Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 2
CII 2023

A Method for Extracting Planar Image Features based on Convolution Neural Network

Shengjun Yang [1, a], Zhonglan Wu [1, b], Xiaohong Li [2, c] and Yun Hong [1, d, *]

[1] School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu, Sichuan, China

[2] Sichuan Dawn Precision Technology Co., Ltd, Meishan, Sichuan, China

[a] 2544187590@qq.com, [b] 190361503@qq.com, [c] 395190046@qq.com, [d, *] 577912131@qq.com

* Corresponding Author

Abstract. This paper is supported by the 2022YFG0070 science and technology plan of Sichuan Province. Image feature extraction technology plays a very important role in the measurement and control process of industrial product processing, but for planar micro defects in the background, the current general single shot multibox detector (SSD) has some disadvantages, such as easy loss of feature information, low detection accuracy, and insufficient number of detection feature maps. In view of the above problems, combined with the characteristics and requirements of image feature extraction in the process of measurement and control processing, this paper proposes and designs BSSD algorithm. The algorithm uses ResNet34 to extract more micro defect information to solve the problem of feature extraction; Seven multi-scale feature maps were selected to increase the number of feature maps for detecting micro defects; A backtracking layer is set up to fuse the abstract information of the high-level network into the shallow network before the multi-scale feature map is input into the classification network to enhance the expression ability of the abstract features. The experimental data show that the accuracy is comparable to DSSD, and the speed is similar to FSSD, which shows a significant advantage in the accuracy of small target detection.

Keywords: Process Measurement and Control; Minor Defects; Target Detection; SSD.

1. Introduction

This paper is supported by the Sichuan Science & Technology Program under Grant 2022YFG0070. In the past few decades, the rapid development of computing resources has promoted the rapid development of technologies such as deep learning and convolutional neural networks in various fields. These technologies rely heavily on computing and data resources. In addition, high-definition resolution cameras provide a more refined data basis for industrial product production lines. At present, advanced machine vision and other inspection systems have achieved good results in inspection accuracy and speed, and have gradually replaced manual operations.

Defect detection usually includes three stages: defect classification, defect location and defect segmentation. The corresponding networks are classification network, detection network and segmentation network. Common surface defect classification networks use convolutional neural network structures, such as MobileNet [1], GoogLeNet, ResNet and VGG, etc. The detection network is divided into two types: two-stage network (such as Faster R-CNN series) and one-stage network (such as SSD series). In the literature [2], SSD combined with YOLO is used to detect whether there are defects in the fasteners of the catenary bracket. This method can transform the defect detection task into a semantic or instance segmentation problem of defective areas and normal areas. Its advantage is that it can not only accurately segment the defect area at the pixel level, but also obtain the location, type and geometric parameters of the defect, such as length, width, height, area, boundary outline and center, etc. As for the segmentation network, it can generally be divided into two types: FCN (fully convolutional network) method [3] and Mask R-CNN method [4]. In the literature [5], a new cascaded autoencoder (CASAE) structure is proposed, using the idea of semantic segmentation to solve the problem of defect location and segmentation. Their method converts input defect images into pixel-level predictive masks based on semantic segmentation, thereby enabling the detection of metal defects and ensuring accuracy and robustness.

Currently, defect detection faces three main problems: small samples, small targets, and real-time performance. In real industrial scenarios, the number of defect pictures is often very limited, perhaps only a few or dozens. Therefore, the small sample problem is a very common challenge. To solve this problem, currently adopted methods include data amplification, synthesis and generation, network pre-training or transfer learning, and unsupervised and semi-supervised model methods. For example, in [6], defects are reconstructed by using Gaussian pyramid combined with semantic segmentation, and small-sample training is implemented in the inference stage through multi-scale detection results.

Another problem is real-time, especially in surface defect detection. The defect detection method based on convolutional neural network in industrial application involves three links: labeling, training and detection, and the real-time mainly focuses on the detection link. However, most methods pay more attention to the accuracy of classification and recognition, while ignoring the consideration of efficiency. The current solutions include introducing the feature pyramid structure, fusing the feature maps of different stages, multi-scale detection to improve the accuracy, and using roialign to replace rolpooling and gan2 to generate super-resolution features and superimpose them on the feature map of the original small target. For example, literature [7] uses Gan and automatic encoder to reconstruct the defect image, and uses local binary mode (LBP) to detect the local contrast defect of the image. It only needs to use the positive sample training algorithm, without the defect sample and manual labeling process.

In recent years, some new methods have been proposed to deal with the above problems. For example, Chunfang Deng et al. proposed the Extended Feature Pyramid Network (EFPN) in 2020, in which an ultra-high resolution pyramid layer for small target detection, namely feature texture migration (FTT), was designed to obtain super-resolution features and extract reliable regional details [8]. Literature [9] uses Mask R-CNN as the detection network and ResNet-101 as the shared feature extraction network. The designed sample annotation scheme has a small range of fluctuating dimensions and sets appropriate anchors size and aspect ratio, so that the Mask R-CNN network can achieve 98.2% defect detection rate. In addition, there are some casting defect 3D positioning methods based on deep learning feature matching [10], small target segmentation and small defect detection methods based on classification network Attention U-Net [11], and small data driven convolution neural network (SDD-CNN) [12].

Although the above methods have achieved good results to some extent, there are still some limitations. First, there are still difficulties in practical application, because there is still a gap between the accuracy rate and the actual demand. Therefore, how to give consideration to both accurate recognition and fuzzy features, real-time and accuracy is still the current difficulty. Secondly, due to the diversity of detection objects and types, the current algorithm ability to extract limited defect features from massive data is not enough to meet industrial needs. Therefore, we need to further study and develop more efficient and accurate defect detection methods to meet the needs of industry.

2. Selection of Target Detection Algorithm

In the development process of target detection algorithm, there are two branches, one is RCNN, fast RCNN and fast CNN, i.e. fastrcnn series [13-16], in which RCNN generates 1k~2k candidate boxes from a graph, extracts features from candidate regions and sends them to SVM classifier, and uses regression to refine the position. Finally, the map is 66% and FPS is 0.02 in voc2007. The fast RCNN network training time is reduced from 84h to 9.5h, and the test time is reduced from 47 seconds to 0.32 seconds. The fast RCNN can improve the target detection speed to 7fps, and the accuracy rate on the voc2007 data set is 73%. The second is the YOlO series [21, 23, 24] and SSD series [17-20] algorithms. The common feature of these algorithms is that there is no additional network to generate candidate regions, and classification and border regression are directly carried out on the feature map. YOLO regards the target detection algorithm as a regression problem, and uses global information for training and prediction. Therefore, compared with fast RCNN, YOLO reduces the background prediction error rate by half. SSD is the inheritance and improvement of YOLO algorithm, and it also uses the idea of dense candidate regions in fast RCNN for reference. SSD draws on the advantages of the two algorithms. Its accuracy is the same as that of fast RCNN, and its speed is the same as that of YOlO. The disadvantage of SSD is that the accuracy of small target recognition is low, and there is a certain gap compared with fast RCN. Therefore, SSD has great value in improving the accuracy of small target detection. On the VOC2007 test data set, the results of each detection algorithm are shown in Figure 1.

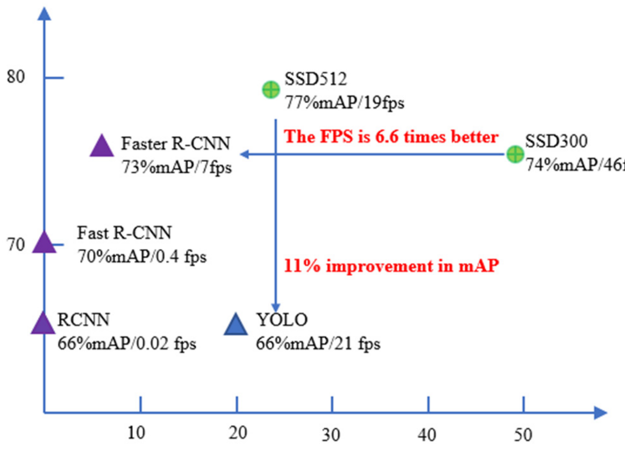


Fig 1. Performance comparison of target detection algorithms on VOC2007

It can be seen that the map of SSD512 and SSD300 are 77% and 74% respectively, which is about 10% higher than that of Yolo's map of 66%. The FPS of SSD512 and SSD300 are 19 and 46 respectively, which are 3-6 times higher than that of fast RCNN. Therefore, SSD is selected as the detection network in this paper.

3. BSSD Algorithm Design

3.1. SSD Network Structure

SSD (Single Shot MultiBox Detector) is a multi-scale target detection algorithm, which was proposed by Wei Liu at the ECCV2016 meeting. The algorithm can detect targets of different sizes by predicting on feature maps of different scales, and directly regression the category and location of targets. SSD also has end-to-end training capability and maintains good detection accuracy in low resolution images. In SSD, there are two network structures, SSD300 and SSD512, which are used to process images with different sizes. This paper focuses on the SSD300, as shown in Figure2 [22].

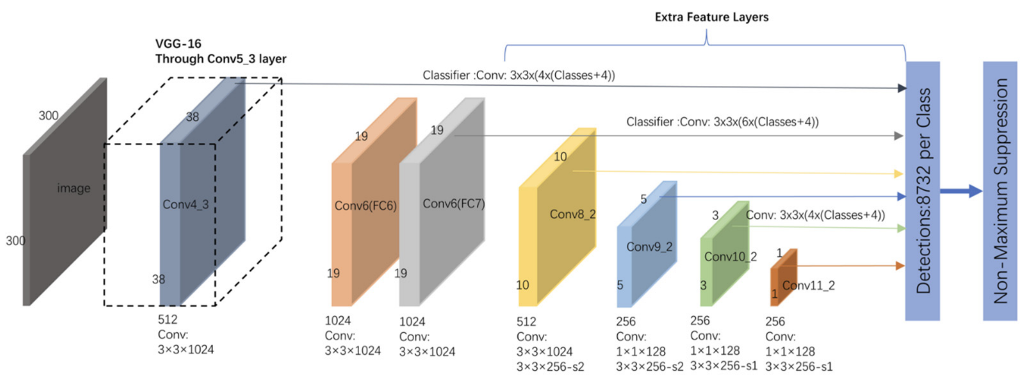


Fig 2. SSD300 network structure diagram

3.2. BSSD Overall Structure Design

The neural network structure designed in this paper is abbreviated as BSSD (backtracking single shot multibox detector). The BSSD network structure is shown in Figure 3.

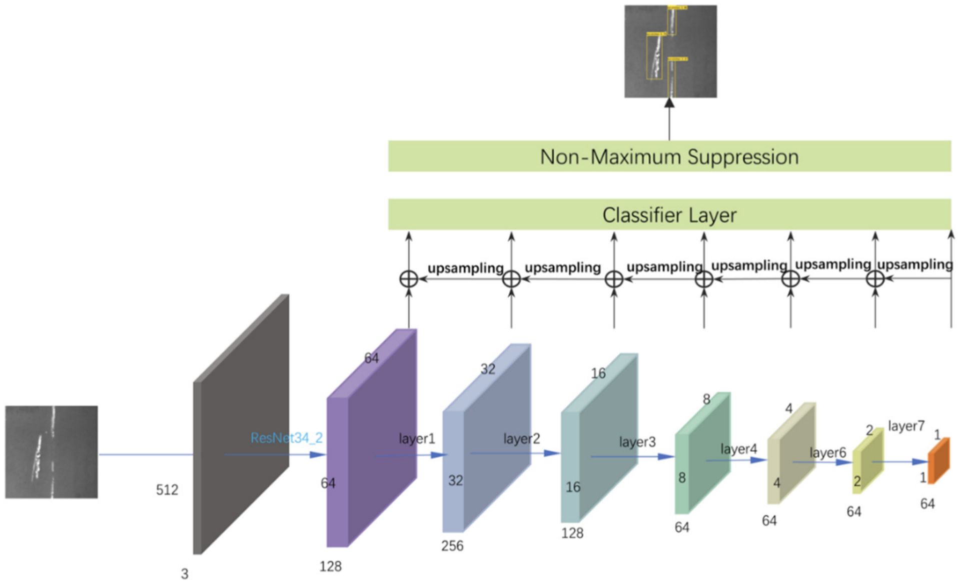


Fig 3. BSSD overall structure diagram

BSSD defect detection network structure is composed of five parts. Firstly, the first two layers of RESNET are used to extract the features of the preprocessed image. The second part is composed of a series of convolutional neural networks, which are used to further extract features and generate multiple feature maps with different scales. The third part is the backtracking layer, which solves the insensitivity of low-level features to abstract features by adding high-level features and low-level features layer by layer. The fourth part is the coding layer, which trains a coding vector combined with the candidate frame for the prediction frame in the training phase. The fifth part is the decoding layer, which uses the training encoding vector to decode when predicting, and selects the optimal prediction frame through NMS algorithm to generate the final result. Compared with the feature extraction network of SSD network, bssd uses RESNET to extract richer semantic information and introduces more shallow feature maps (such as 64, 32, 16, etc.) to increase the input of classification layer. In addition, bssd also uses backtracking layer to enhance the abstract information of shallow feature map. These improvements help to improve the accuracy and robustness of bssd in defect detection tasks.

3.3. BSSD Detailed Architecture Design

In the defect detection task, the data set size has an important impact on the performance of the algorithm. Because the data set used in this paper is relatively small, in order to better extract image features, we use ResNet34 model pre trained on Imagenet provided by pytoch for feature extraction. ResNet34 model consists of five layers in total, and each layer is composed of one or more convolution layers and pooling layers. Considering the task requirements of this article, we only use the first two layers of ResNet34 model for feature extraction, and use its output as the input of subsequent processing, as shown in Figure 4. Therefore, after the image input is preprocessed, we use resnet34 model to extract the features of the first two layers to obtain richer semantic information, so as to improve the accuracy and robustness of the defect detection algorithm.

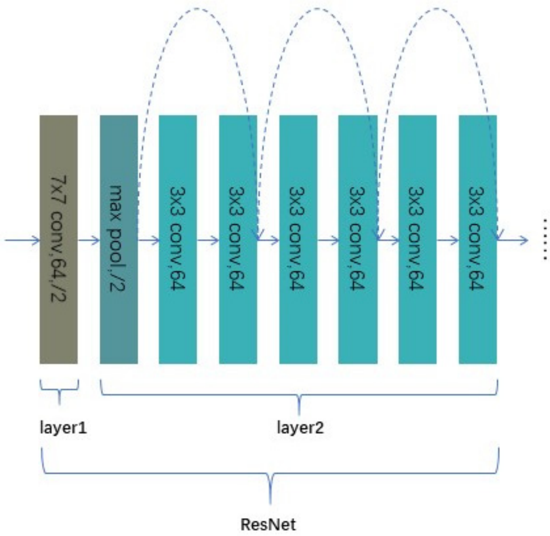


Fig 4. RESNET feature extraction layer structure diagram

As shown in Figure 5, in the convolutional neural network in the field of defect detection, if no additional padding is added, the size of the extracted feature map will gradually decrease and the receptive field will gradually increase. Receptive field refers to the size of the mapping area of pixels on the feature map output by a convolutional neural network on the original image.

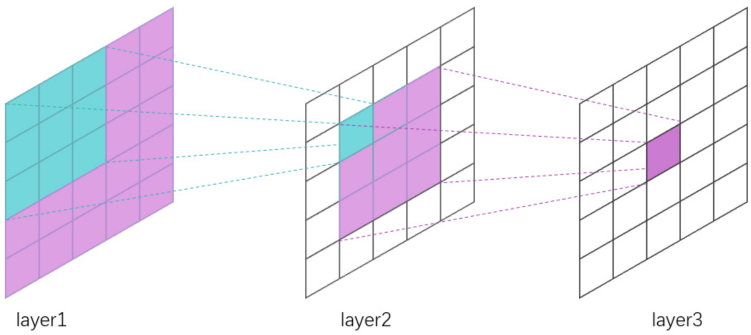


Fig 5. Schematic diagram of receptive field change

Combined with the output of resnet34 model, bssd (defect detection based on single scale feature map) algorithm designed seven multi-scale feature maps. In order to extract more rich feature information, these feature maps use small convolution kernel of 1×1 or 3×3 and relu activation function, and use steps of 1 or 2 for convolution operation. In addition, before the 3×3 convolution operation, the padding process is also carried out to maintain the consistency of the size of the feature map. Through this design, bssd algorithm can better capture the defect information under different scales, and improve the detection accuracy and robustness.

In order to effectively detect micro defects, the bssd model proposed in this paper adopts the selection strategy of shallow features. However, the abstract features extracted from the deep network are also of great significance. Therefore, compared with the general SSD network structure, bssd network adds an additional backtracking layer. After obtaining several feature maps of different sizes, the backtracking layer backtracks the deep features layer by layer and adds them to the shallow features to enhance the representation ability of the shallow features. Because the feature sizes extracted by convolutional neural network of each layer are inconsistent and cannot be added directly, it is necessary to carry out up sampling processing on deep features to make them consistent with the size of shallow features. At present, the commonly used up sampling methods include linear interpolation method, deconvolution method and anti pooling method, which can be used to realize up sampling operation. Through such design and processing, bssd model can make full use of shallow and deep features and achieve better performance in micro defect detection.

The feature conversion layer is only used to convert the number of channels, and there are two common processing methods in the feature conversion layer in defect detection to ensure that the size of the feature map is not affected. One processing method is to add padding to the feature vector and use a large convolutional neural network with convolution kernel size 3×3 for processing. As shown in Figure 6, this method can maintain the same size of the processed feature map and the input feature map, but may affect the characterization ability of the input feature map. Another processing method is to use convolution neural network with convolution kernel size 1×1 for processing.

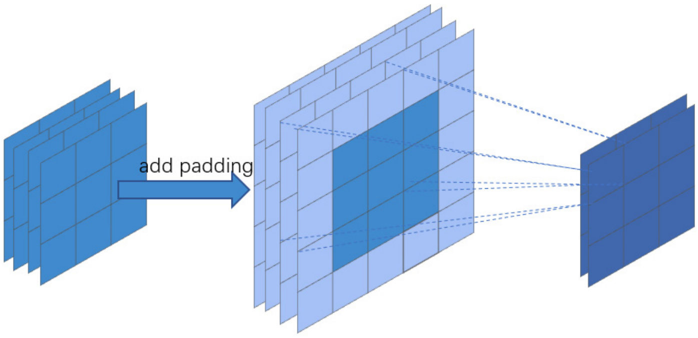


Fig 6. Schematic diagram of 3x3 feature conversion

As shown in Figure 7, this method can keep the image size unchanged without affecting the representation ability of the input feature map. Because the bssd model needs to better retain the characteristics of small defects, in addition to the first processing method, a convolutional neural network with convolution kernel size 1×1 is added for processing. Through this design, bssd model can extract and express features more effectively, and improve the detection accuracy of micro defects.

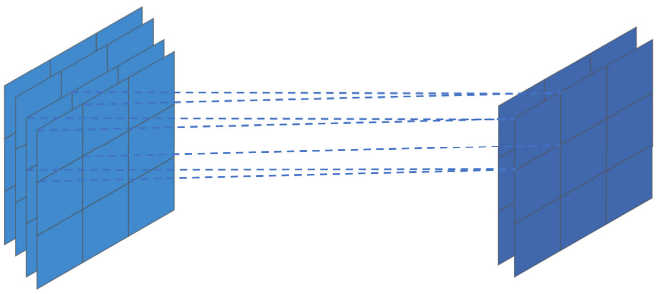


Fig 7. 1x1 feature conversion diagram

In the field of defect detection, each prior box needs to include three main parts: first, the location information of the prior box: use the abscissa and ordinate of the center point, as well as the width and height of the prior box to describe the location of a prior box. The second is the category information of the prior box: because the bssd model designed in this paper only detects one defect category, the category is set to 2 by default (the background is a separate category). The third is the position information of the prediction box: the position information refers to the offset relative to the prior box, which is used to determine the position relationship of the prediction box relative to the marker box. These three parts of information together constitute a priori box. In bssd model, the task of defect detection is realized by classifying and predicting the position of all prior frames. Therefore, it is very important to accurately describe and locate the position and category of each a priori box, which is also the key for bssd model to achieve good results in defect detection.

For location coding, bssd network adopts a similar way to SSD network. Let the position of the prior box be $p = (p^x, p^y, p^w, p^h)$, and the position of the marker box be $a = (a^x, a^y, a^w, a^h)$. In addition, there is an additional variable, variance, for adjusting the detection value. The position code is as shown in formula (1):

$$\begin{aligned} l_j^x &= \left(a_j^x - p_i^x\right) / p_j^w / \mathrm{var}\,iance[0] \\ l_j^y &= \left(a_j^y - p_i^y\right) / p_i^h / \mathrm{var}\,iance[1] \\ l_j^w &= \log\left(\frac{a_j^w}{p_i^w}\right) / \mathrm{var}\,iance[2] \\ l_j^h &= \log\left(\frac{a_j^h}{p_i^h}\right) / \mathrm{var}\,iance[3] \end{aligned} \tag{1}$$

After the positive and negative samples are divided, bssd network trains the selected samples and calculates the loss. The loss function is shown in (2):

$$L(x,c,l,g) = 1/N\left(L_{conf}(x,c) + \alpha L_{loc}(x,l,g)\right) \tag{2}$$

In the above formula, N is the number of positive samples in the prior box, c is the predicted value of category confidence, l is the predicted value of the position obtained above, and g is the position parameter of the marker box. The position loss function L_{loc} usually uses SouthL1loss, and the confidence loss function L_{conf} uses logsoftmax.

The decoding layer of BSSD model is mainly used in testing. The target detection is performed on the image to be detected and the detection frame is generated, which can be regarded as the reverse operation of the coding layer. The decoding layer first decodes the position information of the prediction frame from the prediction frame position encoding l. The decoding algorithm is shown in formula (3):

$$\begin{aligned} g_{predict}^x &= p^w * \left(l^x * \mathrm{var}\,iance[0]\right) + p^x \\ g_{predict}^y &= p^h * \left(l^y * \mathrm{var}\,iance[1]\right) + p^y \\ g_{predict}^w &= p^w \exp\left(l^w * \mathrm{var}\,iance[2]\right) \\ g_{predict}^h &= p^h \exp\left(l^h * \mathrm{var}\,iance[3]\right) \end{aligned} \tag{3}$$

After decoding, multiple prediction frames are usually obtained for a target. At this time, the NMS (Non-Maximum Suppression) algorithm needs to be used to remove the redundant prediction frame [25].

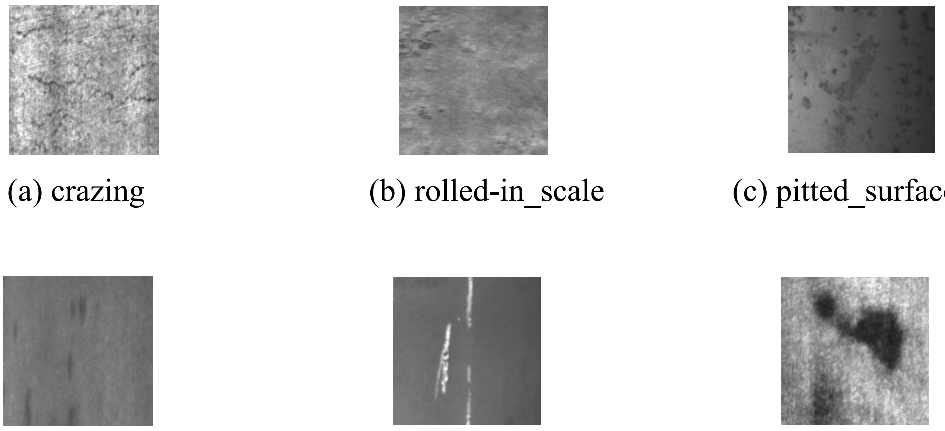
4. Experiment

In the process of production, the process is complex, and there will be a variety of defects. At present, the definition of all defects in the industry is mainly based on the factory production. We lack relevant production data. In order to evaluate the performance of the network on industrial micro defect detection, based on the original data set, for training and testing purposes, we borrowed a data set called SSDD (steel surface defect dataset). At the same time, in order to improve the performance of the network on small-scale data sets, this paper designs a data expansion method to expand the data sets, and designs experiments to verify the effectiveness of the model. In addition, this paper also compares the current mainstream micro feature extraction algorithms. The details are as follows.

4.1. Procedure

This paper is an experiment on Linux operating system using Python based deep learning framework pytorch. After all the experimental environments are configured, the network model can be trained.

Before detecting the two-dimensional surface defects, we first divide the surface defects of hot rolled steel strip into six categories, and the corresponding defect area increases in turn, as shown in Figure 8. In order to make the algorithm better learn the characteristics of defects, the defect image is preferentially used as the data set. The total number of original defect images collected is 1800, all defects are minor defects, and then the data set is expanded by using normal images according to the division of positive and negative samples. Finally, the labelimg tool is used to complete the annotation of the collected images, and a data set is made to train the network.



(a) crazing (b) rolled-in_scale (c) pitted_surface

(d) inclusion (e) scratches (f) patches

Fig 8. Schematic diagram of surface defect types of hot rolled steel strip

4.2. Experimental Validation

The enhanced data set has 12428 images after data preprocessing, and is divided into training set, verification set and test set according to the ratio of 8:1:1. On this basis, experiments are designed to verify the effectiveness of the model, as follows.

4.2.1. Effect of Different Upsampling Methods on the Performance of BSSD Model

SSD model has three optional upsampling methods in the backtracking layer, which are linear interpolation method, deconvolution method and anti pooling method. In order to verify the impact of different upsampling methods on the performance of the model, this experiment uses three different upsampling methods to set up bssd model, and conducts training and testing on ssdd data set, and compares the results. In this part of the experiment, the SSDD data set is used for the experiment after data enhancement. The pictures can be selected repeatedly after use. Two pictures are selected each time for model training, a total of 5000 times of training, and the training loss is recorded every 5 times of training. Through experiments, it can be concluded that the training loss of bssd model set by linear interpolation method and deconvolution method decreases faster, and the model training reaches a stable state faster. However, after the three methods are stable, the loss of bssd model set by deconvolution method can be minimized. In order to further explain the results, this section tests the models under the three settings respectively after training, and the test results are shown in the table1:

Table 1. Model effects using different upsampling methods

Upsampling Mode	mAP	FPS
deconv	84.3	18
unpooling	82.5	19
bilinear	82.3	19

The experiment shows that using deconvolution method for up sampling can achieve higher accuracy with longer image processing time. The difference in the image processing time is within an acceptable range, so deconvolution is selected as the default upsampling method of bssd model.

4.2.2. Influence of Different Methods of Feature Transformation on the Performance of Bssd Model

In the coding layer of BSSD model, convolutional neural network is used to convert the number of channels in the feature map. Different convolutional neural networks have different processing results, so this paper will use convolutional neural networks with convolutional kernel size of 1×1 and 3×3 for feature conversion, and set up two BSSD models for experiments. In order to verify the performance of the model on different data sets, experiments are carried out on the PSDD data set VOC2012 data set. The experimental results are shown in Table 2:

Table 2. Model effect under different feature transformation methods

Method	SSDD	VOC2012
Conv1×1	85.3	83.1
Conv3×3	84.7	83.2

It can be seen from the experiment that using two different methods for feature transformation has little effect on the performance of the model. In the small target detection task, the convolution neural network with the convolution kernel size of 1×1 can achieve better results. Therefore, in the task of surface defect detection of hot rolled steel strip, the convolutional neural network with a convolutional kernel size of 1×1 will be used by default for feature conversion.

4.2.3. Influence of Positive and Negative Sample Partition Ratio on the Performance of BSSD Model

In the training of the coding layer of the model, due to the large difference between the number of positive samples and negative samples, it is necessary to select features from the negative samples according to a certain proportion for the training of the model. The division ratio of positive and negative samples is 1:3 by default in SSD model. In order to obtain the best division results, BSSD model sets different proportions to select negative samples for training, and records the map and positive sample recall rate in each case for result analysis. The experimental results are shown in Figure 9.

Through the experiment, we can see that the recall rate of positive samples decreases with the increase of the proportion of negative samples. The map of model test reached a high level when the ratio of positive and negative samples was 1:3. And when the number of positive samples remains the same, with the increase of the number of negative samples, the total amount of data will also increase, thus increasing the training time. Therefore, BSSD model selects 1:3 as the default positive and negative sample division ratio.

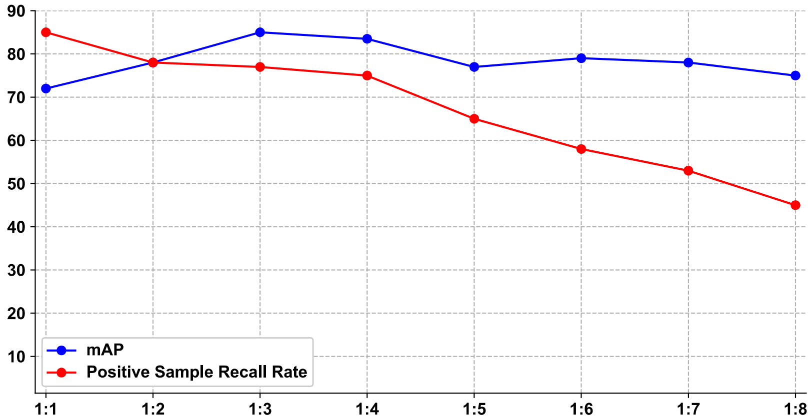


Fig 9. Model performance of different partition proportions of positive and negative samples

4.3. Comparative Analysis of Model Performance

In order to better measure the performance of the model, the experiment in this section will test the faster R-CNN model, YOLO model, SSD300, SSD512 and BSSD model designed in this paper on SSDD data set and VOC2012 data set respectively, and record the map and FPS for comparison. The experimental detection effect and results are shown in the figure 10 and table 3 respectively:

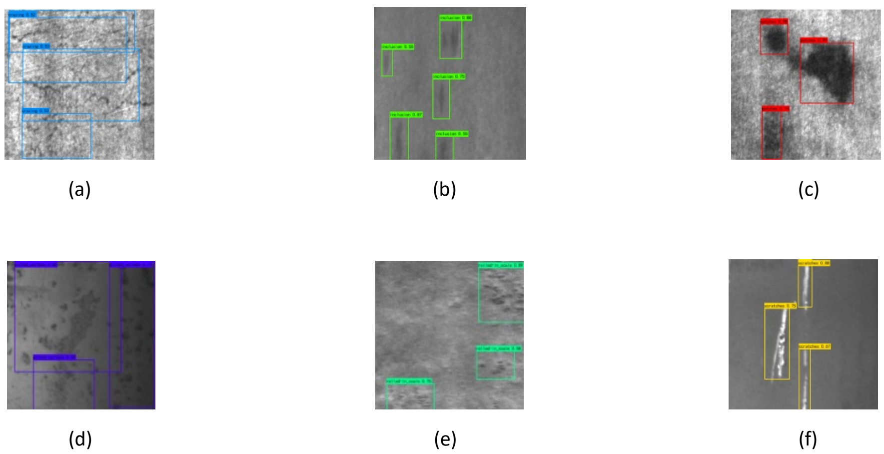


Fig 10. Pictures with different serial numbers represent the detection effect of BSSD network

Table 3. Detection effect of classical target detection algorithm

Model	SSDD		VOC2012	
	mAP	FPS	mAP	FPS
Faster R-CNN	76.3	12	73.2	7
YOLO	63.2	26	66.4	21
SSD300	70.5	55	74.3	46
SSD512	72.2	24	76.8	19
BSSD	82.3	18	76.5	15

The experimental results show that the detection results of BSSD model on VOC2012 dataset are slightly worse than that of SSD512 model, and the experimental results on psdd dataset are better than that of SSD512 model, and the detection speed is slightly lower than that of SSD512 model.

Table 4. Comparison of improved SSD algorithms

Model	SSDD		VOC2012	
	mAP	FPS	mAP	FPS
DSSD	80.2	12	78.4	8
RSSD	77.5	15	74.3	13
FSSD	80.2	21	76.8	17
BSSD	82.3	16	76.5	12

Among the improved SSD algorithm, DSSD has the highest detection accuracy and the slowest speed. The detection accuracy of FSSD is slightly lower than SSD, but the speed is ahead of FSSD. From the data in Table 4, it can be seen that the accuracy of BSSD algorithm in this paper is comparable to that of DSSD, and the speed is similar to that of FSSD.

5. Summary

In this paper, we propose BSSD algorithm based on SSD algorithm. The algorithm uses ResNet34 as the feature extraction layer, and uses 1×1 convolution to replace 3×3 convolution in the feature coding layer, so as to avoid the interference caused by padding in order to enhance the expression ability of micro defect features, the first two layers with large scale in the feature map are input into the classification network as multi-scale features, and the backtracking layer is set to integrate the abstract features of the high-level network, To enhance the ability of feature abstraction of shallow network. Through strict experiments, the effectiveness of the method is verified. In addition, the network layout has been implemented in industrial projects, which shows that its performance and speed meet the corresponding indicators.

Acknowledgments

Thank you very much for the technical support and valuable suggestions provided by Professor Hongtao for this study. We would also like to thank Zhonglan Wu for her theoretical and methodological support. We would also like to thank Sichuan Provincial Science and technology plan 2022YFG0070 for its support.

References

[1] Howard A, Menglong ZhuBo Chen, D. Kalenichenko, W. Wang, Tobias Weyand, M. Andreetto and H. Adam. Efficient Convolutional Neural Networks for Mobile Vision Applications[J]. ArXiv abs/ 1704. 04861, 2017: n. pag.

[2] Deng C, Wang M, Liu L, et al. Extended Feature Pyramid Network for Small Object Detection[J]. 2020.

[3] Shelhamer, Evan, J. Long and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017: 640-651.

[4] Ren, Shaoqing, Kaiming He, Ross B. Girshick and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence 2015, (39): 1137-1149.

[5] Tao, Xian & Zhang, Dapeng & Ma, Wenzhi & Liu, Xilong & Xu, De. Automatic Metallic Surface Defect Detection and Recognition with Convolutional Neural Networks[J]. Applied Sciences. 2018, 8(9):1575.

[6] Mei S, Wang Y, Wen G. Automatic Fabric Defect Detection with a Multi-Scale Convolutional Denoising Autoencoder Network Model. Sensors (Basel), 2018, 18(4):1064.

[7] Zhao Z, Bo L, Rong D, et al. A Surface Defect Detection Method Based on Positive Samples[C]// Pacific Rim International Conference on Artificial Intelligence. Springer, Cham, 2018.

[8] Deng C, Wang ML iu L, et al. Extended Feature Pyramid Network for Small Object Detection[J]. 2020.

[9] Han M, Qingxiang Wu, Xiongjun Zeng. Gear visual micro defect detection based on deep learning [J]. computer system applications,2020,29(03).

[10] Yongxiong Yu, Liuqing Du, Cuilan Zeng, et al. Automatic location method of casting micro defects based on deep learning feature matching [J]. Journal of instrumentation, 2016, 37(06).

[11] Bingyin Ren, Zhiyong Li, Dai Yong. Detection method of slight scratch on mobile phone screen [j]. Journal of Harbin Institute of technology, 2021, 53 (01).

[12] Xu X, Zheng H, Guo Z, et al. SDD-CNN: Small Data-Driven Convolution Neural Networks for Subtle Roller Defect Inspection[J]. Applied Sciences, 2019, 9(7).

[13] Girshick, Ross B. Fast R-CNN[J]. IEEE International Conference on Computer Vision (ICCV),2015: 1440-1448.

[14] Ren S, He K, Girshick R B, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. neural information processing systems, 2015, 22(5): 91~99.

[15] Girshick R. B, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA: IEEE Computer Society, 2014, 580-587.

[16] He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. European Conference on Computer Vision. Zurich, Switzerland: ECCV, 2014. 346-361.

[17] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[C]. European Conference on Computer Vision. Zurich, Switzerland: ECCV, 2016, 21-37.

[18] Fu C, Liu W, Ranga A, et al. DSSD: Deconvolutional Single Shot Detector. Neural Networks, 2017, 17(1):455-468.

[19] Li Z, Zhou F. FSSD: Feature Fusion Single Shot Multibox Detector. Neural Networks[J], 2017,2(9):52~69.

[20] Liu S, Huang D, Wang Y, et al. Receptive Field Block Net for Accurate and Fast Object Detection[C]. European Conference on Computer Vision. Zurich, Switzerland: ECCV, 2018, 404-419.

[21] Redmon J, Divvala S. K, Girshick R. B, et al. You Only Look Once: Unified, RealTime Object Detection[J]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016, 779-788.

[22] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[C]. European Conference on Computer Vision. Zurich, Switzerland: ECCV, 2016, 21-37.

[23] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA: IEEE Computer Society, 2017, 6517-6525.

[24] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement. Artificial Intelligence and Statistics[J], 2018, 88(252):687-690.

[25] Liu Wei, Zhu Hong and Han Gui-Jin, "NMS-based blurred image sub-pixel registration," 2011 International Conference on Image Analysis and Signal Processing, Hubei, 2011, pp. 98-101, doi: 10.1109/IASP.2011.6109006.