

Article

Research on Tiny Target Detection Technology of Fabric Defects Based on Improved YOLO

Xi Yue ^{1,2}, Qing Wang ¹, Lei He ^{1,2}, Yuxia Li ³ and Dan Tang ^{1,2,*}

¹ School of Software Engineering, Chengdu University of Information Technology, Chengdu 610225, China; yuexi@cuit.edu.cn (X.Y.); 3200706043@stu.cuit.edu.cn (Q.W.); helei1978@cuit.edu.cn (L.H.)

² Sichuan Province Engineering Technology Research Center of Support Software of Informatization Application, Chengdu 610225, China

³ School of Software Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; liyuxia@uestc.edu.cn

* Correspondence: tangdan@cuit.edu.cn

Abstract: Fabric quality plays a crucial role in modern textile industry processes. How to detect fabric defects quickly and effectively has become the main research goal of researchers. The You Only Look Once (YOLO) series of networks have maintained a dominant position in the field of target detection. However, detecting small-scale objects, such as tiny targets in fabric defects, is still a very challenging task for the YOLOv4 network. To address this challenge, this paper proposed an improved YOLOv4 target detection algorithm: using a combined data augmentation method to expand the dataset and improve the robustness of the algorithm, obtaining the anchors suitable for fabric defect detection by using the k-means algorithm to cluster the ground truth box of the dataset, adding a new prediction layer in yolo_head in order to have a better effect on tiny target detection, integrating a convolutional block attention module into the backbone feature extraction network, and innovatively replacing the CIoU loss function with the CEIoU loss function to achieve accurate classification and localization of defects. Experimental results show that compared with the original YOLOv4 algorithm, the detection accuracy of the improved YOLOv4 algorithm for tiny targets has been greatly increased, the AP value of tiny target detection has increased by 12%, and the overall mean average precision (mAP) has increased by 3%. The prediction results of the proposed algorithm can provide enterprises with more accurate defect positioning, reduce the defect rate of fabric products, and improve their economic effect.

Keywords: fabric defect; YOLOv4; CBAM; CEIoU; tiny target



Citation: Yue, X.; Wang, Q.; He, L.; Li, Y.; Tang, D. Research on Tiny Target Detection Technology of Fabric Defects Based on Improved YOLO. *Appl. Sci.* **2022**, *12*, 6823. <https://doi.org/10.3390/app12136823>

Academic Editor: João M. F. Rodrigues

Received: 9 June 2022

Accepted: 1 July 2022

Published: 5 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 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

Since the reform and opening up, the textile industry has always been a traditional pillar industry of the Chinese national economy, and it is also an industry with obvious international competitive advantages. It plays an important role in prospering the market, expanding exports, absorbing employment and promoting urbanization. In the process of fabric production, the inspection of fabric defects is a key factor in determining the quality of fabrics. The traditional artificial detection of fabric defects are generally inefficient, high cost and have a high detection error rate. If there have some defects in the surface of the fabric, then the cost price will increase by about 45–60% [1], which will reduce the economic effect of the enterprise. Therefore, the textile industry is in urgent need of a new solution, such as an automatic fabric defect detection system, which can not only significantly reduce the detection time of the same batch of fabric compared to human-based detection, but also improve the detection rate of defects, reducing the operating cost of the enterprise, and improving the overall profit.

Research on the algorithm of automatic fabric detection has been ongoing for many years, and some relatively mature methods have already been published. Various texture

backgrounds, differences in the production environment of enterprises, and classification of various fabric defects have always restricted the accuracy and speed of automatic detection of fabric defects. In recent years, experts and scholars have conducted research in this field, and many different detection methods have been produced. The three most mainstream categories of these methods are: spectrum analysis [2,3], model analysis [4,5] and deep learning-based methods [6–8]. Although most industrial production currently uses the first two methods, the deep learning-based methods have greatly surpassed them in terms of detection speed and detection accuracy, and have begun to be applied to industrial fabric defect detection on a large scale.

Since its introduction, deep learning has been widely used in related research, such as in the medical field [9–13], various types of target recognition [14–16] and natural language processing [17], and has achieved quite good results in these fields. Therefore, using the deep learning method to detect fabric defects will result in more accurate detection results.

In the aspect of using a basic deep learning model on fabric detection, Bu et al. [18] proposed a data description model based on support vector; the optimal Gaussian kernel parameters are selected during training to solve the shortcomings of fabric defects that cannot be handled by a single fractal, however, this method is not particularly effective for image classification with strong textures. Huang et al. [19] proposed an efficient convolutional neural network for defect segmentation and detection; this framework can significantly alleviate the amount of pictures needed while training the neural network and obtain the location of defects with high accuracy. The results show that this network significantly outperforms eight state-of-the-art methods in terms of accuracy and robustness. The methods introduced above use deep learning-based models. Although the detection effect is improved compared with the traditional methods, the basic deep learning model is incompetent for high-precision detection tasks such as Drone-captured Scenarios [20], and the actual detection effect cannot meet the requirements of accurately locating defects.

Therefore, replacing the basic model with a classical model which has a more stable detection effect is the research direction in recent years. Wei et al. [21] proposed a faster region-based convolutional neural network with a visual gain mechanism (Faster VG-RCNN). By analyzing the relationship between the attention mechanism and the visual gain mechanism, it was found that the attention-related visual gain mechanism can change the corresponding magnitude without changing the selectivity, and improve visual perception. Compared with Faster-RCNN, the experimental results show that the detection accuracy is improved by about 4%. Jing et al. [22] proposed a fabric defect detection algorithm based on improved YOLOv3, first using the K-Means algorithm to determine the suitable anchors, then combining the low-level and high-level information, and adding the Yolo detection layer to the feature maps of different sizes. After the improvement, the false detection rate for specific types of fabric is less than 5%. Wang et al. [23] proposed a fabric defect detection algorithm based on improved Yolov5. The Adaptive Spatial Feature Fusion (ASFF) method is used to improve the bad effect of Path Aggregation Network (PANet) on multi-scale features, and an attention mechanism is added to make the network focus on useful information. The final experimental results show that the average accuracy rate in fabric defect detection is 71.70%. Dlamini et al. [24] proposed an improved YOLOv4 network: first, preprocess the flawed image to decompose the image into smaller sizes, and then use the filtering method to denoise the flawed features to enhance the robustness of the model, then deploy the trained model to the hardware; the accuracy of detecting specific flaws reached 95.3%. Kahraman et al. [25] proposed a capsule Network, unlike the traditional CNN, which causes information loss, the capsules are capable of holding more information and are a group of neurons that include not only the probability of a particular object's presence, but also different informative values related to instantiation parameters. The result shows the CapsNet achieved a performance value of 98.7%. Zheng et al. [26] proposed SE-YOLOv5 network, adding the Squeeze-and-Excitation Networks (SE) [27] module to the backbone network of YOLOv5 and replacing the ReLU activation function in the Cross Stage Partial (CSP) module with the Activate or Not (ACON) [28] activation

function. Experimental results show that the accuracy, generalization ability, and robustness are improved. After using the classic model and improving it, it can be seen that the accuracy of the detection results has been greatly improved, and the requirements of accurate positioning of defects can be achieved when detecting ordinary defect targets, however, when encountering tiny targets, the detection effect will be greatly reduced.

In the field of tiny target detection, Cui et al. [29] proposed an improved YOLOv4 network to detect tiny targets in transmission line faults, adding an attention module combining ECAM (Efficient Channel Attention Module) and CBAM (Convolutional Block Attention Module) into the backbone network, using the clustering algorithm to regenerate Anchors, and, using a new loss function, the mAP50 for tiny target transmission line fault detection reaches 95.16%. Moran et al. [30] improved the YOLOv3 network for tiny target prediction problems; the feature map output extracted by the second-layer residual network of the backbone is fused with the feature map output of other residual layers to form a new feature prediction layer, and the anchors are clustered using a clustering algorithm. Great results have been achieved in tiny target prediction from satellite image. Xu et al. [31] proposed to add an attention mechanism and a feature extraction auxiliary network composed of multiple residual blocks. A smaller scale than the backbone network in the bypass of the original YOLOv3 backbone features extraction networks on the tiny target problem, then the auxiliary network will transfer the extracted location features to the backbone network to learn the detailed features of tiny targets more accurately. The mAP of tiny target detection in the car driving scene reaches 84.76%, Zhu et al. [20]. In the tiny target detection for the drone-capture scenario, the TPH-YOLOv5 model is proposed. They add one more prediction head to detect different-scale objects, then use a Transformer Prediction Head (TPH) to replace the original prediction heads to excavate the prediction potential of the network; the results show this model is better than previous SOTA methods by 1.81%. The above models in their respective fields have made corresponding improvements to the structure of the model, according to the characteristics of the tiny targets in their own field datasets, so that the model can more accurately detect the tiny targets existing in the image. In the field of fabric defects, there are few model improvements specifically related to tiny targets.

Based on the above research, an improved YOLOv4 network is proposed to achieve accurate detection of tiny objects with fabric defects. The first step uses a new data augmentation method to expand the dataset to enhance the generalization ability of the model, the second step uses a clustering algorithm to cluster the augmented dataset and selects suitable yolo_anchors, the third step is to add a layer of feature detection specifically for tiny targets in the prediction network. The fourth step is to add a convolutional attention module to the backbone network to enhance the ability of the network to extract features. Finally, the loss function of the network is optimized to speed up the network convergence.

2. Materials and Methods

2.1. Dataset

The fabric defect detection dataset contains a total of more than 2000 pictures with fabric defects, of which more than 1000 pictures are from the public Tianchi dataset, and the remaining 1000 pictures are from the use of a web crawler. After crawling, the high-quality pictures were screened out and labeled, after querying the relevant information on fabric defects, and then added to the total dataset. After sorting and classifying all pictures using the VOC dataset format, there were a total of 15 categories. The names of the labeled images correspond to the labeled xml files one-to-one, and are divided into a training set, test set and validation set according to the ratio of 7:2:1. Some of defect images are shown in Figure 1.

2.2. Structure of YOLOv4

The YOLO [32] series network is one of the most efficient network structures in the single-stage object detection network, the more widely used versions in the industry are YOLOv3 [33] and YOLOv4 [34], other single-stage object detection networks include

Single Shot Multibox Detector (SSD) [35], Fully Convolutional One-Stage Object Detection (FCOS) [36], EfficientDet [37], and so on.

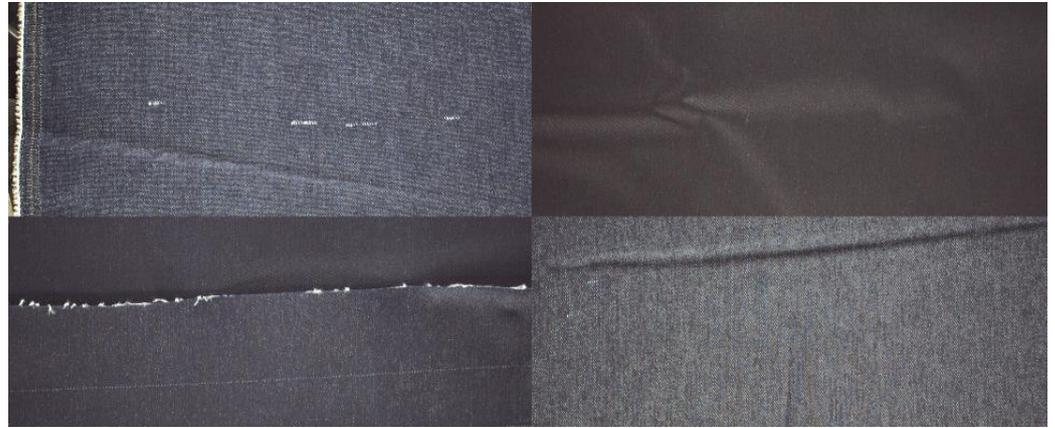


Figure 1. Some iconic defect images.

The YOLOv4 object detection network is mainly composed of the following three parts, the backbone feature extraction network CSPDarknet53, the spatial pyramid pooling layer (SPP), and the feature enhancement network PANet [38].

The CSPDarknet network is formed by adding the CSP module to the Darknet network. The CSP module can enhance the learning ability of Convolutional Neural Network (CNN), making the network more lightweight while maintaining accuracy, reducing computational bottlenecks, and reducing memory costs.

The SPP structure consists of a CBL module and pooling kernels of different sizes. The pooling kernel sizes are 1×1 , 5×5 , 9×9 , and 13×13 , respectively. After CSPDarknet outputs the features of the last layer, the CBL module is used. Perform 3 convolutions on it, and then use these four pooling kernels of different sizes for processing, which greatly increase the receptive field and separates significant contextual features. The CBL convolutional network module is composed of a convolutional network, batch normalization and Leaky ReLU activation function.

PANet is actually an instance segmentation algorithm, which mainly uses up-sampling and down-sampling so that high-resolution features can obtain richer semantic information from low-resolution features, and low-resolution features can be extracted from high-resolution features to obtain more accurate location information.

The YOLOv4 network uses Mosaic data augmentation at the input to expand the dataset. The activation function selects the ReLU [39] activation function that has been widely used in recent years, and uses the idea of cosine annealing to adjust the learning rate.

3. Methods

3.1. Data Augmentation

The YOLOv4 network uses Mosaic data augmentation at the input. The principle is to select 4 pictures in the dataset, and then place the four pictures in the four corners of a canvas after scaling, rotating and color gamut changes. The advantage is that this can enrich the background information of the entire detected object, and can calculate all the data of 4 pictures at the same time when calculating the picture. The Mosaic data augmentation is shown in Figure 2.

In order to solve this kind of problem, this paper plans to add a layer of 104×104 feature output layer to change the minimum detection pixel value capability of the network to 4×4 . The specific implementation method is that after the input image passes through the second Resblock_body of the CSPDarknet53 feature extraction network, not only outputting the extracted features to the next Resblock_body, but these output features are also fused with the features of other feature layers in PANet which have been processed by

the CBL module and upsampling. The fused features will have richer semantic information, and then the feature layer will be downsampled and then passed down and fused with other feature layers. Finally, the feature layer of 104×104 will be output to yolo_head for prediction. Strengthening the network's detection ability for extremely tiny targets, and improving the overall performance of the network for defect detection.

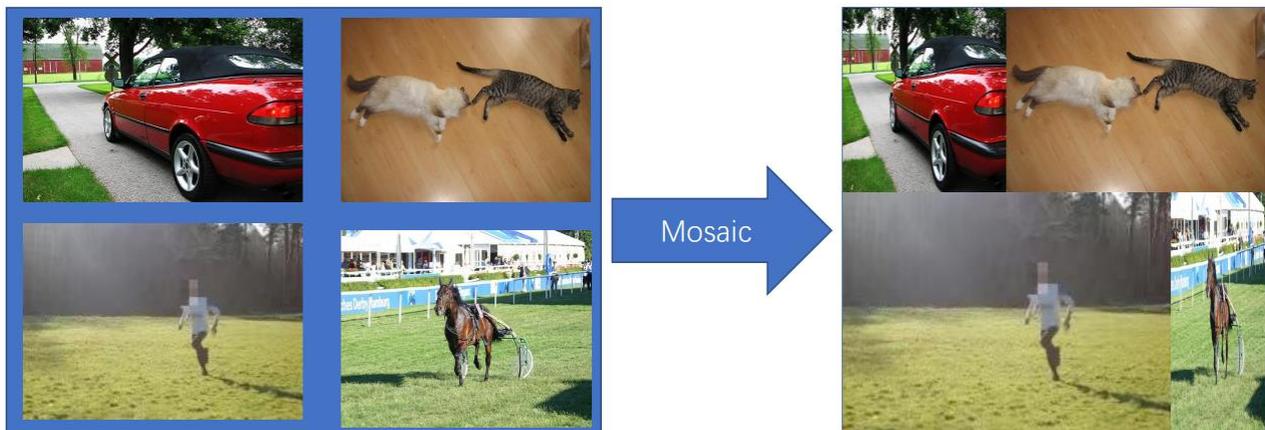


Figure 2. Mosaic data augmentation.

However, in the real production environment, there will be some very tiny targets such as knots and very large targets such as flower board jumps. In the case of 416×416 resolution, the real ground truth box of the tiny target is only about 6×6 on average, if Mosaic data enhancement is used to zoom the image, the already very small real frame will be compressed to a smaller size, which greatly hinders the network's ability to detect such extremely small targets. At the same time, the average real frame of the extremely large target is Around 416×50 , because the Mosaic dataset will crop the original image, this may make the network only train the local information of the large target and cannot pay attention to the global information of the large target.

Therefore, Mosaic data augmentation is no longer used in this paper, and a new data augmentation method that is more suitable for the dataset in this paper is proposed. Four augmentation methods are randomly selected among the 5 methods of impulse noise, Gaussian blur, mirror flip, Multiply and Affine. Four selected methods combined to work on one image simultaneously, enhance the original image N times, and also correspondingly transform the ground truth box in the original image; N is the number of enhancements for each image. Multiply is to multiply each pixel in the image by a preset value to make it look brighter or darker, and Affine is to zoom in, zoom out, move up and down, left and right, and rotate the original image.

After data augmentation, the network can learn more location features and pixel features that the original image does not have, and improve the robustness and accuracy of the network for tiny target detection.

3.2. New PredictionLayer for YOLO Head

In the YOLOv4 network, after feature extraction, the backbone network will output three feature layers of different sizes, and then output to yolo_head for prediction after feature enhancement networks such as SPP and PANet. The sizes of these three feature layers are 13×13 , 26×26 , and 52×52 ; if the input image size is 416×416 , the smallest pixel value that the network can detect is 8×8 .

In the real production environment, there will be a large number of extremely tiny targets with an average ground truth box size of only 5×5 , and if the size of the input image is too small, the YOLO network will automatically enlarge the image and add grayscale bars, but this will distort tiny objects in the image and interfere with the network's ability to detect them.

3.3. Attention Mechanism

CBAM [40] (Convolutional Block Attention Module) consists of a Channel Attention Module and a Spatial Attention Module. These two modules strengthen effective feature information and suppress invalid information in the channel dimension and spatial dimension of deep features, respectively.

In the CBAM module, the input features first go through the channel attention module, and then the channel refined attention features will input into the spatial attention module, after that we will obtain the final refined features.

The structure of channel attention module is shown in Figure 3.

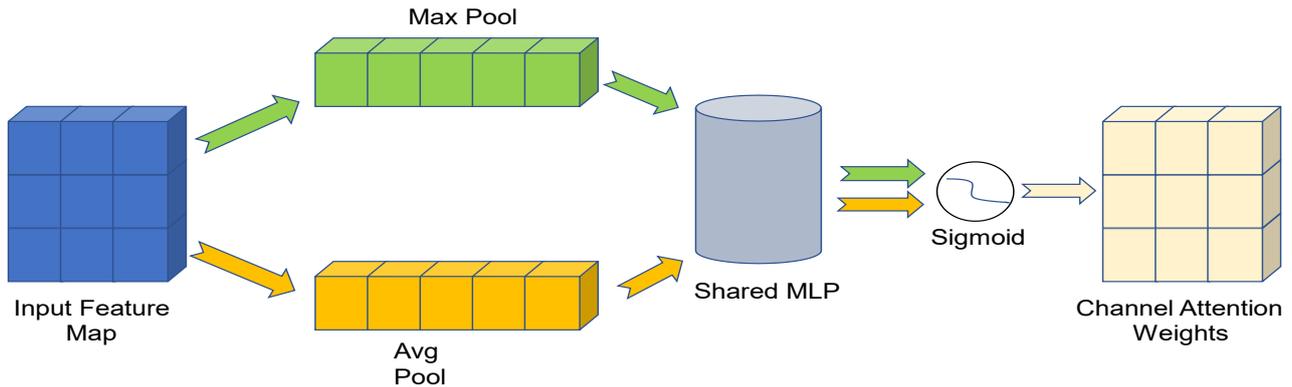


Figure 3. Channel attention module.

Perform a maximum pooling and an average pooling operation on the input feature map in the spatial dimension, respectively, and then pass the pooled output through a shared weight MLP (multi-layer perceptron), and then also output the elements of the two outputs one by one. After the elements are added and passed through the sigmoid function, the channel attention weight is obtained. The formula of the channel attention module is shown in (1).

$$M_c(F) = \sigma\left(W1(W0(F_{avg}^c))\right) + \left(W1(W0(F_{max}^c))\right) \tag{1}$$

where $W1$ and $W0$ represent two convolution operations, F_{avg}^c and F_{max}^c represent average pooling and maximum pooling at the channel level, respectively.

The structure of the spatial attention module is shown in Figure 4.

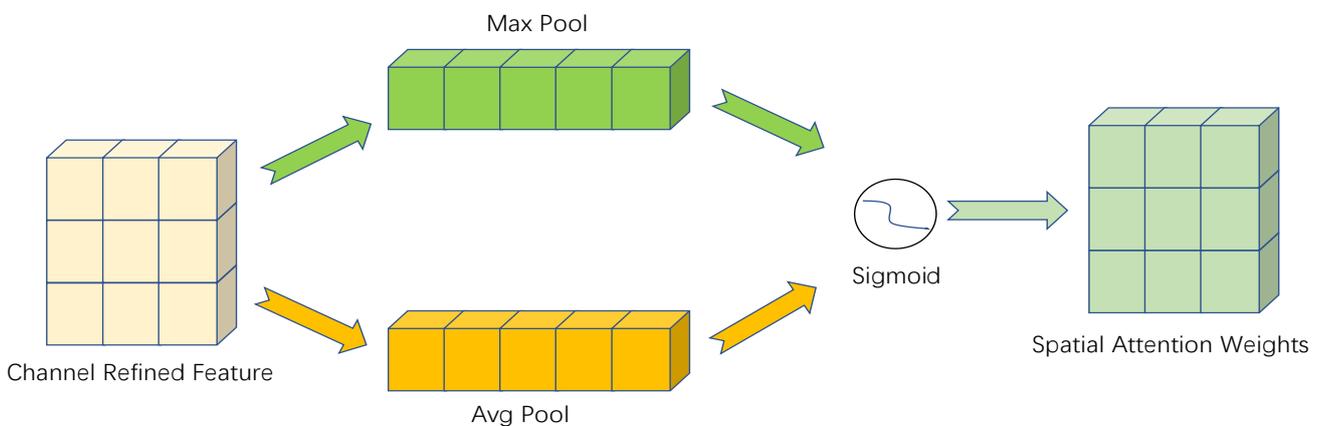


Figure 4. Spatial attention module.

Perform a maximum pooling and average pooling operations on each channel of the Channel refined feature map, and then concatenate those two pooled feature maps, convolution operations and sigmoid activation will be performed on the spliced feature maps. After the above operations, a spatial attention weight with a channel of 1 and

the same size as the input feature map can be obtained; finally, an element-by-element multiplication with the input feature map is performed. The formula of the spatial attention module is shown in (2).

$$M_s(F) = \sigma\left(f^{7 \times 7}(F_{avg}^s; F_{max}^s)\right) \tag{2}$$

where σ represents the sigmoid activation function, $f^{7 \times 7}$ represents the convolution kernel size is 7×7 , F_{avg}^s and F_{max}^s represent average pooling and maximum pooling at the spatial level, respectively.

In this paper, the convolutional attention module is integrated into the feature output layer of each dimension output by the CSPDarknet backbone network, and the output feature after spatial pyramid pooling. It makes the network pay more attention to key pixel features when predicting different feature layers, and improves the prediction accuracy of the network. The improved YOLOv4 network structure is shown in Figure 5.

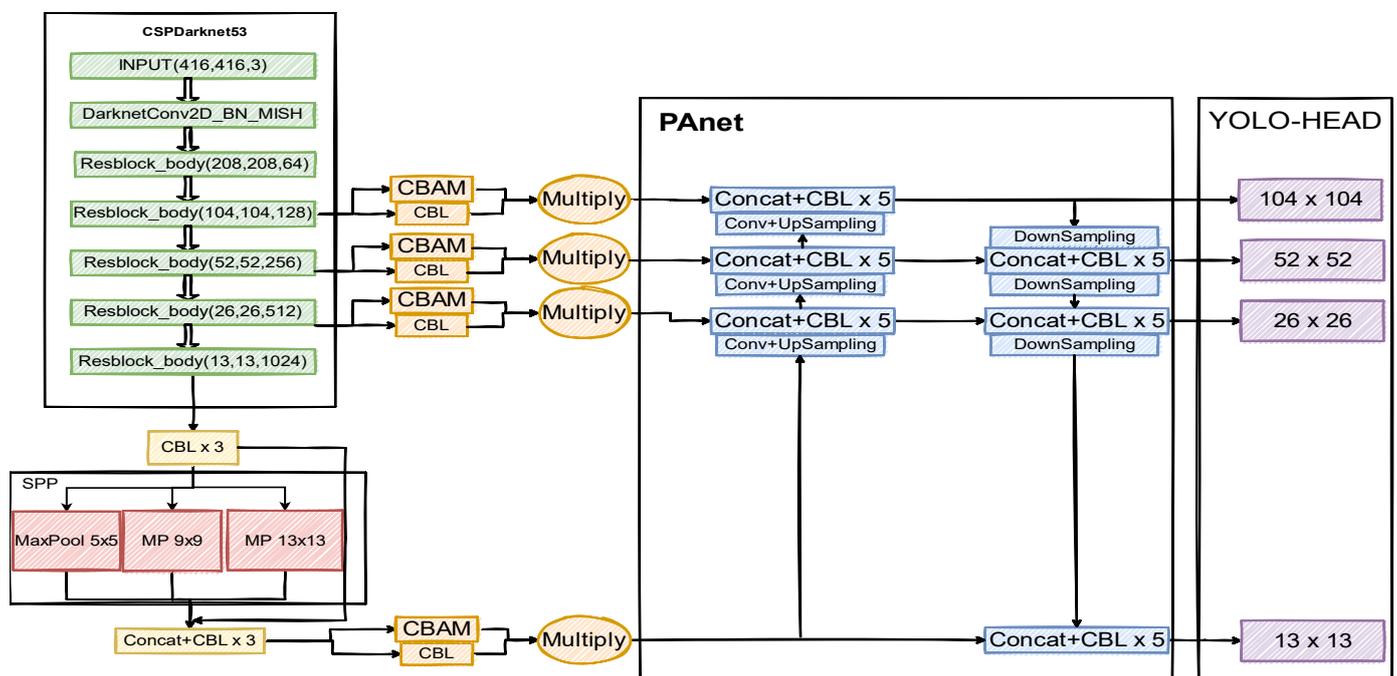


Figure 5. Improved YOLOv4 network structure.

3.4. Clustering Anchors

In the prediction process of the original YOLOv4 network, each YOLO Head prediction feature layer corresponds to 3 priori boxes preset in the yolo_anchors file. Among them, the 9 default anchors in the yolo_anchors file are obtained by the author through clustering all the ground truth boxes in the VOC dataset. These 9 anchors are universal to most object detection situations and will not change in the process of training and prediction.

However, for this dataset, because the extremely tiny targets account for about 20% of all data, the accurate detection of the extremely tiny targets is an important indicator to test the improved network prediction effect. Because this paper adds a new feature layer to the original 3 prediction feature layers, the original yolo_anchors can no longer meet the needs of this experiment.

Therefore, this paper uses the K-means clustering algorithm to cluster the ground truth boxes in the augmented dataset, randomly select 12 points as the cluster center of the algorithm, calculates the similarity between each ground truth boxes and the cluster center, assigns these ground truth boxes to the cluster center with the closest similarity, calculates the mean of all samples in each cluster, and then updates the center of the cluster. Repeat the first two steps until the centers of the 12 clusters no longer change, or the

maximum number of iterations set is reached, then the clustering ends, and the values of the 12 anchors we need are obtained.

The twelve Anchor values obtained in this paper using the K-means algorithm at multiple different feature layer scales are shown in Table 1.

Table 1. Anchors for each feature layer.

Feature Layer	Anchors
104 × 104	(4,6) (7,10) (5,19)
52 × 52	(4,38) (5,22) (16,38)
26 × 26	(28,400) (72,119) (64,168)
13 × 13	(36,349) (122,30) (408,103)

3.5. Optimized Loss Function

The loss function used in the YOLOv4 network is the Complete Intersection over union (CIOU) loss function. On the basis of DIOU, CIOU considers the aspect ratio of the detection frame into the loss function, and increases the aspect ratio influence factor $\alpha\nu$, the accuracy of the frame regression has been further improved.

CIOU is represented by Equation (3).

$$CIOU = IOU - \frac{\rho^2(b_{pred}, b_{gt})}{c^2} + \alpha\nu \tag{3}$$

IOU represents the intersection ratio between the ground truth box and the prediction box, ρ represents the Euclidean distance, b_{pred} and b_{gt} represent the center point of the ground truth box and the prediction box, c represents the diagonal distance of the smallest circumscribed rectangle, ν is used to measure the consistency of the aspect ratio, and α is used to balance the value of ν .

Although CIOU takes into account the overlapping area, the center point distance and aspect ratio of the frame regression, the aspect ratio difference reflected by ν in its formula is not the real difference between the width and height and its confidence.

Therefore, sometimes it will hinder the model from effectively optimizing the similarity and slow down the convergence speed of the model, so this paper replaces the CIOU with the EIOU which is more sensitive to the aspect ratio of the detection frame and has a faster convergence speed.

EIOU is represented by Equation (4), $EIOU_{loss}$ is represented by Equation (5).

$$EIOU = IOU + \frac{\rho^2(b, b^{gt})}{c^2} + \frac{\rho^2(w, w^{gt})}{C_w^2} + \frac{\rho^2(h, h^{gt})}{C_h^2} \tag{4}$$

$$EIOU_{loss} = 1 - EIOU \tag{5}$$

C_w represents the width of the minimum bounding box covering the predicted box and the ground truth box, C_h represents the height of the minimum bounding box covering the predicted box and the ground truth box, w represents the width of the predicted box, w^{gt} represents the width of the ground-truth box, h represents the height of the predicted box, and h^{gt} represents the height of the ground-truth box.

When the value of EIOU is larger, it indicates that the overlap between the predicted box and the ground truth box is larger, the loss value is smaller, and the prediction result of the model is better.

However, in the process of backpropagation, the momentum provided by $EIOU_{loss}$ for network training is constant, and it can also be understood that the gradient of $EIOU_{loss}$ is constant, which cannot give the network a larger momentum to speed up the training of the network when the predicted box and the ground truth box are too far apart. Therefore, this paper proposes a new loss function based on EIOU, namely $CEIOU_{loss}$ (Curve $EIOU_{loss}$).

$CEIOU_{loss}$ is represented by Equation (6).

$$CEIOU_{loss} = 3 \times \ln 3 - 3 \times \ln(2 + EIOU) \quad (6)$$

The comparison of $CEIOU_{loss}$ and $EIOU_{loss}$ is shown in Figure 6.

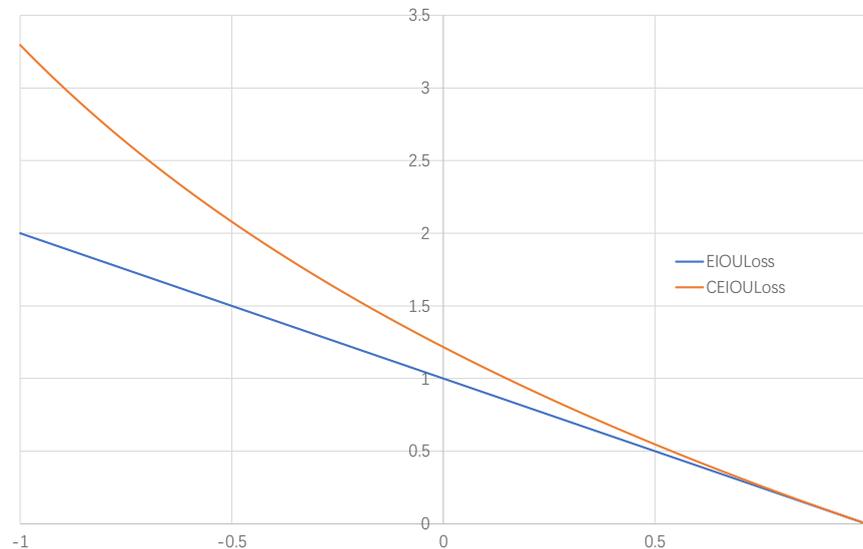


Figure 6. Comparison of $CEIOU_{loss}$ and $EIOU_{loss}$.

It can be seen from the Figure 6 that $EIOU_{loss}$ is a straight line, $CEIOU_{loss}$ is a curve and both lines decrease with the increase in EIOU, but when the EIOU is small, that is, when the gap between the predicted box and the ground truth box is large, $CEIOU_{loss}$ will provide a greater momentum to the training of the network, so that next time the network adjusts the prediction box more accurately, reducing the number of times that the predicted box and the ground truth box overlap, and Since $EIOU_{loss}$ is a straight line, the momentum it provides is equal regardless of how the magnitude of EIOU changes.

4. Results

The environment configuration for this experiment is as follows.

CPU: AMD R7-4800H; RAM: 16G; GPU: Nvidia GeForce RTX 2060 6G; Operating system: Microsoft windows 10; GPU accelerated library: CUDA 11.5 And CUDNN 11.1; Deep learning environment: Tensorflow-gpu 1.14.0 And Keras 2.2.5; IDE: Pycharm; Development language: Python 3.6.

4.1. Experimental Evaluation Criteria

In this paper, mAP50 is used as the evaluation criteria of this experiment. mAP50 is the average value of AP50 value of all classification detection results. AP50 value refers to the closed area of the precision and recall curve when the IOU threshold is 0.5. The calculation formulas of precision, recall, AP50 and mAP50 are shown as Equations (7)–(10).

$$precision = \frac{TP}{TP + FP} \quad (7)$$

$$recall = \frac{TP}{TP + FN} \quad (8)$$

$$AP50 = \int_0^1 P_r dr, IOU \geq 0.5 \quad (9)$$

$$mAP50 = \frac{\sum_{i=1}^n AP50_i}{N} \quad (10)$$

TP (True Positives) represents the number of targets actually detected in this dataset, *FP* (False Positives) represents the number of targets detected incorrectly by the model, and *FN* (False Negatives) represents the number of targets missed by the model during detection.

4.2. Training Process

The training of YOLOv4 uses the idea of transfer learning. Transfer learning refers to the transfer of a model after a large number of datasets and iterative training in a known field to a target field with similar characteristics. Using this pre-trained weight can enable the network model to quickly obtain feature information in new feature fields, reducing the requirements for the number of datasets and training time during network training to a certain extent, and accelerating the network convergence speed.

Therefore, this paper will use the already trained YOLOv4 pre-training weights for transfer training. First, the training in the freezing stage is performed, the parameters of the backbone part of the YOLO network will not change with training, and the rest of the parameters will be fine-tuned with training to adapt the network to the new dataset and speed up the initial learning speed of the network for the dataset. Set the batch size to 16, epoch to 100, and use the early stop mechanism, that is, if the loss value drops below the threshold for multiple consecutive epochs, the training in the freezing phase will be stopped. The learning rate is set to 1×10^{-3} , and the cosine annealing learning rate is added, a larger learning rate will decrease at a slower rate. Then in the middle, the learning rate will decrease faster, and finally the learning rate will decrease slowly, making the training of the network more stable. In the unfreezing phase, the backbone of the network will be unfrozen, and all the parameters of the network will participate in the training and will be changed accordingly according to the dataset. Set batch size to 2, epoch to 100, and learning rate to 1×10^{-4} , and the rest are the same as the freezing phase.

The change in loss value in freezing phase is shown in Figure 7 and unfreezing phase is shown in Figure 8.

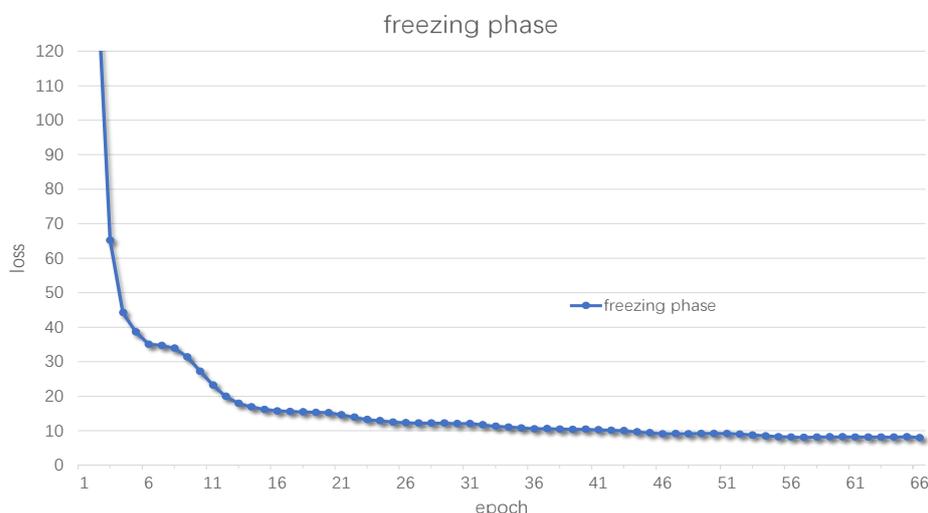


Figure 7. Loss change in freezing phase.

4.3. Experiment Results

In order to further test the performance of the improved algorithm in this paper, three sets of comparative experiments will be conducted, and each set of comparative experiments will use the same training set and test set, the same training strategy and the same evaluation criteria to ensure the validity of the test results. The first set of experiments compares the improved YOLOv4 algorithm proposed in this paper with the current mainstream object detection algorithms such as YOLOv4, YOLOv3, Faster RCNN, etc. The experiment result is shown in Table 2.

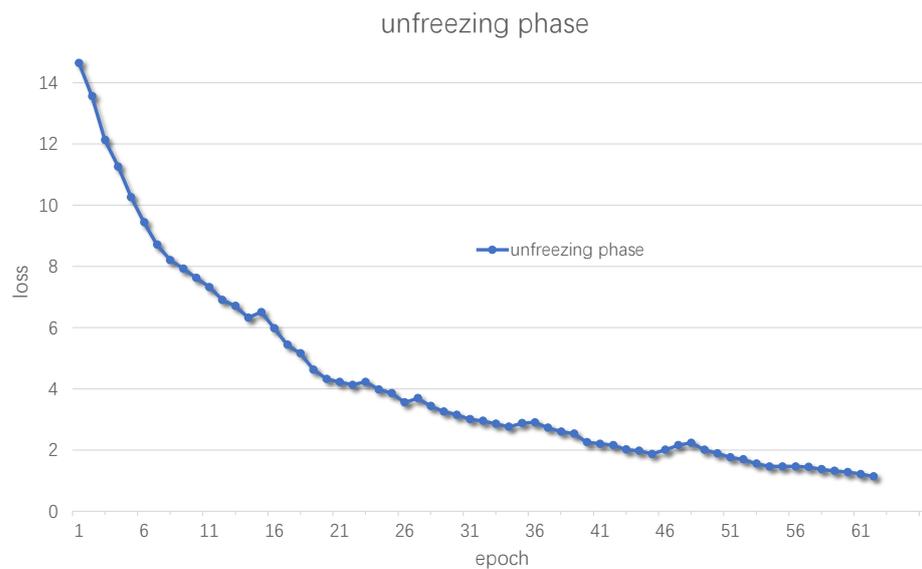


Figure 8. Loss change in unfreezing phase.

Table 2. Different network detection result.

Algorithm	mAP
SSD [35]	50.41%
YOLOv3 [33]	51.30%
Faster-RCNN [8]	51.74%
Cascade-RCNN [41]	52.20%
FPN [7]	52.96%
YOLOv4 [34]	53.49%
Mask-RCNN [6]	53.86%
Model in this paper	56.74%

The second set of experiments is to use the same dataset to compare the improved algorithm proposed in this paper with the improved algorithms for tiny targets which are proposed in other articles, and conduct comparative experiments specifically for tiny targets. The experiment result is shown in Table 3.

Table 3. Comparison of improved model.

Algorithm	AP50			mAP
	Knot	Hole	Hair	
Improved YOLOv3 [30]	55.32%	64.57%	57.40%	52.17%
Improved YOLOv4 [29]	62.40%	73.22%	59.63%	54.98%
Model in this paper	69.60%	81.70%	66.49%	56.74%

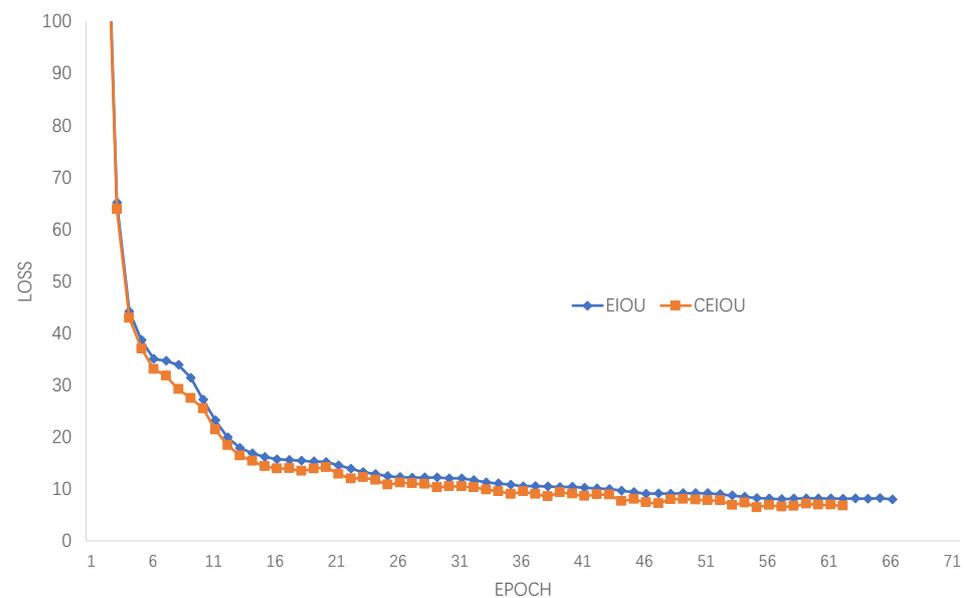
The third group of experiments is to verify the impact of the improved modules proposed in this paper on the detection of fabric defects, and to evaluate the improvement effects of each module, respectively. The experiment result is shown in Table 4.

In the third set of experiments, in order to verify the optimization effect of CEIOU loss in the training phase, EIOU and CEIOU were used to train the same dataset with the same training parameters in the frozen training phase. The obtained loss value comparison chart is shown in Figure 9.

Table 4. Comparison of improved module.

Model	Data Aug	CBAM	New-Anchors	AP50		mAP
				Hole	Knot	
model-1	✓	-	-	66.20%	57.36%	53.80%
model-2	✓	✓	-	71.84%	60.59%	54.60%
Model-3	-	✓	✓	77.35%	66.42%	55.72%
model-4	✓	✓	✓	81.20%	69.49%	56.58%

Among them, New-Anchors means adding a new feature output layer and using new yolo_anchors, '✓' means adding the corresponding module, '-' means not adding the corresponding module.

**Figure 9.** Comparison of CEIOU and EIOU in training.

5. Discussion

Analysis of Experiment Results

By comparing the experimental results, we can find out the quality of each model's detection effect on this dataset and the advantages of this paper's model in fabric defect detection.

Through Table 2, it can be seen that the algorithm in this paper has a great improvement in the overall mAP compared with the basic algorithm, with an average increase of about 4 percentage points, which can more accurately detect the fabric defects existing in the image.

Through Table 3, it can be seen that although the overall mAP improvement is only about 2 percentage points, the algorithm in this paper has great advantages in the detection results of tiny targets. Compared with other detection algorithms for tiny targets, the detection accuracy of the three groups of tiny targets is improved by about 8%, which further highlights the accuracy of the algorithm in this paper for tiny target detection.

As can be seen from the Table 4, the robustness of the model is increased due to the expansion of the training set due to data augmentation, which slightly increases the detection capability of the network. The added attention mechanism has a significant impact on the detection ability of the model. The reason is that the original algorithm does not have a focus on feature extraction and will be interfered with by many irrelevant factors. After adding an attention mechanism, the model is more effective in extracting significant features of fabric defects. After adding a feature output layer and using the new yolo_anchors, the overall mAP of the model has been significantly improved, and the detection of tiny targets has made a groundbreaking improvement. The AP value of tiny

target detection has increased by about 9% which greatly enhanced the model's ability to detect tiny objects in fabric defects.

In order to more intuitively reflect the detection performance of the improved object detection algorithm, first use the original YOLOv4 algorithm to detect the input image, and then use the improved YOLOv4 algorithm to detect the same image. Figures 10–12 shows the results of YOLOv4 and the improved YOLOv4 after detecting the same picture, where (a) is the detection result of the YOLOv4 algorithm, and (b) is the detection result of the improved YOLOv4 algorithm.

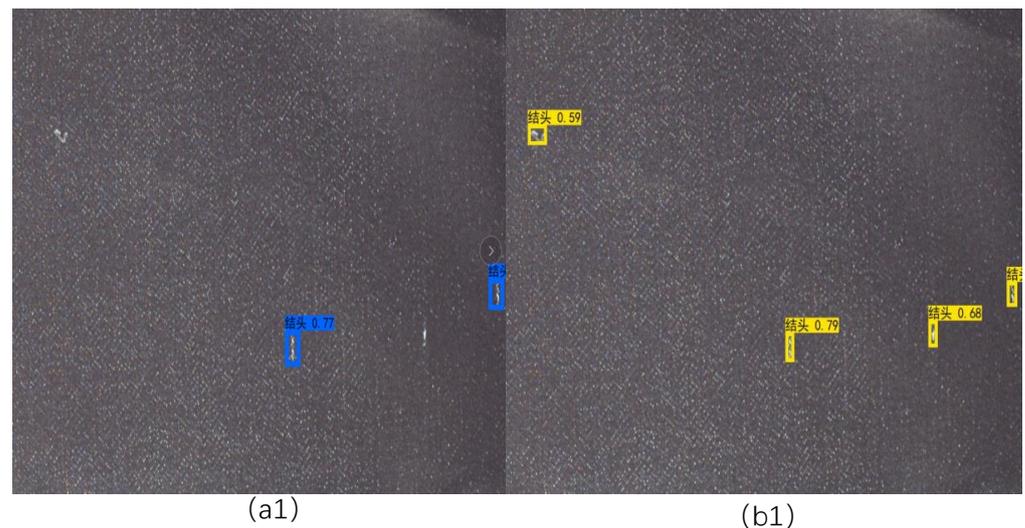


Figure 10. Experimental results of group1. (a1) detection result of YOLOv4; (b1) detection result of improved YOLOv4. Defect sizes are about 3×8 pixels.

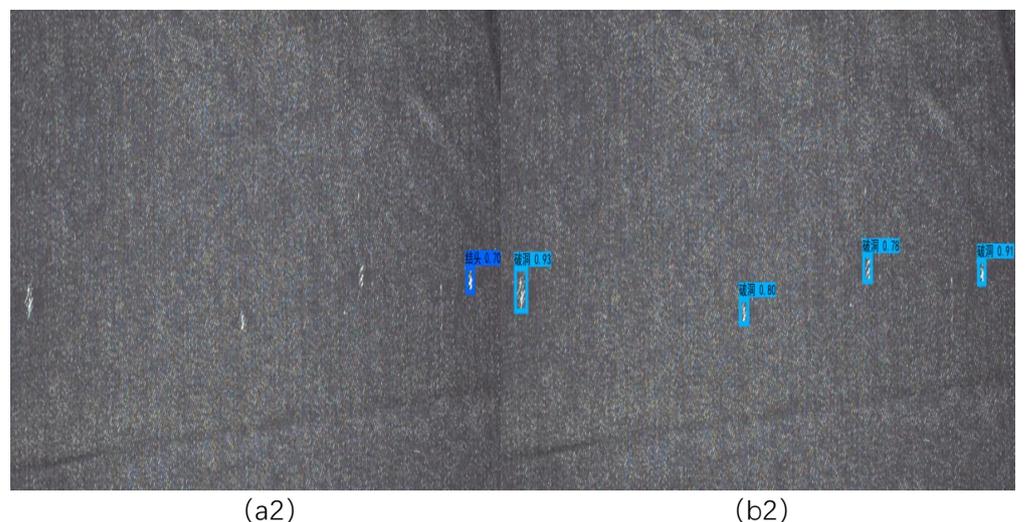


Figure 11. Experimental results of group2. (a2) detection result of YOLOv4; (b2) detection result of improved YOLOv4. Defect sizes are about 4×12 pixels.

In Figure 10, a picture of defects containing 4 defects is used, and these 4 defects are all tiny target knots. After using the YOLOv4 algorithm to detect the image, as (a1) shows that the tiny target has missed detection obviously, only 2 of 4 defects were detected. However, the images detected by the algorithm in this paper perfectly find out all the tiny targets as (b1) shown.

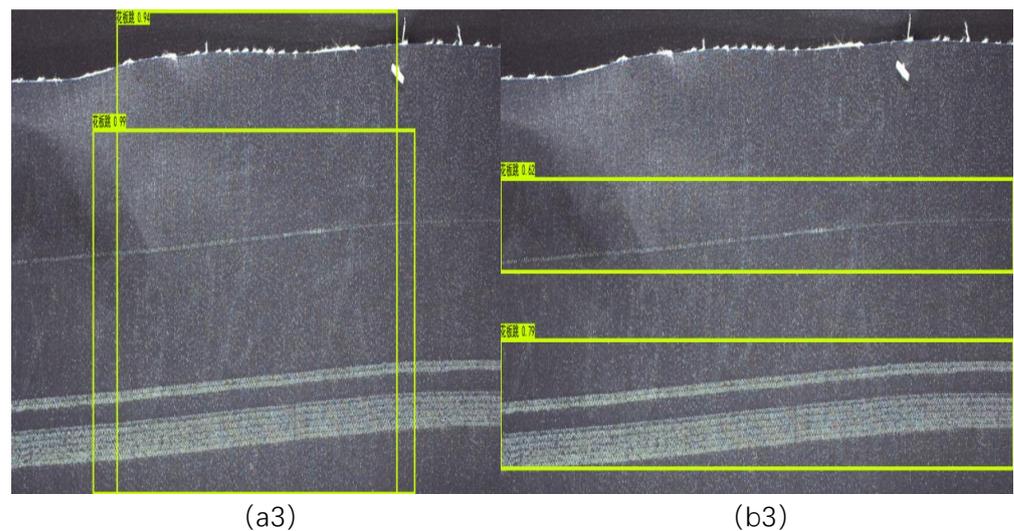


Figure 12. Experimental results of group3. (a3) detection result of YOLOv4; (b3) detection result of improved YOLOv4. Defect sizes are about 416×90 pixels.

In Figure 11, a flawed picture containing 4 tiny targets is used, and these 4 defects are all tiny target knots. After using the original YOLOv4 algorithm to detect the picture, (a2) shows that not only is there a very serious phenomenon of missed detection of tiny targets, but also a false detection occurred, where a tiny target hole was mistakenly detected as a knot. After using the algorithm in this paper to detect the picture, all the tiny targets are detected, and the detection results are correct as (b2) shows.

The third set of comparative experiments is to verify the influence of the clustered new yolo_anchors on the detection results. In Figure 12, a picture containing two defects is used, and those two defects are flower board jumps. After using the YOLOv4 network and the original anchors to detect the picture, as (a3) shows, both defect targets have been detected, but the detection frame only covers a part of the defect and does not wrap the defect perfectly, which will cause some problems in the subsequent processing. After using the algorithm in this paper and the new anchors, as (b3) shows that the algorithm not only detects the two defects, but also the two detection frames perfectly wrap the target to be detected, which completes the detection task very well.

In summary, the improved YOLOv4 detection algorithm proposed in this paper has great advantages in the face of fabric defect detection scenes and tiny target detection scenes, and can maintain a high accuracy.

6. Conclusions

Aiming at the insufficiency of the existing target detection algorithms for detecting small and medium-sized targets in fabric defects, a novel YOLOv4 object detection algorithm based on a series of improvements has been proposed, including the use of new data enhancement to expand the dataset, and the use of clustering algorithms to construct new yolo_anchors, adding a feature output layer for yolo_head to predict tiny targets, adding an attention mechanism so that the network will focus on high-weight areas and a new loss function to speed up the convergence of the model. In the comparative experiments with other improved networks for tiny targets, the overall mAP has increased by about 3 percentage points, and the mAP of tiny target detection has increased significantly, which is enough to reflect the effectiveness of this algorithm in tiny target detection.

With the support of hardware facilities in the future, a larger dataset and number of iterations can be used to enhance the detection ability of the network. The structure of the network can be further optimized to improve the detection ability of the network for regular size defects, as well as the use of an anchor-free method to reduce calculations while predicting and obtaining prediction boxes more accurately, or to make the network

more easily transplanted into simple devices and mobile terminals to achieve faster and real-time detection.

Author Contributions: Conceptualization, X.Y.; methodology, X.Y.; software, Q.W.; validation, Q.W.; formal analysis, Y.L.; investigation, X.Y.; resources, Y.L.; writing—original draft preparation, Q.W. and D.T.; visualization, Q.W.; writing—review and editing, X.Y. and Q.W.; supervision, D.T.; funding acquisition, L.H. All authors have read and agreed to the published version of the manuscript.

Funding: Supported by the Research and application of deep learning open sharing platform based on natural language processing(2020YFQ0056), the Key Projects of Global Change and Response of Ministry of Science and Technology of China under Grant 2020YFA0608203, in part by the Science and Technology Support Project of Sichuan Province under Grant 2021YFS0335.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available from the corresponding author. Send a request in corresponding author's email, then you will receive the data.

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

References

1. Tiwari, V.; Sharma, G. Automatic fabric fault detection using morphological operations on bit plane. *Int. J. Comput. Sci. Netw. Secur. (IJCSNS)* **2015**, *15*, 30.
2. Ngan, H.Y.; Pang, G.K.; Yung, S.; Ng, M.K. Wavelet based methods on patterned fabric defect detection. *Pattern Recognit.* **2005**, *38*, 559–576. [[CrossRef](#)]
3. Bianconi, F.; Fernández, A. Evaluation of the effects of Gabor filter parameters on texture classification. *Pattern Recognit.* **2007**, *40*, 3325–3335. [[CrossRef](#)]
4. Cohen, F.S.; Fan, Z.; Attali, S. Automated inspection of textile fabrics using textural models. *IEEE Trans. Pattern Anal. Mach. Intell.* **1991**, *13*, 803–808. [[CrossRef](#)]
5. Xu, Y.C.; Meng, F.W.; Wang, L.Z.; Zhang, M.Y.; Wu, C.S.; Assoc Comp, M. Fabric Surface Defect Detection Based on GMRF Model. In Proceedings of the 2nd International Conference on Artificial Intelligence and Information Systems (ICAIS), Chongqing, China, 28–30 May 2021.
6. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
7. Lin, T.-Y.; Dollár, 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. 2117–2125.
8. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [[CrossRef](#)]
9. Karpiński, R.; Krakowski, P.; Jonak, J.; Machrowska, A.; Maciejewski, M.; Nogalski, A. Diagnostics of Articular Cartilage Damage Based on Generated Acoustic Signals Using ANN—Part I: Femoral-Tibial Joint. *Sensors* **2022**, *22*, 2176. [[CrossRef](#)]
10. Liao, Y.-T.; Lee, C.-H.; Chen, K.-S.; Chen, C.-P.; Pai, T.-W. Data Augmentation Based on Generative Adversarial Networks to Improve Stage Classification of Chronic Kidney Disease. *Appl. Sci.* **2021**, *12*, 352. [[CrossRef](#)]
11. Liu, S.; Yang, B.; Wang, Y.; Tian, J.; Yin, L.; Zheng, W. 2D/3D Multimode Medical Image Registration Based on Normalized Cross-Correlation. *Appl. Sci.* **2022**, *12*, 2828. [[CrossRef](#)]
12. Sun, M.; Lu, L.; Hameed, I.A.; Kulseng, C.P.S.; Gjesdal, K.-I. Detecting Small Anatomical Structures in 3D Knee MRI Segmentation by Fully Convolutional Networks. *Appl. Sci.* **2021**, *12*, 283. [[CrossRef](#)]
13. Zheng, W.; Tian, X.; Yang, B.; Liu, S.; Ding, Y.; Tian, J.; Yin, L. A Few Shot Classification Methods Based on Multiscale Relational Networks. *Appl. Sci.* **2022**, *12*, 4059. [[CrossRef](#)]
14. Guo, S.-S.; Lee, K.-H.; Chang, L.; Tseng, C.-D.; Sie, S.-J.; Lin, G.-Z.; Chen, J.-Y.; Yeh, Y.-H.; Huang, Y.-J.; Lee, T.-F. Development of an Automated Body Temperature Detection Platform for Face Recognition in Cattle with YOLO V3-Tiny Deep Learning and Infrared Thermal Imaging. *Appl. Sci.* **2022**, *12*, 4036. [[CrossRef](#)]
15. Zhou, Y.; Wen, S.; Wang, D.; Mu, J.; Richard, I. Object Detection in Autonomous Driving Scenarios Based on an Improved Faster-RCNN. *Appl. Sci.* **2021**, *11*, 11630. [[CrossRef](#)]
16. Dewi, C.; Chen, R.-C.; Jiang, X.; Yu, H. Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V4. *Multimed. Tools Appl.* **2022**, *1*–25. [[CrossRef](#)]
17. Zheng, W.; Zhou, Y.; Liu, S.; Tian, J.; Yang, B.; Yin, L. A Deep Fusion Matching Network Semantic Reasoning Model. *Appl. Sci.* **2022**, *12*, 3416. [[CrossRef](#)]
18. Bu, H.-G.; Wang, J.; Huang, X.-B. Fabric defect detection based on multiple fractal features and support vector data description. *Eng. Appl. Artif. Intell.* **2009**, *22*, 224–235. [[CrossRef](#)]

19. Huang, Y.; Jing, J.; Wang, Z. Fabric Defect Segmentation Method Based on Deep Learning. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–15. [[CrossRef](#)]
20. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788. [[CrossRef](#)]
21. Wei, B.; Hao, K.; Gao, L.; Tang, X.-S. Detecting textile micro-defects: A novel and efficient method based on visual gain mechanism. *Inf. Sci.* **2020**, *541*, 60–74. [[CrossRef](#)]
22. Jing, J.; Zhuo, D.; Zhang, H.; Liang, Y.; Zheng, M. Fabric defect detection using the improved YOLOv3 model. *J. Eng. Fibers Fabr.* **2020**, *15*. [[CrossRef](#)]
23. Wang, Y.; Hao, Z.; Zuo, F.; Pan, S. A Fabric Defect Detection System Based Improved YOLOv5 Detector. *J. Phys. Conf. Ser.* **2021**, *2010*, 012191. [[CrossRef](#)]
24. Dlamini, S.; Kao, C.-Y.; Su, S.-L.; Kuo, C.-F.J. Development of a real-time machine vision system for functional textile fabric defect detection using a deep YOLOv4 model. *Text. Res. J.* **2021**, *92*, 675–690. [[CrossRef](#)]
25. Kahraman, Y.; Durmuşoğlu, A. Classification of Defective Fabrics Using Capsule Networks. *Appl. Sci.* **2022**, *12*, 5285. [[CrossRef](#)]
26. Zheng, L.; Wang, X.; Wang, Q.; Wang, S.; Liu, X. A Fabric Defect Detection Method Based on Improved YOLOv5. In Proceedings of the 2021 7th International Conference on Computer and Communications (ICCC), Chengdu, China, 10–13 December 2021; pp. 620–624.
27. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
28. Ma, N.; Zhang, X.; Liu, M.; Sun, J. Activate or not: Learning customized activation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 8032–8042.
29. Cui, J.; Hou, X. Transmission line fault detection based on YOLOv4 with attention mechanism. *Foreign Electron. Meas. Technol.* **2021**, *40*, 24–29. (In Chinese) [[CrossRef](#)]
30. Moran, J.; Haibo, L.; Zhongbo, W.; Miao, H.; Zheng, C.; Bin, H. Improved YOLO V3 algorithm and its application in small target detection. *Acta Opt. Sin.* **2019**, *39*, 0715004. [[CrossRef](#)]
31. Xu, Q.; Lin, R.; Yue, H.; Huang, H.; Yang, Y.; Yao, Z. Research on Small Target Detection in Driving Scenarios Based on Improved Yolo Network. *IEEE Access* **2020**, *8*, 27574–27583. [[CrossRef](#)]
32. 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.
33. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
34. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
35. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
36. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 9627–9636.
37. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
38. Wang, K.; Liew, J.H.; Zou, Y.; Zhou, D.; Feng, J. Panet: Few-shot image semantic segmentation with prototype alignment. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 9197–9206.
39. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the Icm1, Haifa, Israel, 21–24 June 2010.
40. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
41. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.