Mattress Defect Detection and Classification Based on Faster R-CNN Algorithm

Zhuang Zichen(23020221154155 AI class), Fang Baohui(23020221154084 Informatics class), Deng Jinglong(23020221154080 AI class), Liu Zhanpeng(36920221153104 AI class), Luo Hao(36920221153106 AI class)

Abstract

As a common household goods, quality of the mattress is of great concern. Production will inevitably cause defects in the mattress, such as stain, threads, holes. Mattress needs to be inspected before they leave the factory. According to this requirement, this paper designs a mattress defect detection and classification method based on Faster R-CNN algorithm (Faster Region Convolutional Neural Networks) . The task of this project consists of two parts: defect classification and defect localization. The Faster R-CNN deep learning target detection network is implemented using the Pytorch deep learning framework in Python, and the defect detection task of the input mattress image is completed through comparison experiments and parameter optimization. 300 images of mattresses will be used as the dataset. VGG19 network and a variety of different CNN backbones will be used for the comparison experiment. K-means algorithm will be used to cluster the defects in the dataset. Due to the difficulty in detecting thread defects, Faster R-CNN network uses Reduced Focal Loss as the classification loss function. Finally realizing the detection and classification of mattress defects.

Introduction

The assembly line of the textile industry has high requirements for product quality. In the production process, products often have defects (stains, threads, holes, etc.) due to equipment operation errors, and once defective products flow into the market, they will damage consumer rights and interests, affect corporate image, and cause economic losses. These defects directly affect the overall product quality and require defect detection. In the past, product defect detection was mainly carried out manually. However, the speed of human eye detection lags behind the production speed of the assembly line, so the company not only has to invest a lot of manual testing of mattresses, but also the detection stage may become an efficiency bottleneck in assembly line production. In addition, the human eye is prone to fatigue after a long period of labor, which will lead to missed detection and reduce the accuracy of defect detection. In recent years, deep neural network technology has enabled computer systems to accurately detect damage to textile surfaces. Taking the Faster R-CNN algorithm as an example, it can adaptively identify different manifestations of the same defect, and can classify and locate multiple defects. Therefore, the Faster R-CNN algorithm has more and more applications in the field of defect detection(Ren et al. 2015).

In summary, based on the following three reasons, this paper proposes the topic of "mattress defect detection and classification based on the Faster R-CNN algorithm":

- Mattress fabric defects will seriously affect the quality and sales of products.
- Manual detection is prone to missed detection under long-term high-intensity labor.
- Faster R-CNN algorithm has strong adaptive ability and high accuracy in the field of defect detection.

In this paper, the mattress defect detection and classification based on the Faster R-CNN algorithm can replace the manual quality inspection of mattresses to a certain extent. This can not only improve the quality of the company's mattress products, but also reduce the missed detection rate and reduce labor costs.

In terms of the current state of research, the idea of using computer vision technology on industrial assembly lines has been around for a long time. Earlier, defect detection was achieved by extracting defect features and texture arrangement rules, but once the texture of the defect changed, it could not be accurately identified, even different shapes of the same defect. At present, some methods commonly used for fabric defect detection are structure-based and statistical-based, spectrum-based, and deep learning-based. Among them, the disadvantage of structure-based method is insensitive to changes in texture structure; based on statistics is characterized by simple operation but insensitivity to small defects; based on spectrum is characterized by directional selectivity, which is accompanied by high computational cost. Method based on deep learning has the advantages of strong adaptability, high accuracy, accurate detection and classification.

Due to these insurmountable barriers to structure-based, statistical-based, and spectrum-based defect detection, deep learning is used for defect detection. The main object detection methods based on deep learning include SSD, YOLO and Faster R-CNN. Among them, YOLO and SSD algorithms belong to the first-stage object detection algorithms, which have the characteristics of fast detection speed, but the detection accuracy is not as good as the two-stage object detection algorithm (such as Faster R-CNN). Therefore, the Faster R-CNN algorithm was chosen for this project(Jiang

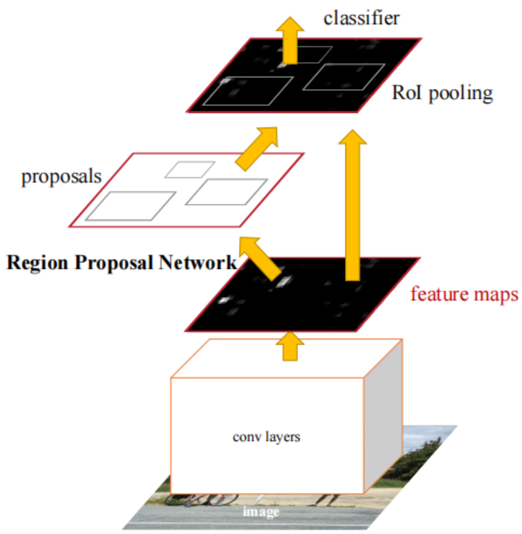

Figure 1: Faster R-CNN network structure

et al. 2022).

In this paper, the feature map of the input image is extracted through the backbone network and the proposals extracted by the RPN network are used as the input of the Faster R-CNN network. Through the classification and regression of the Faster R-CNN network, the defects are accurately classified and located, and the network can accurately classify and locate surface defects by improving the network structure, adjusting network parameters, optimizing anchor box design and adjusting the learning rate. The key issues that this article intends to address are as follows:

- The original dataset is not enough to support deep learning training.
- Selection of backbone networks.
- Design of the Anchor box.
- Design of the loss function.

Related Work

Fabric defect detection is a essential step of quality control in the textile manufacturing industry. Traditional fabric inspections are usually performed by manual visual methods, which are low in efficiency and poor in precision for long-term industrial applications. At present, defect detection for fabrics mainly includes computer vision algorithms based on structure, statistics, spectrum and deep learning(Rasheed et al. 2020).

Structure-based computer vision algorithms can be used to extract image features(Ahmed, Gao, and Woo 2018). proposed to conduct the low rank and sparse decomposition jointly and extract weaker defects feature based on wavelet integrated alternating dictionary matrix transformation(Gao et al. 2018). Utilized an unsupervised sparse component extraction algorithm to detect micro defects in a thermography imaging system by building an internal sub-sparse grouping mechanism and adaptive fine-tuning strategy. However, this method is not sensitive to changes in texture structure and is greatly affected by illumination. Statistical computer vision methods can be used to discover information such as image color, variance, etc., which can be used for image detection tasks(Wang et al. 2018). Established a successive optical flow for projecting the thermal diffusion and constructed principal component analysis to further mine the spatial-transient patterns for strengthening the detectability and sensitivity(Hamdi et al. 2018); utilized non-extensive standard deviation filtering and K-means to cluster fabric defect block; Although its operation is simple, it has the problem of insensitivity to small flaws.

Structure-based, statistics-based and spectrum-based defect detection all have the above insurmountable obstacles, while the deep learning target detection method has strong adaptive ability and has received extensive attention in recent years(Mei, Wang, and Wen 2018). Reconstruct fabric defect image patches with a convolutional denoising auto-encoder network at multiple Gaussian pyramid levels and to synthesize the detection results from the corresponding resolution channels(Li et al. 2018). Introduced DetNet, which is specifically designed to keep high resolution feature maps for prediction with dilated convolutions to increase receptive fields(Ren et al. 2015). Proposed the Region Proposal Network (RPN) to generate proposals in a supervised way based on sliding convolution filters. Although generic models are simple and easy to deploy, they are either over-fitting or feature extracting insufficiency.

Among the large number of approaches proposed, region-based CNNs (RCNN) have received significant attention due to their effectiveness. However, extracting candidate regions is time-consuming. Consequently, it does not have the ability of practical application. On this basis, the Fast R-CNN algorithm was derived in 2015, which reduced the computational requirements of the algorithm through RoI Pooling and accelerated the detection speed of R-CNN. Then in 2017, the Faster R-CNN (Ren et al. 2015) algorithm combined with the feature pyramid FPN network appeared. It not only used the RPN network to generate candidate regions, but also performed multi-scale feature fusion through the feature pyramid, which achieved the effect of accurate detection of targets at different scales. In 2021, CADN (Zhang et al. 2021) proposed a novel class-aware Conv-Pooling module. This module can explore weak image label annotations to extract spatial information.As verified, weakly supervised defect detection is achieved and competitive results are obtained by using the proposed CADN method. In this paper, we choose Faster R-CNN as our base detector, and improve its generalization ability for object detection in a new target domain.

Method

Shared Convolutional Layer Backbone Network

The shared convolutional layer backbone network is used for feature extraction of raw images. The quality of the extracted feature map is important in the final experimental results. Commonly used CNN networks include AlexNet, VGG, ResNet and so on. The different network layers and network structures of these networks lead to different feature extraction capabilities and different computing requirements. In particular, in the Faster R-CNN feature extraction stage, there is no need to classify the image, so only the part before the fully connected layer of the CNN network is used. Different CNN backbone networks have different feature extraction capabilities and require different computing powers. Therefore,it is important to choose an appropriate CNN backbone network.

The backbone network selected for this topic is VGG19. The VGG19 backbone network achieved the highest average accuracy in this project. Compared with other networks, the number of network layers is not deep, the number of parameters is moderate, and it has good detection efficiency. Comparative experiments and result analysis of the backbone network will be presented in the experimental section.

Region Proposal Network

Region Proposal Network (RPN) is used to extract proposals from feature images.In the Region Proposal Network, we use the output of the backbone network as the input of the feature map of the RPN.We first pass the input through two 3*3 convolution kernels, and the output results are used for classification and regression. Then, for each pixel after output, several anchor frames are generated according to the set anchor parameters (aspect ratio, scale information).The softmax classifier scores the anchor box to determine whether it is a positive or negative sample.We use the softmax classifier to score the anchor box to determine whether it is a positive or negative sample.Then, the positive and negative samples are retained according to the set ratio, and the position information of the prediction frame is fine-tuned and regressed. Finally, the proposal information is generated at the proposal layer.The specific structure is shown in the figure below.

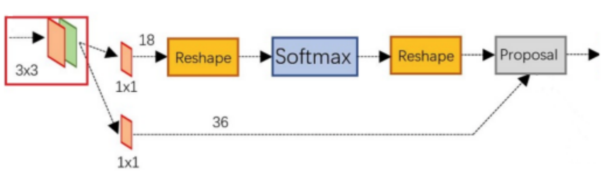

Figure 2: Region proposal network structure

Anchors Design Each pixel of the RPN feature map is generated with multiple anchors. They have different scales. Then we will filter the most suitable anchor parameters from

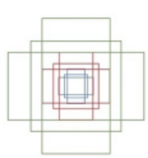

Figure 3: Anchors selection strategy

different anchors as the target anchor. VGG16 is chosen as the backbone network. The extracted feature map will be down-sampled by 16 times, [1:2, 1:1, 2:1] is selected as the aspect ratio, and [8, 16, 32] is used as the scale standard.We will calculate the scale range of the anchor mapping to the original image according to the following formula.

$$\text{Area} = (\text{Scale} \times \text{Featstride})^2 \tag{1}$$

Bring in 3 different aspect ratios to get the anchor area for each pixel.The anchor parameter also depends on the scale size and aspect ratio of the target. We found that when the object of target detection is a target with a large aspect ratio such as cracks and street lights, it is usually necessary to increase the aspect ratio of the anchors. As shown in the figure below. We use a clustering algorithm in order to select

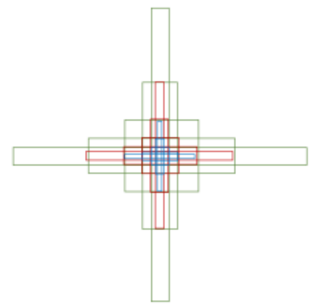

Figure 4: Schematic diagram of anchors with a large aspect ratio span

the appropriate anchor parameters. By clustering the scale and aspect ratio of the dataset targets, select the appropriate anchor parameters for this dataset.

Loss Function Design For training RPNs, we assign a binary class label (of being an object or not) to each anchor. We consider an anchor that has an IoU overlap higher than 0.7 with any ground-truth box as positive labels, and the rest as negative labels. In addition, the RPN network contains two partial tasks of classification and regression. Therefore, classification and regression will be described in sections in this paper.

Classification Loss Function Design In this paper, the classification loss function adopts the classic cross-entropy loss function. The Cross-Entropy Loss is shown in the following formula.

$$\text{Cross Entropy } (p, y) = \begin{cases} -\log(p), \text{if } y = 1 \\ -\log(1-p), \text{if } y = 0 \end{cases} \tag{2}$$

It is often used in classification problems to describe the loss between discrete predicted classifications and true classifications. Because the RPN network can artificially control the number of positive and negative samples, there will be no imbalance between positive and negative samples. Therefore, this paper uses the cross-entropy loss function as the classification loss function.

Regression Loss Function Design Due to the L2 loss function, when the error between the network prediction value and the real value is large, there is a problem of gradient explosion. In addition, the L2 loss function is more sensitive to individual outliers in the network. The Smooth L1 can solve the above problems very well. Therefore, this paper uses the Smooth L1 loss function as the regression loss function.

Overall Loss Function Design

In this Faster R-CNN network, it includes the background classification of RPN and the specific defect classification of Fast R-CNN network. Among them, the RPN network only classifies the foreground and foreground information, and the number of foreground and foreground categories can be limited by setting the number of positive and negative sample anchors. Therefore, the classification loss function of RPN can be used. However, in the specific defect classification of Fast R-CNN, there is no guarantee that the number of samples of defects is balanced and the degree of difficulty is similar. Therefore, it is necessary to add Focal Loss loss function optimization to the Fast R-CNN classification network. The mathematical expression of the Focal Loss loss function is shown below.

$$\text{Focal Loss}\left(p_t\right) = -\left(1-p_t\right)^{\Upsilon} \log \left(p_t\right) \qquad (3)$$

Among them, Υ is the focusing factor of Focal Loss.when Υ is 0, Focal Loss becomes a cross-entropy loss function. However, we found that object detection requires learning not only for defective objects but also for background classes. Therefore, on the basis of the Focal Loss loss function, we adjust the weight of the loss function of different categories of targets by adding the parameters of α_i. The modified Focal Loss is shown in the following formula.

$$\text{Focal Loss}\left(p_t\right) = -\alpha_{\mathrm{i}}\left(1-p_t\right)^{\Upsilon} \log \left(p_t\right) \qquad (4)$$

For samples that are difficult to classify, it can be considered to increase its weight in the loss function through the Υ index. This makes the neural network pay more attention to the classification of difficult samples during training, which greatly improves the accuracy of target detection. But artificially reducing the weight of easy-to-classify samples will lead to a decrease in recall rate. There is a slight negative impact on the object detection task. In order to continue the characteristics of Focal Loss inclined to difficult samples and improve the problem that Focal Loss reduces the recall rate, we finally adopted Reduced Focal Loss as our loss function. The Reduced Focal Loss is shown in the following formula.

$$\text{ReducedFocalLoss}\left(p_t\right) = -fr(pt, th) \log \left(p_t\right) \qquad (5)$$

$$fr(x) = \begin{cases} 1, \text{ if } p_t < th \\ \frac{\left(1-p_t\right)^{\Upsilon}}{th^{\Upsilon}}, \text{ if } p_t \geq th \end{cases} \qquad (6)$$

Training Faster R-CNN

RPN networks can be trained end-to-end by backpropagation and stochastic gradient descent. Each image produces multiple positive and negative sample anchors, with negative samples accounting for the majority. If the loss function of all anchors in each batch is directly optimized, it will cause the loss to be biased towards negative samples. We refer to the processing method of the original author: randomly sample 256 anchors in the image, calculate the loss function of a small batch of data, and the ratio of positive and negative anchors sampled can reach 1:1. If there are fewer than 128 positive samples in the image, the minibatch data is populated with negative samples. The shared convolutional layer is initialized by a pre-trained ImageNet classification model. The other network layer parameters are randomly initialized using weights obtained in a zero-mean Gaussian distribution with a standard variance of 0.01. Our implementation uses pytorch.

Next, we utilized the candidate box obtained by RPN for object detection. The detection network uses Fast R-CNN. Independently trained RPN and Fast R-CNN will modify the convolutional layers in different ways. Instead of learning two separate networks, we need to make convolutional layers shared between two networks. To this end, the original author proposes three methods for training a unified network with a shared convolutional layer composed of RPN and Fast R-CNN, including alternating training, approximate joint training and non-approximate joint training. The training method we chose is approximate joint training. In this solution, the RPN and Fast R-CNN networks are combined into one network during training. In each SGD iteration, forward delivery generates a region proposal, which is treated as a fixed, precomputed proposal when training the Fast R-CNN detector. Backpropagation proceeds as usual, wherein for the shared layer, the backpropagation signal from RPN loss and Fast R-CNN loss is combined. This solution is easy to implement, but ignores the derivative of the coordinates of the bounding box with respect to proposals.

Implementation Details

Anchor was used on 3 scales, with bounding box areas of 8, 16 and 32 pixels, respectively. Its aspect ratios are 1:1, 1:2 and 2:1. These hyperparameters were carefully selected for this dataset, and we'll provide details about the experiment in the next section. The setting of anchor requires the use of clustering functions to explore the data distribution characteristics. We use the K-means clustering algorithm to perform cluster analysis on the scale of defects, and select the scale scales parameter and aspect ratio ratio parameters according to the cluster center.

When we crop to the image boundary, a proposal bounding box that crosses the boundary may be produced. Anchor boxes that cross image boundaries need to be handled carefully. Our approach is ignoring all cross-border anchors during training, so they do not contribute to the loss. If cross-boundary outliers are not ignored in training, the objective function produces large error terms and training is difficult to converge.

Experiment

Datasets and Experimental Settings

We collected 200 pictures with a resolution of 600*600, and used labelImg to manually label the defect targets of the images. The ground truth data format is an xml file. As a large number of mattress defects could not be captured in a short time, there were fewer raw mattress defect images available.

Therefore, image enhancement techniques were used to expand the dataset. Original training dataset was 200 images, and 2200 valid images were obtained after data augmentation. This experiment mainly includes two parts: ablation experiment and final detection effect experiment. We conduct ablation experiment on target scale, target frame ratio, NMS threshold, loss function parameters and backbone selection respectively, use 2200 600*600 images after data enhancement as training data and 100 images as test data to visualize the final detection results, and the resources used in this experiment are RTX 3090 GPU.

Anchor Parameter Setting

Get Scales Parameter Using the K-means clustering algorithm, the clustering center hyperparameters were set to three and the scale of the defects were clustered and analyzed to obtain three clustering centers, which are [5505.12, 12794.40, 24033.60], taking into account the characteristics of the VGG network, the graph is down-sampled by a factor of 16, therefore, the scale parameter [4, 6, 8] is chosen. The results are showed in Fig5.

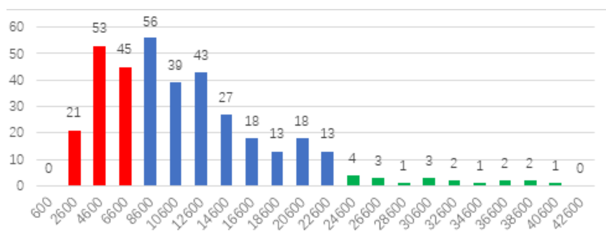

Figure 5: Defect target scale clustering results

Get Aspect Ratio Parameter The aspect ratio ratios parameter is different from the scale scales parameter, which is a two-dimensional clustering. Firstly, the aspect ratio information of the three types of defects is aggregated and cluster analysis is performed. The clustering results are showed in Fig6. Finally, ratios of [2:1, 1:1, 1:2] were finally selected.

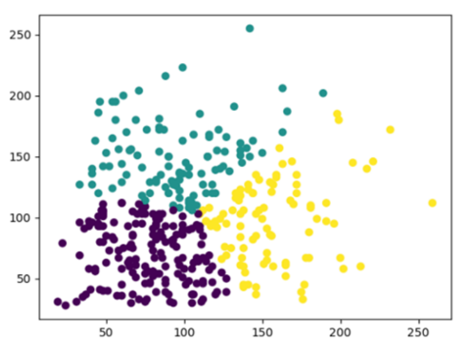

Figure 6: Defect aspect ratio clustering results

Non-Maximum Suppression A threshold value is set for NMS to determine how far two candidate frames overlap before they are considered to mark the same target. For this subject, the overlap of defective images is low, and the NMS threshold can be set relatively low. After several tests, better experimental results were achieved when the threshold value was set to 0.15.

Backbone Network Comparison Test The VGG16, VGG19, Resnet50 and ResNet101 networks of the backbone network were selected for the control experiments, and the experimental results are shown in Table 1. The VGG19 backbone network achieves the highest accuracy mean in this subject. Compared with other networks, its network layers are not deep and the number of parameters is moderate, which possesses better detection efficiency. Therefore, the VGG19 backbone network is selected for this study.

Backbone	Stain/%	Thread/%	Hole/%	Average/%
VGG16	0.8794	0.7681	0.8835	0.8437
VGG19	0.9916	0.8545	0.8744	0.9068
ResNet50	1.0000	0.6904	0.9062	0.8655
ResNet101	0.9964	0.7585	0.8977	0.8842

Table 1: The impact of backbone network on accuracy

Loss Function

In the specific defect classification in this Faster R-CNN network, it is not guaranteed that the number of samples of defects is balanced with similar difficulty. Therefore, Focal Loss function optimization needs to be added to the Fast R-CNN classification network. γ is the focusing factor of Focal Loss. The impact of γ on the loss function is shown in Fig7. After comprehensive consideration, the Focal Loss parameter γ is selected as 2.0. On the basis of the Focal Loss func-

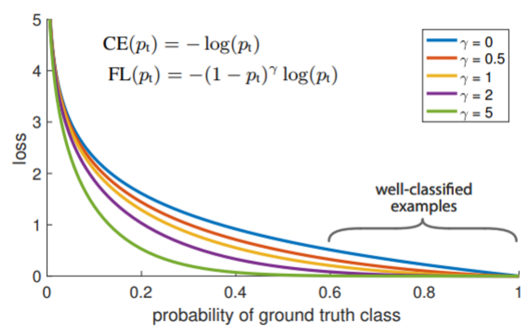

Figure 7: The influence of γ parameters on Focal Loss

tion, the weights of the loss function for different classes of targets are adjusted by adding the parameter of α_i. Therefore, the weights of the three categories of defects and background in the loss function can be set by artificially. After a comprehensive consideration, the category weights of [0.25, 0.5, 0.9, 0.7] are selected for background, stain, thread and hole, respectively.

Both Focal Loss and Reduced Focal Loss have the ability to mine difficult samples, and Reduced Focal Loss also has the ability to improve the recall rate. In order to compare the effects of the two loss functions, select a more appropriate loss function, and test the two functions and four different backbone networks respectively. The experimental results are shown in Table 2 and Table 3.

Backbone	Stain/%	Thread/%	Hole/%	Average/%
VGG16	0.8672	0.7979	0.8948	0.8533
VGG19	0.8794	0.8831	0.8723	0.8783
ResNet50	0.9964	0.5864	0.8785	0.8204
ResNet101	0.9964	0.7894	0.9091	0.8983

Table 2: Focal Loss accuracy

Backbone	Stain/%	Thread/%	Hole/%	Average/%
VGG16	0.8794	0.7681	0.8835	0.8437
VGG19	0.9916	0.8545	0.8744	0.9068
ResNet50	1.0000	0.6904	0.9062	0.8655
ResNet101	0.9964	0.7585	0.8977	0.8842

Table 3: Reduced Focal Loss accuracy

Analysis of Results

Evaluation Indicators The main evaluation metrics for this defect detection topic are the mean average accuracy (mAP), and the mean accuracy, AP, for each type of defect target. mAP and AP are determined by accuracy and recall.The mathematical expression of the accuracy and recall is shown in Equation 7 and Equation 8 .

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

Results The average accuracy of "stain", "thread" and "hole" are 0.9916, 0.8545 and 0.8744, respectively, as shown in Figure 8. Using visdom to visualize the test set prediction results, the prediction results are shown in Figure 9.

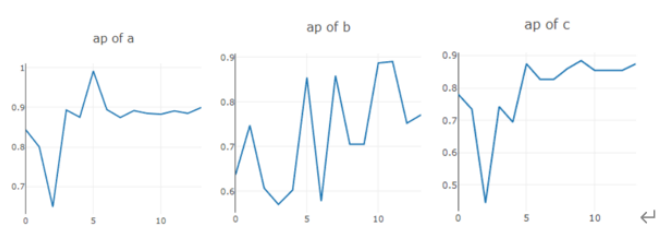

Figure 8: Accuracy of stain, thread and hole

Conclusion

In this paper, Faster R-CNN network is improved to detect mattress defects.

Dataset augmentation strategies are adopt, and augmented datasets can more correctly and extensively represent the features of defects, allowing the neural network to learn the general patterns of the features.

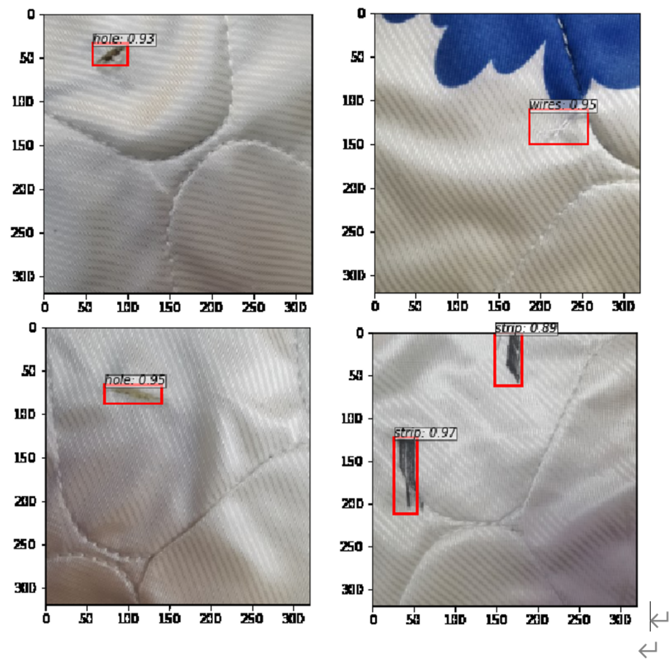

Figure 9: Results of prediction

For the experimental dataset, a number of different backbone networks, VGG16, VGG19, ResNet50 and ResNet101 were compared and tested. The results show that VGG19 takes into account the mAP and detection efficiency of the network, and is more suitable for the feature extraction backbone in this project.

The K-means algorithm is adopt to clusters the defective targets of the dataset, and the 3 cluster centres are calculated to obtain information about the targets in the corresponding feature maps with the aim of setting more reasonable anchor scale and aspect ratio.

For different categories of defects with different classification difficulties, the percentage of difficult samples is relatively small. Using the Reduced Focal Loss loss function, the category weights are set separately for the background target and each category of defective targets, and the weight share of the classification difficult category is artificially increased. The parameters and thresholds of the Reduced Focal Loss function are also set so that the classification difficult samples occupy a greater weight in the loss function, avoiding the problem of a large number of simple samples dominating the direction of the loss function.

The network's detection accuracy was improved through constant trial and error through comparative experiments, achieving mAP of 90.68%, with an average of 99.16% for "stains", 85.45% for "threads" and 87.44% for "holes". The average accuracies for "stain", "thread" and "hole" were 99.16%, 85.45% and 87.44% respectively. In the training phase and detection phase, the computing efficiency reached 9.78 fps and 23.18 fps respectively.

References

Ahmed, J.; Gao, B.; and Woo, W. L. 2018. Wavelet-integrated alternating sparse dictionary matrix decomposition in thermal imaging CFRP defect detection. *IEEE Transactions on Industrial Informatics*, 15(7): 4033–4043.

Gao, B.; Lu, P.; Woo, W. L.; Tian, G. Y.; Zhu, Y.; and Johnston, M. 2018. Variational Bayesian subgroup adaptive sparse component extraction for diagnostic imaging system. *IEEE Transactions on Industrial Electronics*, 65(10): 8142–8152.

Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.

Hamdi, A. A.; Sayed, M. S.; Fouad, M. M.; and Hadhoud, M. M. 2018. Unsupervised patterned fabric defect detection using texture filtering and K-means clustering. In *2018 international conference on innovative trends in computer engineering (ITCE)*, 130–144. IEEE.

Jiang, P.; Ergu, D.; Liu, F.; Cai, Y.; and Ma, B. 2022. A Review of Yolo algorithm developments. *Procedia Computer Science*, 199: 1066–1073.

Li, Z.; Peng, C.; Yu, G.; Zhang, X.; Deng, Y.; and Sun, J. 2018. Detnet: A backbone network for object detection. *arXiv preprint arXiv:1804.06215*.

Mei, S.; Wang, Y.; and Wen, G. 2018. Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model. *Sensors*, 18(4): 1064.

Rasheed, A.; Zafar, B.; Rasheed, A.; Ali, N.; Sajid, M.; Dar, S. H.; Habib, U.; Shehryar, T.; and Mahmood, M. T. 2020. Fabric defect detection using computer vision techniques: a comprehensive review. *Mathematical Problems in Engineering*, 2020.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Şeker, A.; Peker, K. A.; Yüksek, A. G.; and Delibaş, E. 2016. Fabric defect detection using deep learning. In *2016 24th Signal Processing and Communication Application Conference (SIU)*, 1437–1440. IEEE.

Wang, Y.; Gao, B.; lok Woo, W.; Tian, G.; Maldague, X.; Zheng, L.; Guo, Z.; and Zhu, Y. 2018. Thermal pattern contrast diagnostic of microcracks with induction thermography for aircraft braking components. *IEEE Transactions on Industrial Informatics*, 14(12): 5563–5574.

Zhang, J.; Su, H.; Zou, W.; Gong, X.; Zhang, Z.; and Shen, F. 2021. CADN: A weakly supervised learning-based category-aware object detection network for surface defect detection. *Pattern Recognition*, 109.