automatic detection of surface defects in industrial production, promoting the production efficiency.

However, the proposed method still needs to be further improved. At present, the amount of computation of the method is relatively large and detection speed needs to be further modified. Furthermore, the highly reflected light in the industrial environment will affect the defect detection, which requires the features extracted by the detection method to be more robust. In the future, we will work on developing faster and more robust detection methods for steel strip defects.

Author Contributions: Methodology, X.Z. and M.W.; Supervision, X.Z. and J.L.; Writing—original draft, X.Z. and M.W.; Writing—review and editing, X.Z., M.W., Q.L., Y.F., Y.G., H.L. and J.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Guangdong Basic and Applied Basic Research Foundation, grant number 2020A1515010651.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are openly available at the following URL/DOI: http://faculty.neu.edu.cn/songkechen/zh_CN/zhym/263269. (Accessed on 30 December 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, S.; Zeng, Y.; Li, S.; Zhu, H.; Liu, X.; Zhang, X. Surface Defect Detection of Rolled Steel Based on Lightweight Model. *Appl. Sci.* **2022**, *12*, 8905. [CrossRef]
2. Luo, Q.; Sun, Y.; Li, P.; Simpson, O.; Tian, L.; He, Y. Generalized Completed Local Binary Patterns For Tme-Efficient Steel Surface Defect Classification. *IEEE Trans. Instrum. Meas.* **2018**, *68*, 667–679. [CrossRef]
3. Mordia, R.; Verma, A.K. Visual Techniques for Defects Detection in Steel Products: A Comparative Study. *Eng. Fail. Anal.* **2022**, *134*, 106047. [CrossRef]
4. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
5. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [CrossRef]
6. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
7. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
8. Li, Z.; Tian, X.; Liu, X.; Liu, Y.; Shi, X. A Two-Stage Industrial Defect Detection Framework Based on Improved-YOLOv5 and Optimized-Inception-Resnetv2 Models. *Appl. Sci.* **2022**, *12*, 834. [CrossRef]
9. Ling, Q.; Liu, X.; Zhang, Y.; Niu, K. Insulated Gate Bipolar Transistor Solder Layer Defect Detection Research Based on Improved YOLOv5. *Appl. Sci.* **2022**, *12*, 11469. [CrossRef]
10. Su, B.; Chen, H.; Zhou, Z. BAF-Detector: An Efficient CNN-Based Detector for Photovoltaic Cell Defect Detection. *IEEE Trans. Ind. Electron.* **2021**, *69*, 3161–3171. [CrossRef]
11. Lin, T.-Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 936–944.
12. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
13. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef]
14. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [CrossRef] [PubMed]
15. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 3–19.
16. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local Neural Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
17. Song, K.; Yan, Y. A Noise Robust Method Based on Completed Local Binary Patterns for Hot-Rolled Steel Strip Surface Defects. *Appl. Surf. Sci.* **2013**, *285*, 858–864. [CrossRef]
18. Chu, M.; Gong, R.; Gao, S.; Zhao, J. Steel Surface Defects Recognition Based on Multi-Type Statistical Features and Enhanced Twin Support Vector Machine. *Chemom. Intell. Lab. Syst.* **2017**, *171*, 140–150. [CrossRef]
19. Bai, X.; Fang, Y.; Lin, W.; Wang, L.; Ju, B.-F. Saliency-Based Defect Detection in Industrial Images by Using Phase Spectrum. *IEEE Trans. Ind. Inf.* **2014**, *10*, 2135–2145. [CrossRef]
20. He, Y.; Song, K.; Meng, Q.; Yan, Y. An End-to-End Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features. *IEEE Trans. Instrum. Meas.* **2019**, *69*, 1493–1504. [CrossRef]
21. Cheng, X.; Yu, J. RetinaNet with Difference Channel Attention and Adaptively Spatial Feature Fusion for Steel Surface Defect Detection. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 2503911. [CrossRef]
22. 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 Trans. Instrum. Meas.* **2021**, *70*, 5011111. [CrossRef]
23. Gao, Y.; Lin, J.; Xie, J.; Ning, Z. A Real-Time Defect Detection Method for Digital Signal Processing of Industrial Inspection Applications. *IEEE Trans. Ind. Inf.* **2020**, *17*, 3450–3459. [CrossRef]

24. Song, G.; Song, K.; Yan, Y. EDRNet: Encoder–Decoder Residual Network for Salient Object Detection of Strip Steel Surface Defects. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9709–9719. [[CrossRef](#)]
25. Liu, Z.; Tang, R.; Duan, G.; Tan, J. TruingDet: Towards high-quality visual automatic defect inspection for metal surface. *Opt. Lasers Eng.* **2021**, *138*, 106423. [[CrossRef](#)]
26. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
27. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective Search for Object Recognition. *Int. J. Comput. Vision* **2013**, *104*, 154–171. [[CrossRef](#)]
28. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
29. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
30. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
31. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
32. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [[CrossRef](#)]
33. Tang, M.; Li, Y.; Yao, W.; Hou, L.; Sun, Q.; Chen, J. A Strip Steel Surface Defect Detection Method Based on Attention Mechanism and Multi-Scale Maxpooling. *Meas. Sci. Technol.* **2021**, *32*, 115401. [[CrossRef](#)]
34. Su, B.; Chen, H.; Chen, P.; Bian, G.; Liu, K.; Liu, W. Deep Learning-Based Solar-Cell Manufacturing Defect Detection With Complementary Attention Network. *IEEE Trans. Ind. Inf.* **2020**, *17*, 4084–4095. [[CrossRef](#)]
35. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
36. Wu, Y.; Chen, Y.; Yuan, L.; Liu, Z.; Wang, L.; Li, H.; Fu, Y. Rethinking Classification and Localization for Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 10183–10192.
37. Glorot, X.; Bordes, A.; Bengio, Y. Deep Sparse Rectifier Neural Networks. In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS), Fort Lauderdale, FL, USA, 11–13 April 2011; pp. 315–323.
38. Chockler, H.; Farchi, E.; Godlin, B.; Novikov, S. Cross-Entropy Based Testing. In Proceedings of the Formal Methods in Computer Aided Design (FMCAD), Austin, TX, USA, 11–4 November 2007; pp. 101–108.
39. Glorot, X.; Bengio, Y. Understanding The Difficulty of Training Deep Feedforward Neural Networks. In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS), Sardinia, Italy, 13–15 May 2010; pp. 249–256.
40. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6154–6162.
41. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.