



Article Fabric Surface Defect Detection Using SE-SSDNet

Hanqing Zhao¹ and Tuanshan Zhang^{2,*}

- ¹ Department of Materials Science and Engineering, Northwestern Polytechnical University, Xi'an 710072, China
- ² Mechanical and Electrical Engineering Department, Xi'an Polytechnic University, Xi'an 710048, China
- * Correspondence: zhangtuanshan@xpu.edu.cn

Abstract: For fabric defect detection, the crucial issue is that large defects can be detected but not small ones, and vice versa, and this symmetric contradiction cannot be solved by a single method, especially for colored fabrics. In this paper, we propose a method based on a combination of two networks, SE and SSD, namely the SE-SSD Net method. The model is based on the SSD network and adds the SE module for squeezing and the Excitation module after its convolution operation, which is used to increase the weight of the model for the feature channels containing defect information while re-preserving the original network to extract feature maps of different scales for detection. The global features are then subjected to the Excitation operation to obtain the weights of different channels, which are multiplied by the original features to form the final features so that the model can pay more attention to the channel features with a large amount of information. In this way, large-scale feature maps can be used to detect small defects, while small-scale feature maps are used to detect relatively large defects, thus solving the asymmetry problem in detection. The experimental results show that our proposed algorithm can detect six different defects in colored fabrics, which basically meets the practical needs.



1. Introduction

In the textile industry, the fabric production process involves many aspects, of which product quality control is the most critical step to ensure efficiency, the most important factor affecting product quality is the fabric surface defects, currently, the textile industry has defined more than 70 kinds of fabric defects [1–4].

Depending on the different texture backgrounds and defect shapes of fabrics, fabric defect detection algorithms are broadly classified into two categories: traditional algorithms and learning-based algorithms. Most of the traditional algorithms are based on feature engineering with prior knowledge, covering statistical, structural, spectral, and model-based methods. Learning-based algorithms can be further divided into classical machine learning algorithms and deep learning algorithms. Machine learning uses mathematical algorithms to learn and analyze data to make predictions and take decisions in the future, which has been widely employed in recent years and achieved stratifying results in various disciplines and industries [5]. Traditional machine learning algorithms, such as KNN and neural networks [6], are widely used for fabric defect detection probabilities, and feature engineering is one of the main processes of this approach. Traditional methods and conventional machine learning algorithms are not described in this paper.

The rest of this paper is arranged as follows. The second section introduces other researchers' related work; the third section introduces the details of our method; and the fourth section discusses the experimental setups and analyzes the results. Finally, we summarize our work in Section 5.



Citation: Zhao, H.; Zhang, T. Fabric Surface Defect Detection Using SE-SSDNet. *Symmetry* **2022**, *14*, 2373. https://doi.org/10.3390/ sym14112373

Academic Editors: João Ruivo Paulo, Cristina P. Santos and Gabriel Pires

Received: 8 October 2022 Accepted: 5 November 2022 Published: 10 November 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/).

2. Related Work—Deep Learning-Based Method for Fabric Defect

The application of deep learning-based methods for the automatic detection of fabric defects in the textile industry is generally divided into two steps. The first step is the extraction of the fabric defect area, which is usually captured by an industrial camera on a fabric inspection machine. The second step is defect image processing.

The research focus of deep learning models for fabric image defect detection mainly includes convolutional neural networks (CNN) [7] and recurrent neural networks (RNN) [8]. Convolutional neural networks have a strong nonlinear fitting ability, good robustness, and strong generalization ability. The experimental results show that the CNN method can accurately obtain the location and class of defects in the color fabric image dataset and improve the detection efficiency of defective images, but the detection speed still needs to be improved.

From the perspective of object detectors, deep learning-based detectors can be further divided into single-stage detectors and two-stage detectors, and the commonly used algorithms are shown in Table 1 [9].

One-Stage Detectors	Two-Stage Detectors
YOLO	Faster RCNN
SSD	Mask RCNN
YOLOv5/v6/v7	Cascade RCNN
RefineDet	FPN
RetinaNe	R-FCN

Table 1. Deep learning algorithms for detecting object.

One-stage detection algorithms do not have a separate suggestion generation stage. Typically, these algorithms consider all locations on the image as potential objects and try to categorize each region of interest as a target object or background. The recently proposed single multi-box detector (SSD) [10] is a typical single-stage detector that has obtained good detection performance in object detection. The algorithm is based on the de-signification of a convolutional neural network (CNN). Jin et al. [11] made some improvements to the fabric defect scenario and the experimental results showed reasonableness and effectiveness. this is one of the reasons why we chose this method.

Regarding the two-stage detector, a set of sparse proposals are generated in the first and second stages, and the features of the generated proposals are fed into the DCNN for predicting the results. As an excellent two-stage detector, Faster R-CNN is an object detection model that improves Fast R-CNN by utilizing a region proposal network (RPN) with a CNN model [12–14]. Jun et al. [15] proposed a framework that utilizes the Inception-V1 model and the LeNet-5 model, and the method includes local defect prediction in the first stage and overall defect identification in the second stage.

From the published literature, all the above methods have been applied. For example, to further improve the detection of fabric blemishes, Ouyang et al. [16] introduced a new paired-potential activation layer in CNN and used a combination of statistical blemish information and CNN for fabric blemish detection, which improved the detection accuracy of fabric blemishes with complex features and unbalanced data sets. AN et al. [17] proposed an improved Faster-CNN method using a deep residual network instead of traditional VGG-16 for feature extraction, which increased the feature pyramid module but also increased the number of anchor frames. For the problem of uneven distribution of defect samples and poor diversity of existing color fabric defect samples in the process of fabric image acquisition, K. Zhang et al. [18] proposed that the application of the method can accurately obtain the location and category of defects in the color fabric image data set, and improve the detection of defects. This method can accurately obtain the location and category of the blemishes, but the detection efficiency of the blemishes, but the detection speed still needs to be improved. In order to improve the detection speed, the regression-based YOLO series and SSD series algorithms [19]

have made great progress, and the experimental results show that the method can achieve high accuracy and meet the needs of real-time detection of color fabric defects. However, the detection performance of this method decreases when detecting fabrics with complex features. For the application scenario with limited computing resources, Zhu et al. [20] improved the structure of Dense Net and proposed a lightweight convolutional network that is more adaptable to the resource-constrained edge computing scenario, but the detection of defects is limited. For the problem of data imbalance that often exists in practical production, Wang et al. [21] proposed a lightweight deep learning model, which improved the DeeplabV3+ model, drawing on its advantages in multi-scale target detection, making the network structure lighter and also improving the detection speed, but it could not help with small defects. For the data labeling problem, Huang et al. [22] proposed an efficient convolutional neural network for defect segmentation and detection, which greatly reduces the expensive cost of manual labeling of the dataset, and in combination with non-defective samples, only a small number of defective samples are needed to learn the potential features of defects and to obtain high accuracy defect localization. However, this method of expanding samples also has shortcomings: first, it is difficult to generate high-resolution images and second, the generation process still requires a small number of real defect images. In recent years, the use of fabric structure parameters for defect detection, fabric structure parameters mainly including fabric density, weave pattern, color pattern, etc., can be pre-set for these parameters raw, so that it can reduce the reliance on network parameters in defect detection, such as the transformation of Local Binary Patterns(LBP) [23,24] for face recognition. It can also provide some new ideas for other identification problems in the textile industry, such as fabric defect detection, fabric appearance analysis and fabric inversion modeling, but the identification of small defects is not very good. In summary, the above methods can only solve the problem of partial localization.

Compared with the detection algorithms mentioned above, SENet [25], which stands for Squeeze-and-Excitation Networks, got recognized for its effectiveness by winning the ImageNet 2017 classification competition, and its proposed SE module is simple in idea and easy to implement. SENet mainly learns channel correlations and filters out channel-specific attention, which slightly increases the computational effort, but with better results. SENet is not a complete network, but a substructure that can be embedded in other classification or detection models. The core idea of SENet is to train the model by learning the feature weights according to the loss so that the effective feature map has more weight and the ineffective or small feature map has less weight to achieve better results.

In this article, we propose a SE-SSD Net deep learning fabric defect algorithm, which can detect six types of common defects in colored fabrics. The model is based on a Single Shot Multibox Detector network [26]. The SE module is added after convolution operation, which is the Squeeze and Excitation module, improving the weights of the feature channels with defect information, retaining the original network, and extracting feature maps of different scales for detection at the same time. Large-scale feature maps can be used to detect larger ones, this is an application of the concept of symmetry in detection. Finally, based on the practical production in the factory, defect samples were collected to form fabric defect data sets. The samples with different defects are shown in Figures 1 and 2.

The experimental results show that our proposed algorithm can detect various types of fabrics with various defects.

The main contributions of this paper are as follows:

- We propose a novel deep learning algorithm SE-SSD Net which can be used for fabric defect detection.
- (2) Compared with traditional methods, our proposed method can detect six different types of defects in colored fabrics.
- (3) Our proposed method can adapt to the textured background of fabrics of varying complexity.

(4) The method overcomes the interference of image noise and improves the efficiency of detection. In addition, to better suit the industrialization needs, we conducted experimental validation on the textile defect dataset from Xi'an Polytechnic University and our own dataset, and the results proved that the method is the state-of-the-art method to solve such problems.



Figure 1. Simple defect sample.



Figure 2. Complex defect samples.

3. Proposed Approach

3.1. Overview

This section presents the SE-SSDNet-based method for fabric surface defect detection. Figure 3 shows the detection process of the proposed method, which mainly includes image resizing, data enhancement, model training, and testing.





3.2. Data Augmentation

To test the prediction accuracy of our proposed model, we evaluated our model using the textile defect detection database of Xi'an Polytechnic University and the fabric defect dataset collected by the authors, respectively. Due to the small sample size of the fabric defect dataset collected by the authors, the model is prone to underfitting during the training process, which leads to large errors in the determination of surface defects. Therefore, this dataset requires data enhancement processing [27].

Data augmentation allows limited data to produce more data, increasing the number and diversity of training samples, and thus improving the robustness of the model. Randomly changing the training samples can reduce the dependence of the model on certain attributes, thus improving the generalization ability of the model. The main approach is to apply a series of methods to randomly change the training data, such as panning, rotation, cropping, and background color modification. Each enhanced image can be considered as a new image, so we can continuously provide new training samples for the model, thus enabling the model to learn more discriminative and generalization features. Data augmentation techniques can improve the accuracy of the model and reduce overfitting. In addition, data augmentation can increase the amount of data. When the real training sample cannot be increased, data augmentation is a very effective way to overcome this problem. This process consists of two steps, the first step is image acquisition and the second step is image data expansion, as shown in Figure 4.





3.3. Network Architecture

3.3.1. SE Module

The SE module models the dependencies among each channel of the feature map obtained after the convolution operation, to assign the weights of the channels and introduce the attention mechanism into the network. Its structure is shown in Figure 5, where Uis the output of any convolutional layer and \tilde{x} is the output after merging the weights. Embedding the SE module after the convolutional layers facilitates capturing the spatial correlation between features. After the channel weights are assigned, the output has feature maps of the same size and same dimensionality.



Figure 5. SE module structure.

The SE module consists of three parts. First is the squeeze operation, which is used to describe global information. Next is the Excitation operation for obtaining channel weights. Finally, the fusion operation is used for weight assignment.

The squeezing operation of the SE module is to compress the features in the spatial dimension to keep the input dimension and the output dimension consistent. The information of all points on each output channel is averaged into a value that has a global receptive field representing the corresponding global distribution on the feature channel. It also enables layers close to the input to obtain global information as well. This operation uses a global pool that compresses the input features with the size of $C \times H \times W$ to the feature descriptions of $1 \times 1 \times C$ The calculation formula is:

$$Z_{c} = F_{sq}(\mu_{c}) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mu_{c}(i,j)$$
(1)

where μ_c represents the *C* channel in *U*.

After the Squeeze operation, the global features of the channel are obtained. Then the global features are operated with Excitation operation to learn the relationship among each channel, and the weight of each channel is generated by using the parameter W. In order to reduce the model complexity and improve the generalization ability, this operation uses two full connection layers to complete. In order to reduce the amount of calculation, the first full connection layer compresses C channels into C/r channels, followed by the ReLU activation functions. The second full connection layer is restored to C channels with the Sigmoid activation function at the end. The calculation formula is (2):

$$s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2\delta(W_1z))$$
(2)

where *r* refers to the scaling ratio, the default value is 16. W_1, W_2 are two fully connected layers, $W_1 \in R^{\frac{c}{r} \times c}$, $W_2 \in R^{\frac{c}{r} \times c}$.

 δ is ReLU function, σ is the gate function, here refers to the sigmoid function. The *s* obtained in the second step has the feature map information of each channel, and each number in *s* represents the weight of each channel, respectively. Finally, the original feature layer U is combined with the weights obtained. The channel weights are allocated according to the channel multiplication and then weighted. The calculation formula is (3):

$$\widetilde{X}_c = F_{scale}(\mu_c, s_c) = s_c * \mu_c \tag{3}$$

3.3.2. Detection Network Architecture

Our proposed fabric defect detection network is the modification of the SSD network [28]. In this network, the SE module is added to its convolutional layers to improve the network attention. The basic model is VGG. It consists of 16 convolution layers, and the sizes of the convolution cores on each convolutional layer are the same. FC6 and FC7 are converted into 3×3 convolution layer conv6 and 1×1 convolution layer conv7, FC8 layer, and all dropout layers are removed, and then SE is added on this basis SE_0, SE_1, SE_2, conv_8, SE_3, conv_9, SE_4, conv_10, SE_5, conv_11 layer. Take part of the layer output as the effective feature layer. The network structure is shown in Figure 6:



SE Module

Figure 6. Network structure.

During the training process, the default box that matches the ground truth in the image is determined first, and the bounding box corresponding to the default box will be responsible for predicting defects. Suppose there are m feature layers. On each feature layer, the generation formula of the default box is as follows:

$$S_k = S_{\min} + \frac{S_{\max} - S_{\min}}{m - 1} (k - 1), k \in [1, m]$$
(4)

where S_k indicates the ratio of the prior frame size to the picture, S_{min} and S_{max} indicates the minimum and maximum values of the ratio, respectively.

A large number of default boxes will be generated on the same feature map. NMS (no maximum suppression) algorithm [29] is applied to filter the generated default boxes, and the optimal detection frame is obtained after removing unnecessary default boxes. When selecting default boxes, confidence and IOU (Intersection over Union) [30] are the two indicators that are considered. Confidence refers to the matching degree of the target in the default box, so as to score each default box. IOU is the overlap rate between the default box and the ground truth box generated by the algorithm. The default box with the largest IOU is taken as the training object so that each label box corresponds to a default box. For the remaining unmatched default boxes, select the IOU of the labeled box greater than 0.5 as the negative sample, and the IOU of the labeled box less than 0.5 as the positive sample. In order to balance the sizes of positive and negative samples, the negative samples are arranged in descending order of confidence error, and the larger error is selected as the negative training sample. Finally, the ratio of positive and negative samples is close to 1:3.

The loss function of the network is defined as the weighted sum of two parts, including location error (Localization loss) and confidence error (Confidence Loss). It is defined as follows:

$$L(x,c,l,g) = \frac{1}{N}(L_{conf}(x,c) + \alpha L_{loc}(x,l,g))$$
(5)

where *N* is the number of positive samples in the default box, α is the weight between positioning and classification loss, and the default value is set to 1.

For the position error, the loss is the loss of only positive samples, which is defined as the preselected box and the bounding box SmoothL1. The parameters of the loss function regression are the center offset of the default box (cx, cy) and the height h and width w of the box. The calculation formula is as follows:

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & if|x| < 1\\ |x| - 0.5 & otherwise \end{cases}$$
(6)

$$L_{loc}(x,l,g) = \sum_{i \in Posm \in \{cx,cy,w,h\}}^{N} \sum x_{ij}^{k} smooth_{L1}(l_i^m - \widetilde{g}_j^m)$$
(7)

where $\tilde{g}_{j}^{cx} = (g_{j}^{cx} - d_{i}^{cx})/d_{i}^{w}, \tilde{g}_{j}^{cy} = (g_{j}^{cy} - d_{i}^{cy})/d_{i}^{h}, \tilde{g}_{j}^{w} = log\left(\frac{g_{j}^{w}}{d_{i}^{w}}\right), \tilde{g}_{j}^{w} = log\left(\frac{g_{j}^{w}}{d_{i}^{w}}\right).$

For confidence, the classification loss includes positive samples and negative samples. Using multi-class SoftMax cross-entropy loss [31], the calculation formula is as (8):

$$L_{conf}(x,c) = -\sum_{i\in Pose}^{N} x_{ij}^{p} log(\tilde{c}_{i}^{p}) - \sum_{i\in Neg} log(\tilde{c}_{i}^{0})$$
(8)

where $\widetilde{c}_i^p = \frac{exp(c_i^p)}{\sum_p exp(c_i^p)}$.

In the prediction phase, the category and confidence value of each predictor box are first determined based on the category confidence, and then the predictor boxes that belong to the background are filtered out. The prediction boxes with confidence thresholds lower than 0.5 are filtered out, the retained prediction boxes are decoded and their true position parameters are obtained based on the default boxes. Meanwhile, the position of the prediction box should not exceed the picture. After decoding, the prediction boxes are sorted in descending order according to the confidence level. Finally, Top-k prediction boxes [32] are kept. Based on this, the non-maximum suppression (NMS) algorithm is used to filter out the prediction frames with a large amount of overlap, and the retained prediction frames are the detection results.

4. Experiments and Discussion

In this section, we choose two datasets for validation, namely, a textile defect detection database from Xi'an Polytechnic University and a defect dataset collected by the authors. Our own dataset is from a textile company in China. It contains six types of defects commonly found in color fabrics, each of which is classified into simple and complex textures. These defects are named Podong, Maobian, Huangze, Duanjingwei, and Dimo according to the cause and appearance shape, as shown in Figure 7. In the following, the experimental setup is first described in detail, and then the experiments were performed in two directions:

- (1) The proposed detection model is applied to two fabric datasets;
- (2) The proposed defect detection model is compared with the other three representative detection models.



Figure 7. (**a**–**f**) shows six different defect samples. (1–6) shows simple texture samples. (7–12) shows complex texture samples.

4.1. Experimental Setup

To validate the performance of our proposed fabric defect detection model, we conducted experiments using Pytorch on Ubuntu 18.04 operating system. The system is powered by an Intel Core i9-9900k processor with an NVIDIA GeForce GTX2080Ti (11 GiB) graphics card. The dataset used consists of 2000 original images divided into a training set, a validation set, and a test set. The number of samples in each sub-dataset is 1400, 220, and 380, respectively. Accuracy (ACC) and loss curve (Loss) are applied to evaluate the performance of the detection model. ACC is the most common evaluation metric. In general, the higher the ACC, the better the performance of the model. The loss curve can be demonstrated that the proposed loss function converges to 0 as the learning process proceeds. The accuracy is defined as Equation (9).

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$
(9)

In this equation, TP represents the number of samples correctly classified as positive; FP represents the number of examples incorrectly classified as positive; TN represents the number of examples correctly classified as negative; and FN represents the number of examples incorrectly classified as negative. We plot the data output from the learning process as shown in Figures 8 and 9. From the figure, we can see that the ACC of the method proposed in this paper is about 81%, while the ACC of S_MobileNet, S_EfficientNet, and SSD is about 79%, 78%, and 78%, respectively. According to the loss curves, the loss values of the whole model converge steadily during the learning process. These evaluation metrics prove that the method proposed in this paper can improve the accuracy of fabric defect detection and that detection performance is relatively stable.



Figure 8. Accuracy curve.



Figure 9. Loss curve of the proposed method.

During the training process, the network structure was implemented by Pytorch. The learning rate was set to 0.0003. Considering the image size and GPU memory limitations, we set only 32 images for each iteration, in other words, the batch size was 32.

4.2. Result Analysis

According to the method proposed in this paper, some typical test results are shown in Figures 10 and 11. The proposed detection model can successfully detect defects with different shapes, sizes, and positions.



Figure 10. Simple textile defect detection results.



Figure 11. Complex textile defect detection results.

Figures 10 and 11 respectively show the detection results of defects on fabrics with simple texture and complex textures. It can be seen that even tiny defects (such as those in Figure 8) can be accurately identified by the model. In addition, the model also showed excellent adaptability to the brightness variation under different light conditions (in Figure 8). Furthermore, this result confirms the robustness of the proposed defect detection model to texture features and illumination changes.

4.3. Method Comparison

In order to further verify the effectiveness of the proposed model in fabric defect detection, its detection performance is compared with three other representative defect detection methods. They are S_MobileNet, S_EfficientNet, and SSD, which are currently the most representative methods based on learning models. Therefore, this paper presents a comparative model for the detection of defects. Evaluation is based on real fabric samples in the same environment. The result is shown in Table 2, which summarizes the performance of the four inspection models for different defect types, and it can be seen from Figure 12 that the performance of our inspection model is better than the other three models.

Table 2. Model performance comparison.

	PoDong	Mobian	Hze	Djwei	Cadong	Dimo
Ours	82.28%	78.01%	90.54%	61.75%	90.49%	86.99%
S_Mobilenet	78.48%	82.77%	88.22%	49.05%	87.31%	86.83%
S_Efficientnet	76.39%	78.21%	85.26%	51.75%	86.70%	89.35%
SSD	81.10%	73.72%	88.11%	55.03%	88.54%	83.88%



Figure 12. Comparison of methods.

4.4. Discussion

With the above elaboration, the effectiveness of SE-SSD is verified, but to be deployed in the industry, there are issues of generalization, robustness, and real time to be considered. The following items are worth discussing, and they are also the focus of our next R&D.

- (1) For point defects of fabrics, sometimes there are misjudgments. For such defects, we adopt the fabric surface defect detection algorithm based on the GAN network [33] and Hough transform, by introducing the generative GAN network to simulate defect samples, performing random Hough transform on the obtained defect samples. Then, plotting the defect location in the original fabric image and returning the coordinate value of the defect location greatly improves the detection efficiency.
- (2) For the probability problem of texture affecting defect misclassification, the traditional fabric surface defect detection algorithm based on L0 gradient minimization smoothing [34] and K-means clustering can be used. The core idea is to improve the contrast between the defect region and the background region, use the image smoothing method to remove the image texture and retain the larger edges in the image, then treat the defect region and the background region as two different classes of pixel points. Finally, use the K-means clustering method to segment the defect region and the background region as the defect region and the background region as the defect region and the background region.
- (3) For real-time consideration, the SSD model can be compressed by reducing the number of channels and setting a priori frame values that are more in line with the data according to the characteristics of cloth defects, especially the loss function can be optimized, and different model frames can be used for different textures of cloth. As the training time of SE-SSD is too long, the data set can be divided according to the proportion, 90% of the data set is divided into the training set and 10% of the data set is

divided into the test set. Specific data enhancement means are applied to the training set to simulate various lighting conditions through color space transformation for data augmentation to improve the generalization ability and real-time performance of the model. The next step focuses on the effect of cloth parameters on the model, thus improving the real-time performance in terms of model prediction.

5. Conclusions and Future Work

5.1. Conclusions

In this paper, we propose an end-to-end fabric defect detection model by combining deep learning techniques, which is the first adaptive and multi-scale detection of fabric defects by combining the SE module and SSD network. The fabric detection model obtained by training a certain amount of data can accurately detect fabric surface defects for fabric surface quality problems of different sizes, textures, and lighting conditions. The method proposed in this chapter first performs a Squeeze operation on the convolved feature map of the SSD network to obtain global features. Then, the global features are subjected to an Excitation operation to obtain the weights of different channels and then multiply them with the original features to form the final features. According to the different weight values, the model pays more attention to the channel features with high information content.

The innovative aspects of this paper are as follows:

We propose a novel deep learning algorithm SE-SSDNet that can be used for fabric defect detection. It is particularly suitable for fabric defects with different textures and shapes as well as colored fabrics.

In this paper, by introducing the squeeze and Excitation modules into the SSD detection network, the network attention mechanism is enhanced, and the detection efficiency and self-adaptability are improved.

The experimental results are compared with those on Efficient Net and MobileNetV3, and the results demonstrate that our method greatly improves the accuracy and detection efficiency of fabric defect detection. The experimental results show that our model can detect defects adaptively in simple and complex texture backgrounds with an average accuracy of 81.7%. Through comparison experiments, the average accuracy of our model outperforms the other three models, effectively improving the efficiency of fabric detection.

5.2. Future Work

During the experiment, there is basically no problem for monochrome fabrics, but there are still some defects that cannot be detected for brightly colored color fabrics, such as Djwei, and the accuracy of the model cannot meet the actual detection requirements, in addition, real-time is also the key to detecting surface defects on fabrics. We are currently optimizing and adjusting with textile enterprises, in the future, we plan to make appropriate modifications to the detection network and increase the fabric parameters, so that the model has a higher detection rate for that various types of defects, through a comprehensive comparison and statistics of the real-time detection of textile enterprises, the deployment of the model to the automatic detection system is the direction of our future work and key tasks.

Author Contributions: Conceptualization, T.Z.; methodology, T.Z.; software, H.Z.; data curation, H.Z.; writing—original draft preparation, H.Z.; writing—review and editing, T.Z.; visualization, H.Z.; supervision, T.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received Major Projects of Guangzhou City of China funding (201907010012).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- 1. Hanbay, K.; Talu, M.F.; Özgüven, Ö.F. Fabric defect detection systems and methods—A systematic literature review. *Optik* **2016**, 127, 11960–11973. [CrossRef]
- 2. Bullon, J.; Arrieta, A.G.; Encinas, A.H.; Dios, A.Q. Manufacturing processes in the textile industry. Expert Systems for fabrics production. *Adv. Distrib. Comput. Artificial Intell. J.* 2017, *6*, 15–23.
- 3. Singh, K.; Kaleka, J.; Kaleka, J. Identification and classification of fabric defects. Int. J. Adv. Res. 2016, 4, 1137–1141. [CrossRef]
- 4. Song, L.; Li, R.; Chen, S. Fabric defect detection based on membership degree of regions. *IEEE Access* 2020, 99, 1. [CrossRef]
- 5. Li, Y.; Zhang, D.; Lee, D.-J. Automatic fabric defect detection with a wide-and-compact network. *Neurocomputing* **2019**, *329*, 329–338. [CrossRef]
- 6. Patil; Minal; Verma, S.; de Wako, J. A review on fabric defect detection techniques. Int. Res. J. Eng. 2017, 4, 131–136.
- Zhao, Y.; Hao, K.; He, H.; Tang, X.; Wei, B. A visual long-short-term memory based integrated CNN model for fabric defect image classification. *Neurocomputing* 2019, 380, 259–270. [CrossRef]
- Peng, P.; Wang, Y.; Hao, C.; Zhu, Z.; Liu, T.; Zhou, W. Automatic Fabric Defect Detection Method Using PRAN-Net. *Appl. Sci.* 2020, 10, 8434. [CrossRef]
- 9. Li, C.; Li, J.; Li, Y.; He, L.; Fu, X.; Chen, J. Fabric Defect Detection in Textile Manufacturing: A Survey of the State of the Art. *Secur. Commun. Netw.* **2021**, 2021, 13. [CrossRef]
- 10. Liu, Z.; Liu, S.; Li, C. Fabric defects detection based on SSD. In Proceedings of the 2nd International Conference on Graphics and Signal Processing: ICGSP'18, Sydney, Australia, 6–8 October 2018.
- 11. Jin, L.; Liu, G. An Approach on Image Processing of Deep Learning Based on Improved SSD. Symmetry 2021, 13, 495. [CrossRef]
- 12. Zhou, H.; Jang, B.; Chen, Y.; Troendle, D. Exploring faster RCNN for fabric defect detection. In Proceedings of the 2020 Third International Conference on Artificial Intelligence for Industries (AI4I), Irvine, CA, USA, 21–23 September 2020.
- Zhao, Z.; Gui, K.; Wang, P. Fabric defect detection based on cascade faster R-CNN. In Proceedings of the 4th Inter-national Conference on Computer Science and Application Engineering, Sanya, China, 20–22 October 2020.
- 14. Wu, J.; Le, J.; Xiao, Z.; Zhang, F.; Geng, L.; Liu, Y.; Wang, W. Automatic fabric defect detection using a wide-and-light network. *Appl. Intell.* **2021**, *51*, 4945–4961.
- 15. Jun, X.; Wang, J.; Zhou, J.; Meng, S.; Pan, R.; Gao, W. Fabric defect detection based on a deep convolutional neural network using a two-stage strategy. *Text. Res. J.* **2020**, *91*, 130–142. [CrossRef]
- 16. Ouyang, W.; Xu, B.; Hou, J.; Yuan, X. Fabric defect detection using activation layer embedded convolutional neural network. *IEEE Access* **2019**, *7*, 70130–70140. [CrossRef]
- 17. An, M.; Wang, S.; Zheng, L.; Liu, X. Fabric defect detection using deep learning: An improved faster r-approach. In Proceedings of the International Conference on Computer Vision, Image and Deep Learning, Chongqing, China, 10–12 July 2020; pp. 319–324.
- Zhang, K.; Yan, Y.; Li, P.; Jing, J.; Liu, X.; Wang, Z. Fabric Defect Detection Using Salience Metric for Color Dissimilarity and Positional Aggregation. *IEEE Access* 2018, 6, 49170–49181. [CrossRef]
- 19. 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.
- 20. Zhu, Z.; Han, G.; Jia, G.; Shu, L. Modified dense Net for automatic fabric defect detection with edge computing for minimizing latency. *IEEE Internet Things J.* 2020, *7*, 9623–9636. [CrossRef]
- Wang, Z.; Jing, J.F. Pixel-wise fabric defect detection by CNNs without labeled training data. *IEEE Access* 2020, *8*, 161317–161325. [CrossRef]
- 22. Huang, Y.Q.; Jing, J.F.; Wang, Z. Fabric defect segmentation method based on deep learning. *IEEE Trans. Instrum. Meas.* 2021, 70, 1–15. [CrossRef]
- Pourkaramdel, Z.; Fekri-Ershad, S.; Nanni, L. Fabric defect detection based on completed local quartet patterns and majority decision algorithm. *Expert Syst. Appl.* 2022, 198, 116827. [CrossRef]
- 24. Fekri-Ershad, Shervan, Tajeripour, Multi-Resolution and Noise-Resistant Surface Defect Detection Approach Using New Version of Local Binary Patterns. *Appl. Artif. Intell.* 2017, *31*, 395–410. [CrossRef]
- 25. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 2020, 42, 2011–2023. [CrossRef]
- Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector, Computer Vision—ECCV 2016 Lecture Notes in Computer Scienc; Springer International Publishing AG: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
- Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
- 28. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* 2014, 1409–1556. [CrossRef]
- 29. Chu, J.; Zhang, Y.; Li, S.; Leng, L.; Miao, J. Syncretic-NMS: A Merging Non-Maximum Suppression Algorithm for Instance Segmentation. *IEEE Access* 2020, *8*, 114705–114714. [CrossRef]

- Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666.
- Li, X.; Wang, W.; Hu, X.; Li, J.; Tang, J.; Yang, J. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 11632–11641.
- Nemoto, K.; Hamaguchi, R.; Imaizumi, T.; Hikosaka, S. Classification of rare building change using CNN with multi-class focal loss. In Proceedings of the 2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 4663–4666.
- Liu, J.; Wang, C.; Su, H.; Du, B.; Tao, D. Multistage GAN for Fabric Defect Detection. *IEEE Trans. Image Process.* 2020, 29, 3388–3400.
 [CrossRef] [PubMed]
- 34. Zhang, H.; Ma, J.; Jing, J.; Li, P. Fabric Defect Detection Using L0 Gradient Minimization and Fuzzy C-Means. *Appl. Sci.* **2019**, *9*, 3506. [CrossRef]