

Received August 9, 2020, accepted August 30, 2020, date of publication September 2, 2020, date of current version September 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3021189

Pixel-Wise Fabric Defect Detection by CNNs Without Labeled Training Data

ZHEN WANG¹ AND JUNFENG JING¹

School of Electronics and Information, Xi'an Polytechnic University, Xi'an 710048, China

Corresponding author: Junfeng Jing (jingjunfeng0718@sina.com)

This work was supported in part by the Youth Innovation Team of Shaanxi Universities, and in part by the Graduate Scientific Innovation Fund for Xi'an Polytechnic University under Grant chx2020014.

ABSTRACT Surface inspection is a necessary process of fabric quality control. However, it remains a challenging task owing to diverse types of defects, various patterns of fabric texture, and application requirements for detection speed. In this article, a lightweight deep learning model is therefore proposed to complete the segmentation of fabric defects. The input of the model is a fabric image, and the output is a binary image. Generally known, a deep learning model usually needs much data to update the parameters. Still, as an abnormal phenomenon, fabric defects are unpredictable, which makes it impossible to collect a large number of data. Distinct from other models, the proposed method is a supervised network but does not need manually labeled samples for training. A fake sample generator is designed to simulate the defect image, which only needs the defect-free fabric image. The proposed model is trained with fake samples and verified with real samples. The experimental results show that the model trained with false data is useful and achieves high segmentation accuracy on real fabric samples. Besides, a loss function is proposed to deal with the problem of imbalance between the number of background pixels and the number of defective pixels in the fabric image. Comprehensive experiments were performed on representative fabric samples to verify the segmentation accuracy and detection speed of this method.

INDEX TERMS Fabric defect, deep learning, image segmentation, defect detection, imbalanced dataset.

I. INTRODUCTION

Weaving clothes is a great leap in the history of human evolution. The textile industry is as old as human civilization. Fabric is closely related to human life. It is not only the key material of clothing but also applied to many industrial products [1]. Since the first industrial revolution, the weaving of fabrics has mainly depended on machines. At present, the process is an automated operation without any intervention. However, due to the influence of fiber quality and other factors, fabric defects are inevitable. Fabric defects reduce the quality of products and affect the profits of enterprises. So, the last step in weaving is the detection of fabric surface defects. Unfortunately, the inspection process is still heavily dependent on experienced workers. There are many drawbacks in manual inspection methods, such as fatigue of the human eye when working long hours, and increasing labor costs year by year. In order to improve the effectiveness and

efficiency of fabric inspection, it is necessary to realize the automation of fabric inspection in order to save labor costs and achieve higher accuracy and efficiency [2].

In recent years, the machine vision-based method has been widely used in fabric defect detection. These methods are mainly divided into two categories, traditional methods based on image processing and learning methods based on convolutional neural networks [3]. Traditional methods usually deal with a single image and use hand-designed features to detect defects, such as filters, texture, and color features. These methods often need to set different parameters for different textures and defects [4]. With the abundance of computing resources and the explosion of data, the methods based on deep learning are gradually applied to defect detection. These methods use convolution to extract features automatically through learning, which reduces the steps of manual feature extraction, but this process needs much data [5]. However, as an abnormal phenomenon, the occurrence of defects is unpredictable, and it is almost impossible to collect a large number of samples. Besides, in practical

The associate editor coordinating the review of this manuscript and approving it for publication was Zheng Liu¹.

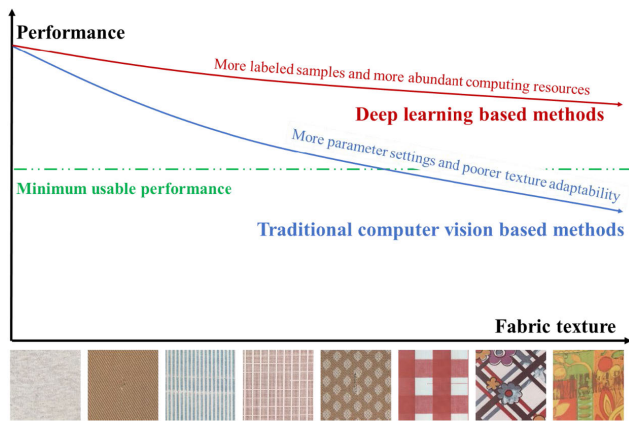


FIGURE 1. A comparison of deep learning-based methods and machine vision-based methods.

application, the detection is online, which requires high real-time performance, so it is necessary to improve the detection speed of the deep learning model. A comparison between the depth learning-based approach and the machine vision-based approach is shown in Fig. 1. This article focuses on using deep learning methods to address the detection of fabric surface defects. The focus is to design an efficient and lightweight convolutional neural network detection framework, but the training process of the model does not need the manually labeled samples.

The fabric has a repetitive texture, and the defect of the fabric is the destruction of this regular texture. A defective fabric image can be seen as a superposition of textures and defects [6]. Inspired by this, we propose a formula to describe the defect image, as shown in Fig. 2. In the figure, Mask is the expected detect result and also the part that needs to be manually labeled in deep learning. We found that both Mask and Defect can be automatically generated by rules to simulate real defect samples. This method only requires non-defective samples, which is easy to obtain.

In recent years, many image segmentation network models have been proposed, and these models have achieved amazing results in general image segmentation [7]–[9]. However, these models are not always applicable to the defect segmentation task. For the defect segmentation task, there is no fixed pattern of defects and fabric textures, and the efficiency of the network must be considered. Therefore, we improve the DeeplabV3+ model [10], learn from its advantages for multi-scale target detection, and make the network lighter to improve the detection speed.

In brief, our significant technical contributions are the following:

- (1) A defect detection framework based on deep learning is constructed, which trains the model through false data sets and does not need to label data sets.
- (2) A loss function is proposed to solve the problem of imbalance between the number of background pixels and the number of defect pixels in the fabric image

(3) Compared with existing methods, The proposed model has fewer model parameters and can significantly shorten detection time. It is also more suitable for online automated detection.

In the following, we first review related work in Section II, then overview the pipeline of our method in Section III. Section IV reports and discusses our experimental results. Finally, Section V provides a summary of this work and our concluding remarks.

II. RELATED WORK

The most common fabric defect detection methods based on traditional image processing can be divided into four categories: model-based method, frequency-domain method, and statistical method. A novel automatic detection method is presented based on frequency domain filtering and similarity measurement, yet this model cannot be used to segment defect at pixel level [11]. Li *et al.* [12] proposed a fabric defect detection method based on saliency features, which achieve the segmentation of fabric defects for a variety of textures, but the average of a single image is 397 milliseconds, which is not suitable for real-time applications. Zhang *et al.* [13] proposed a defect segmentation method based on texture elimination and image clustering, which has excellent results for plain, twill, pattern, and other textured fabrics. However, many parameters need to be set manually. Li *et al.* [14] presented a yarn-dyed fabric detection method. In this method, only one Gabor filter is applied, and its parameters are determined automatically by using random drift particle swarm optimization (RDPSO) algorithm. It can segment small texture accurately, but it is challenging to apply to large texture fabric. Besides, fabric defect detection methods using infrared imaging have also been applied [15]. In summary, traditional image processing methods usually rely on manual setting of parameters, and it is challenging to meet the requirements of real-time detection.

In recent years, with the improvement of computer performance and information explosion, deep learning (DL)-based method have become more and more popular in fabric defect detection. The DL-based method does not need to extract features manually; it automatically extracts and recognizes features from images. This automatic process makes the deep learning model have high accuracy for computer vision applications such as defect detection. The application of deep learning in fabric defect detection can be divided into three categories [16]: defect classification, defect location, and defect segmentation. Jing *et al.* [17] proposed an improved method of fabric defect classification based on the AlexNet network, which achieved the defect classification of yarn-dyed fabric. A compact network is proposed for the defect classification of knitted fabrics, which performs well in detection accuracy with a smaller model size [18]. A YOLO model-based fabric defect location method is proposed to improve the speed of defect detection. Ouyang *et al.* [19] proposed a fabric defect segmentation method based on convolution neural network embedded in an active layer. In order to detect

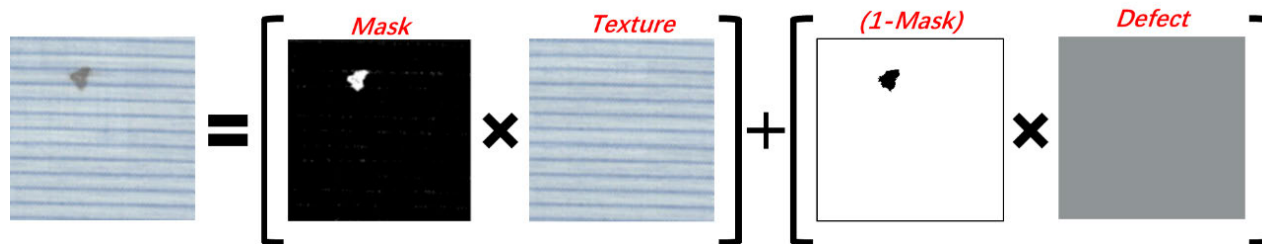


FIGURE 2. A proposed fabric defect detection formula.

pattern fabric defects, a hybrid method of traditional image processing and deep learning is proposed [20], which can achieve accurate detection of common defects in yarn-dyed fabric, such as holes, carrying, knots. Although the above methods use deep learning to extract features and achieve excellent detection performance automatically, they are all supervised learning methods, which need to collect, clean, and label training data sets.

Many unsupervised methods are also applied to defect detection. Mei *et al.* proposed a multi-scale convolution denoising network for fabric segmentation, but the performance of complex texture fabric detection needs to be further improved [16]. Li *et al.* proposed a method of pattern fabric defect detection. Even if the negative sample is not enough, it can obtain satisfactory detection accuracy [2]. However, it is challenging to integrate this method into the automatic defect detection system because it cannot be detected in real-time. Besides, Table 1 shows the number of training set images required by some unsupervised methods. Although these methods do not require annotated data, they require more than thousands of unlabeled data.

TABLE 1. The number of training set images required by some unsupervised methods.

Year	Methods	Number of images in the training set
2018	AffinityNet[23]	10,582
2018	Double-DIP[6]	200
2018	MSCDAE[16]	2000
2019	Unsupervised segmentation and ELM[24]	3500
2020	Multistage GAN[25]	12957

Large and well-annotated datasets, such as ImageNet, COCO, and Pascal VOC, are considered to be the key to promote computer vision research [21]. However, it is costly to create such a dataset. Another option is to use simulation data for model training. For example, in autopilot, the image recognition model is trained using a simulation environment [10]. For the deep reinforcement learning of robot tasks, the model needs to be trained in the false synthesis domain, because the training in the real environment may be very expensive [22]. For defect detection tasks, many

methods use GAN networks to generate false data to expand the number of samples. However, these methods have two disadvantages. One is that it is challenging to generate high-resolution images. The other is that the generation process still requires a small number of real defect images.

III. METHODOLOGY

The flow of the method is shown in Fig. 3. First, we use the generator of rule constraints to generate false data sets, and then we build a lightweight model, which includes encoder and decoder. In order to adapt to different sizes of fabric defects, the encoder part includes a pyramid convolution module. The input of the model is a gray image, and the output is the defect segmentation result. In the training stage, the false data set is used to train the model. The test stages use real data sets for testing.

A. GENERATE FAKE DATA

Different weaving methods, yarn materials, and even weaving machines cause various kinds of defects. So, there are many standards about the types of fabric defects. However, the shape of defects mainly includes two types, as shown in Fig. 4, one is point defects such as Knots, Holes, and Oil Spot, the other is strip defects caused by abnormal yarn, such as Overshot, End Out and Jerk-in, etc. [26].

As shown in Fig. 5, the fabric defect image can be regarded as the superposition of background texture and defect. *Mask* represents defect segmentation image; texture represents a defect-free image, *Defect* represents defect part texture.

We construct a fake data generator to construct the training set. *Mask* and *Defect* are generated according to rules. According to the typical shapes of fabric defects, there are two kinds of *Mask*: round and rectangle, and three kinds of *Defect*: rotation, alteration, and shadow, as shown in Fig. 6.

B. MODEL ARCHITECTURE

Fabric defect segmentation can be regarded as a binary image segmentation task, where “0” and “1” refer to “defect-free” and “defect”, respectively. A deep learning-based fabric defect segmentation model requires fusion of low-level features and high-level features. Low-level features are minor details of the image, like lines or dots. High-level features are built on top of low-level features to detect more significant defects in the image. The proposed detection model is

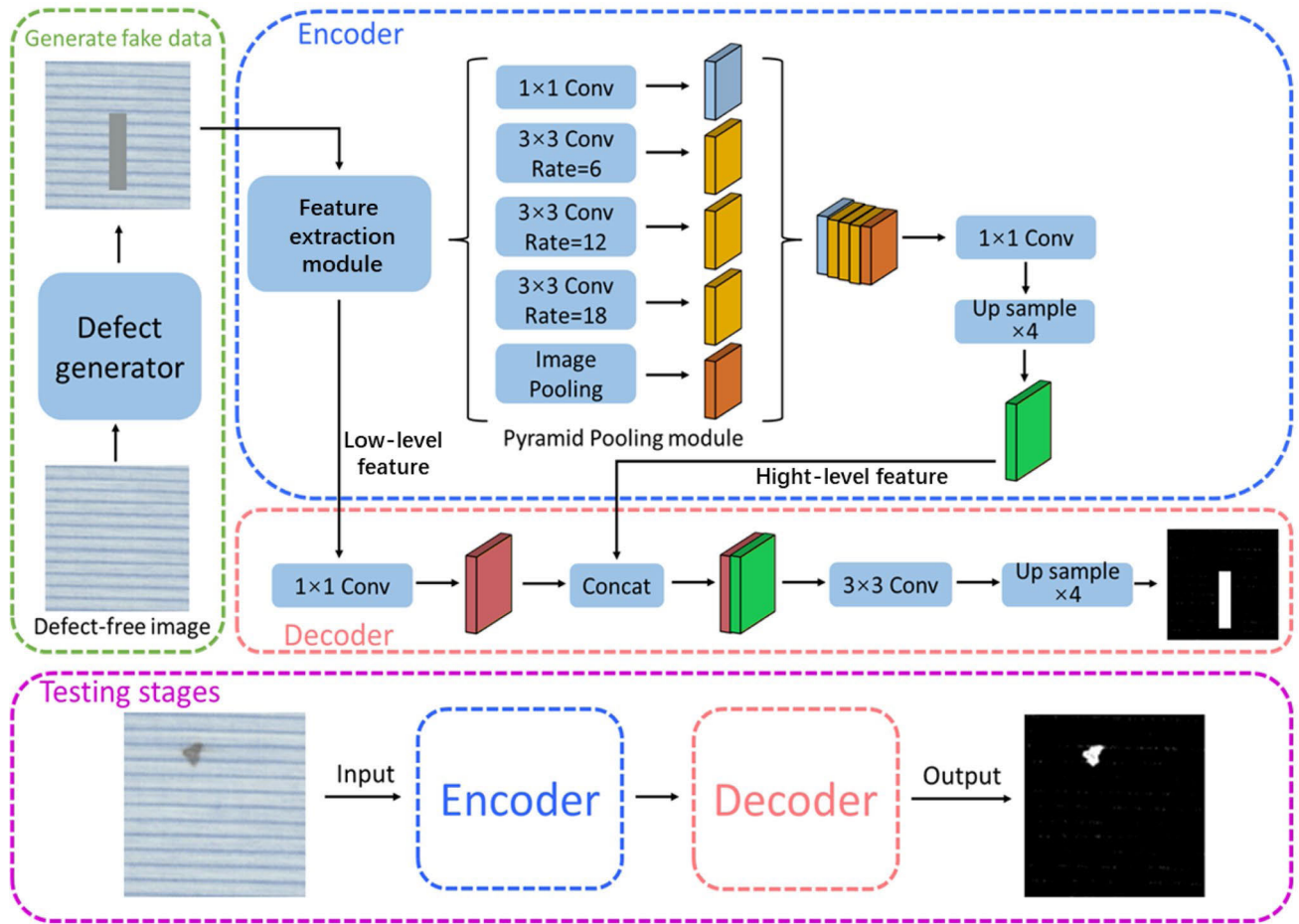


FIGURE 3. The overall block diagram of the proposed method.



FIGURE 4. Common types of fabric defects.

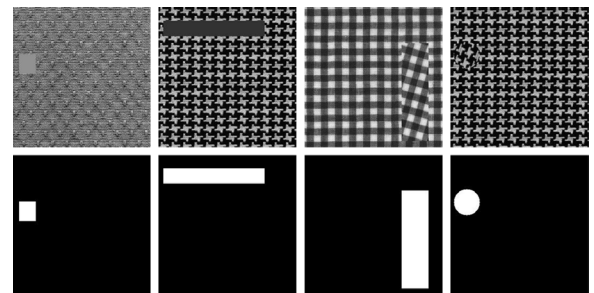


FIGURE 6. Fake fabric image samples with defects.

$$\begin{bmatrix} \text{Mask} & \text{Texture} \end{bmatrix} = \begin{bmatrix} \text{Mask} \end{bmatrix} \times \begin{bmatrix} \text{Texture} \end{bmatrix} + \begin{bmatrix} (1-\text{Mask}) & \text{Defect} \end{bmatrix}$$

FIGURE 5. Fake data set generation formula.

a lightweight network whose fusion of low-level features with high-level features. It consists of three parts: 1) lightweight low-level feature extraction module 2) pyramid pooling module 3) decoder module for feature upsampling.

Inspired by [10], the lightweight feature extraction module uses Depth-wise Convolution to reduce network parameters

and increase network detection speed. Standard convolution layer of a neural network involves input * output * width * height parameters, where width and height are width and height of filter. For an input channel of 30 and an output of 30 with a 3*3 filter, this will have 8130 parameters. Having so many parameters increases the chance of over-fitting. However, the depth-wise convolution only contains 2730 weight parameters.

The texture size and defect size of the fabric are variable. To solve the problem of different scales, a multi-scale module is added to the encoder part. The pyramidal pooling module

consists of a series of dilated convolution connected in parallel, combining local area context information with global context information. Convolutional networks were originally proposed for image classification when pooling and down-sampling enhance translation invariance, but result in loss of detail information, which can result in loss of detail in image segmentation tasks [10]. For segmentation tasks that need to be combined with image context information, the use of dilated convolution can significantly increase the receptive field and preserve the details. The dilated convolution is shown in Fig. 7. When the convolution kernel size is 3×3 , the receptive field of the conventional convolution is 3, and the receptive field of the dilated convolution is 5.

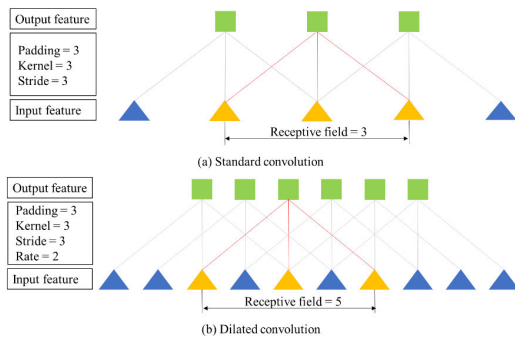


FIGURE 7. The comparison between dilated convolution and standard convolution.

The encoder features are first bilinearly unsampled by a factor of 4 and then concatenated with the corresponding low-level features. Before concatenating, 1×1 convolutions are applied on the low-level features to reduce the number of channels. After concatenation, a few 3×3 convolutions are applied, and the features are unsampled by a factor of 4. This gives the input size of the network is the same as the output size. The final layer is the sigmoid activation layer, which normalizes the feature map to $[0,1]$, representing the probability of the defect.

C. LOSS FUNCTION

The loss function is to measure the distance between the predicted label and the real label. For general image segmentation tasks, the loss functions are MSE, BCE and so on [27]. These loss functions first calculate the loss of each pixel separately and then sum as the final loss value. However, in the task of defect segmentation, the proportion of defect to the background is often unbalanced. As shown in Fig. 8, the loss of defect parts to the whole is relatively low. Using loss functions such as BCE will cause slow network convergence or even underfitting [28].

For defect segmentation tasks, the missed inspection rate and the false inspection rate are often important indexes, because for a sample with 5% defects, when all of them are predicted to be defect-free, the accuracy of this model is still 95%, which cannot objectively reflect the performance of the model. Therefore, we combine BCE loss and Dice loss

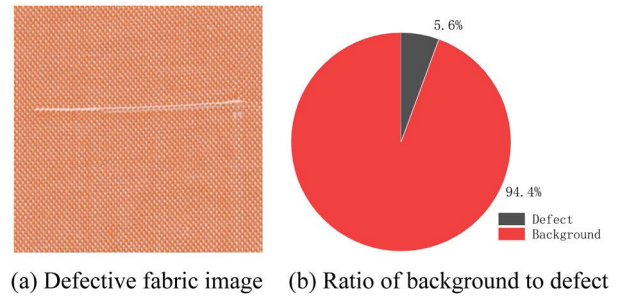


FIGURE 8. The imbalanced number of defects and background pixels.

and directly take the missed detection and false detection as the optimization goal to improve the learning ability of the model. The proposed *Defect Loss* can be defined as follows.

$$DefectLoss = Dic + CE \quad (1)$$

$$Dic = \frac{2 \sum_{i=1}^N p_i t_i + S}{\sum_{i=1}^N p_i + \sum_{i=1}^N t_i + S} \quad (2)$$

$$CE = -\frac{1}{N} \sum_{i=1}^N \beta (t_i - \ln p_i) + (1 - \beta) [(1 - t_i) \ln (1 - p_i)] \quad (3)$$

where i is the index of each pixel, N is the number of pixels in a picture, p_i is the probability of prediction into defects, and t_i is the label of pixel i . Also, add a smoothing term S to prevent the denominator from being 0.

IV. EXPERIMENT AND DISCUSSION

This section describes a set of experiments to evaluate the performance of the proposed method. The proposed method is compared with two fabric defect detection methods, PTIP [20] and LGM-FC [13], in terms of detection speed and accuracy. Accurately, to illustrate the detection speed of the proposed model, a comparison was made with several related methods in terms of detection time and the number of model parameters. Second, the proposed *Defect loss* function is compared to several commonly used loss functions to demonstrate its performance in the case of data imbalance. Third, the use of feature visualization demonstrates that fake datasets can fit well with real datasets. Finally, the combined detection performance of the proposed model with several excellent conventional methods is compared in both qualitative and quantitative terms.

A. EXPERIMENT PREPARATION

The proposed model is trained on a computer with two Nvidia GTX 1080Ti GPUs, and all compared experiments were conducted on the same computer, which was equipped with 128GB of RAM, an Intel Core i7 processor, and an Ubuntu 64-bit operating system. The proposed model was trained three times in the same configuration, and the model

was evaluated in the same configuration. The experimental results were obtained by calculating the mean of the three results. The proposed method was implemented using Pytorch. We used Adam optimizer to update the proposed model and initialized the weight of each layer using a Gaussian distribution with a zero mean and a standard deviation of 0.001. The initial learning rate was set to 0.001. The momentum was 0.9. The batch size is set to 2, with a total of 1800 iterations.

B. DATA SET

In order to verify the performance of the proposed method, two public datasets were utilized in this work: Fabric images Database (FID) [29] provided by Hong Kong University and Yarn-dyed Fabric Database (YFD) [30] which is collected from Guangdong Esquel Textiles (Guangdong Sheng, China). The images have a size of 256×256 pixels and contain multiple styles of textures (including star-patterned fabric, dot-patterned fabric, box-patterned fabric), some of which are shown in Fig. 9. In this method, the defect-free image is used to generate false data, and the defect image is used to evaluate the performance of the model.

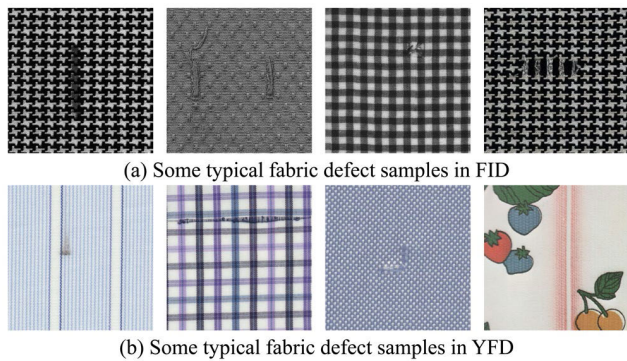


FIGURE 9. Some typical fabric defect samples.

C. METRICS

There are many metrics used to evaluate the performance of defect detection methods, the most commonly used one is ACC [31]. However, sometimes it cannot accurately reflect the segmentation performance (generally, the defect part of the fabric can only account for 5% of the whole picture, if a model predicts that the whole picture is defect-free, then the ACC of this algorithm is 0.95). Therefore, to evaluate the proposed model performance fairly and objectively. This article adopted four evaluation metrics [32]: ACC, Precision, Recall, and F-Measure, which are defined as below [33]:

$$ACC = \frac{TP + TN}{FN + TP + FP + TN} \tag{4}$$

$$Recall = \frac{TP}{FN + TP} \tag{5}$$

$$Precision = \frac{TP}{FP + TP} \tag{6}$$

$$F - Measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{7}$$

where TP , TN , FP , and FN represent the true positive, true negative, false positive, and false negative. $F-Measure$ is a comprehensive evaluator that utilizes both the $Precision$ and $Recall$ indicators. As the general rule, a higher $F-measure$ reflects a better detection performance.

D. EFFICIENCY EVALUATION

An online fabric defect detection system must be able to meet the real-time requirements. In order to evaluate the running speed of the model, the average detection time (ADT) for different size input image model is analyzed. Fig. 10 shows the curve of detection time with the size of the input image. As the image resolution increases, the detection time is also increased. Therefore, in order to balance the detection time and accuracy, the input image size of the model is set to 256×256 pixels. It can be observed that when the image size is 256×256 pixels, the ADT of the proposed model is 56 ms, which meets the real-time inspection requirement.

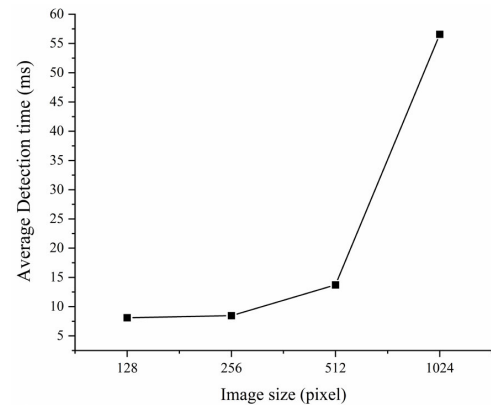


FIGURE 10. Detection time of different size images.

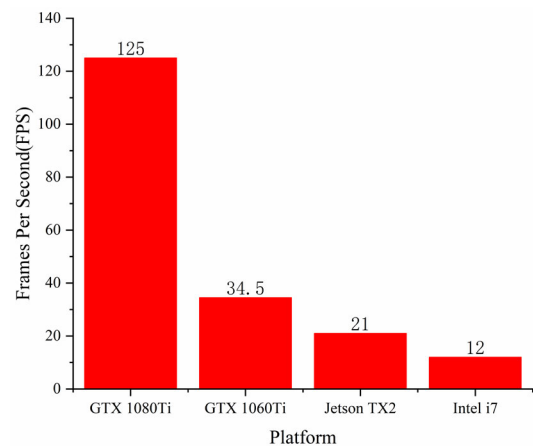


FIGURE 11. Detection time on different platforms.

As shown in Fig. 11, we also evaluated the detection time of the model on different hardware platforms, including GPU platform: GTX 1080Ti, GTX 1060Ti, CPU platform Intel i7 and embedded platform Jatson TX2. It can be seen that the proposed model has an excellent performance in detection time on GPU, CPU, and even embedded ARM platform.

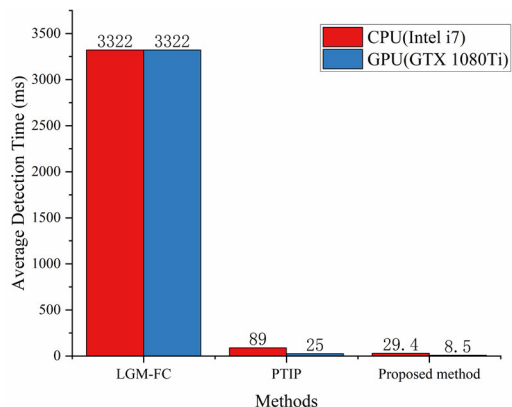


FIGURE 12. Comparison of detection time of different methods.

The efficiency of the proposed method, PTIP, LGM-FC method is tested on GPU and CPU platforms, respectively. The comparison results are shown in Fig. 12. The LGM-FC is a traditional machine vision-based method that first uses L0 Gradient Minimization to remove the fabric texture and then uses the clustering method to segment the defects. The detection time is long due to the iterative optimization required for each detection. PTIP is a method based on convolutional neural network, which first blocks the fabric according to its texture and then classifies it using a convolutional neural network. As the parameters of this method take longer to detect than the proposed method.

This experiment shows that the proposed model is more efficient than the existing methods. The proposed model not only has fast detection speed on GPU but also can achieve real-time detection effect on edge computing devices. This proves that our method can meet the real-time and low-cost requirements of industrial applications.

E. EFFECTIVENESS OF FAKE DATA

In order to verify the similarity between the fake data set and the real data set, principal components analysis (PCA) is used to reduce the dimension of the output of the encoder part of the network. For the convenience of visualization, we reduce the feature dimension to 2D, and the feature distribution of real data and fake data is shown in Fig. 13. As Fig. 13 shows, the feature distance between real data and false data is small, which proves that the network trained with fake data can be used to segment real defects.

F. INFLUENCE OF THE LOSS FUNCTION

As shown in Fig. 14, we have performed a qualitative comparison of several loss functions [34] using the DIC coefficient. It can be seen that the proposed loss function *Defect loss* has not only the highest segmentation accuracy but also the highest stability. The main advantage of *Defect loss* is to balance the missing rate and the false detection rate so that the model can quickly and stably converge when the number of background pixels and the number of defective pixels is unbalanced.

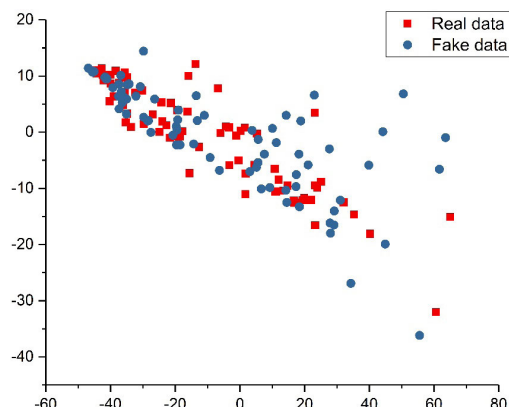


FIGURE 13. Visualization of the features of fake data and real data.

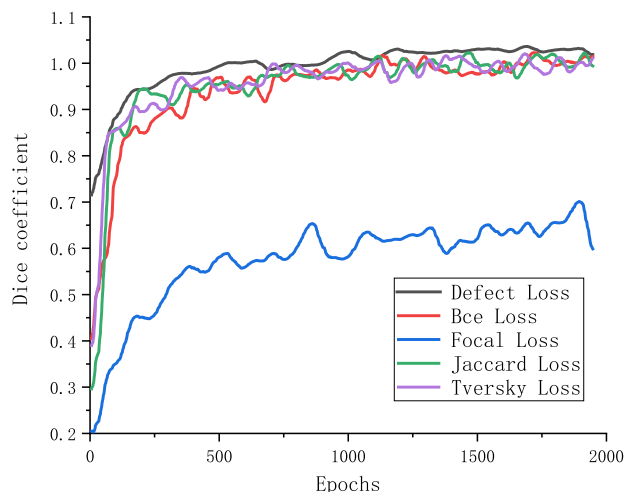


FIGURE 14. Comparison of different loss functions.

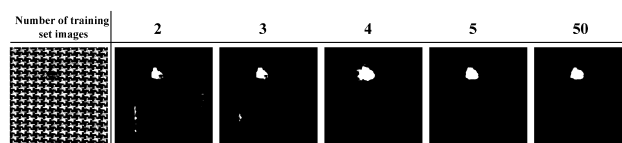


FIGURE 15. Small sample learning performance.

G. SMALL SAMPLE LEARNING PERFORMANCE

In order to evaluate the learning performance of the proposed model, we train and test the model respectively when the number of training sets is set to 2, 3, 4, 5, 50. The detect results are shown in Fig. 15. In addition, the comparison between the proposed method and the supervised method is shown in Table 2. The two methods are set to the same network structure, loss function and the number of iterations. It can be seen that the detection accuracy will increase with the increase of training samples. It is worth noting that even when the training samples are less than 5, the defect segmentation accuracy of the proposed model is significantly higher than that of the supervised method. Although when the number of training sets reaches 50, the accuracy of the supervised method is still less than that of the proposed method. In summary, the proposed detection method which does not need

TABLE 2. Small sample learning performance comparison.

Number of training set images	2	3	4	5	50
Ours	0.83	0.83	0.84	0.85	0.86
Supervised learning	0.36	0.45	0.56	0.60	0.82

to label any data is significantly better than the supervised method which needs to label data in few-shot learning.

H. PERFORMANCE COMPARISON

The proposed method was compared qualitatively with PTIP and LGM-FC methods on the same data set. As an illustration, five representative defect images and corresponding detection results on the given dataset are shown in Fig. 16 and Fig.17, respectively.

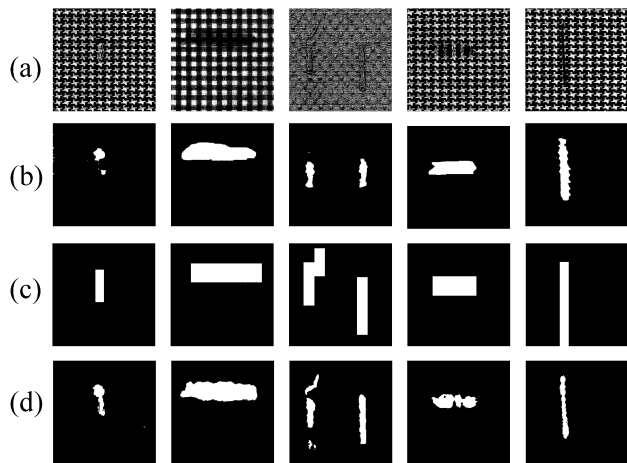


FIGURE 16. Comparison of FID data set detection results. (a) FID data set. (b) Results of LGM-FC. (c) Results of PTIP. (d) Results of ours.

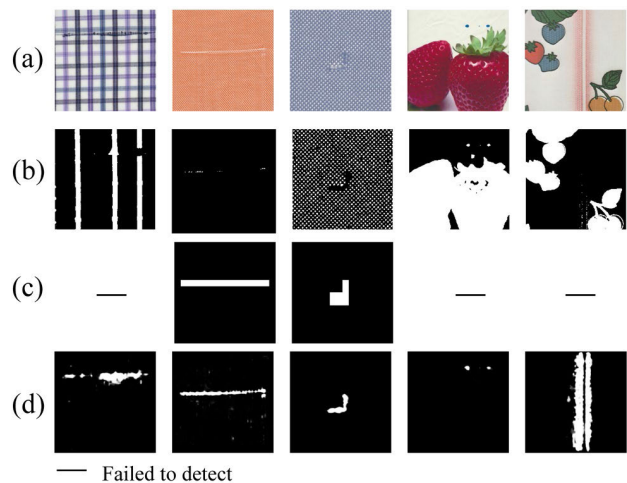


FIGURE 17. Comparison of YFD data set detection results. (a) YFD data set. (b) Results of LGM-FC. (c) Results of PTIP. (d) Results of ours.

Fig. 16 shows the results of the FID dataset, where the fabric images all have a regular texture. It can be seen that the

LGM-FC causes some defects to be misjudged as background because the LGM-FC uses the L0 gradient minimization method to remove the fabric texture and in the process of removing the texture, it also removes the detail at the edge of the defective area. In addition, since PTIP first divides the textured fabric into squares according to the period and then uses convolutional neural network to judge the squares as defects or backgrounds, the segmentation region is based on the squares, which makes the background region easy to judge as defects. The proposed approach is an end-to-end segmentation model that incorporates multi-scale modules to improve segmentation accuracy for small defects. Fig. 17 shows the results of the YFD dataset detection, where the fabric images contain some irregular textures. Since both the LGM-FC and PTIP methods are based on texture feature detection, neither method can detect images of irregularly textured fabrics. Moreover, the proposed method achieves good detection results for both regular and irregular textures.

TABLE 3. Acc, recall, precision, and F-measure of the proposed method.

Dataset	Methods	ACC	Recall	Precision	F-Measure
FID	LGM-FC	0.96	0.77	0.79	0.78
	PTIP	0.97	0.76	0.81	0.79
	Our	0.98	0.85	0.86	0.86
YFD	LGM-FC	-	-	-	-
	PTIP	-	-	-	-
	Our	0.97	0.86	0.83	0.85

TABLE 4. Comparison of detection time (ms).

Methods	LGM-FC	PTIP	FDD	ER	Ours
GPU	-	25.0	-	-	8.5
CPU	3322	89.0	1040	1200	29.4

The statistical results of our detection model on the FID and YFD are shown in Table 3. ACC of the proposed method can reach 0.97, and it can be demonstrated from the Recall and Precision that the proposed method achieves a balance between the missed and false detection rates, and F-Measure can reach 0.85 on both data sets. The detection time of the proposed method, LGM-FC, PTIP, FDD [30], and ER [29] on GPU and CPU platforms, is shown in Table 4. We can see that our method is faster than the other four methods.

V. CONCLUSION

This article describes an unsupervised defect detection method that is suitable for the detection of various textured fabric defects and requires only a small number of defect-free texture samples for training. In addition, the proposed Defect loss improves the segmentation performance when the defect is not balanced with the background number. A series

of experimental results on a variety of textured fabric detection data sets show that this method can achieve the most advanced detection accuracy and high detection efficiency. Besides, the method can be run in real-time, even on low-cost hardware.

REFERENCES

- [1] C. Li, G. Gao, Z. Liu, M. Yu, and D. Huang, "Fabric defect detection based on biological vision modeling," *IEEE Access*, vol. 6, pp. 27659–27670, 2018.
- [2] Y. Li, W. Zhao, and J. Pan, "Deformable patterned fabric defect detection with Fisher criterion-based deep learning," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 1256–1264, Apr. 2017.
- [3] D. Yapi, M. S. Allili, and N. Baaziz, "Automatic fabric defect detection using learning-based local textural distributions in the contourlet domain," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 1014–1026, Jul. 2018.
- [4] H. Y. T. Ngan, G. K. H. Pang, and N. H. C. Yung, "Automated fabric defect detection—A review," *Image Vis. Comput.*, vol. 29, no. 7, pp. 442–458, Jun. 2011.
- [5] G. Hu, J. Huang, Q. Wang, J. Li, Z. Xu, and X. Huang, "Unsupervised fabric defect detection based on a deep convolutional generative adversarial network," *Textile Res. J.*, vol. 90, nos. 3–4, pp. 247–270, Feb. 2020.
- [6] Y. Gandelsman, A. Shocher, and M. Irani, "'Double-DIP': Unsupervised image decomposition via coupled deep-image-priors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 11026–11035.
- [7] X. Tao, D. Zhang, W. Ma, X. Liu, and D. Xu, "Automatic metallic surface defect detection and recognition with convolutional neural networks," *Appl. Sci.*, vol. 8, no. 9, p. 1575, Sep. 2018.
- [8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
- [9] J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient CNNs in the wild," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 975–984, Mar. 2019.
- [10] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.
- [11] B. Zhang and C. Tang, "A method for defect detection of yarn-dyed fabric based on frequency domain filtering and similarity measurement," *Autex Res. J.*, vol. 19, no. 3, pp. 257–262, Sep. 2019.
- [12] M. Li, S. Wan, Z. Deng, and Y. Wang, "Fabric defect detection based on saliency histogram features," *Comput. Intell.*, vol. 35, no. 3, pp. 517–534, Aug. 2019.
- [13] H. Zhang, J. Ma, J. Jing, and P. Li, "Fabric defect detection using L0 gradient minimization and fuzzy C-means," *Appl. Sci.*, vol. 9, no. 17, p. 3506, Aug. 2019.
- [14] Y. Li, H. Luo, M. Yu, G. Jiang, and H. Cong, "Fabric defect detection algorithm using RDPSO-based optimal Gabor filter," *J. Textile Inst.*, vol. 110, no. 4, pp. 487–495, Apr. 2019.
- [15] A. A. Hamdi, M. M. Fouad, M. S. Sayed, and M. M. Hadhoud, "Patterned fabric defect detection system using near infrared imaging," in *Proc. 8th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS)*, vol. 51, Dec. 2017, pp. 111–117.
- [16] S. Mei, Y. Wang, and G. Wen, "Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model," *Sensors*, vol. 18, no. 4, p. 1064, Apr. 2018.
- [17] J. Jing, A. Dong, and P. Li, "Yarn-dyed fabric defect classification based on convolutional neural network," *Opt. Eng.*, vol. 56, no. 9, Sep. 2017, Art. no. 093104.
- [18] Y. Li, D. Zhang, and D.-J. Lee, "Automatic fabric defect detection with a wide-and-compact network," *Neurocomputing*, vol. 329, pp. 329–338, Feb. 2019.
- [19] W. Ouyang, B. Xu, J. Hou, and X. Yuan, "Fabric defect detection using activation layer embedded convolutional neural network," *IEEE Access*, vol. 7, pp. 70130–70140, 2019.
- [20] J. Jing, H. Ma, and H. Zhang, "Automatic fabric defect detection using a deep convolutional neural network," *Coloration Technol.*, vol. 135, no. 3, pp. 213–223, Jun. 2019.
- [21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3431–3440.
- [22] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 95–104.
- [23] J. Ahn and S. Kwak, "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 4981–4990.
- [24] L. Liu, J. Zhang, X. Fu, L. Liu, and Q. Huang, "Unsupervised segmentation and elm for fabric defect image classification," *Multimedia Tools Appl.*, vol. 78, no. 9, pp. 12421–12449, May 2019.
- [25] J. Liu, C. Wang, H. Su, B. Du, and D. Tao, "Multistage GAN for fabric defect detection," *IEEE Trans. Image Process.*, vol. 29, pp. 3388–3400, Dec. 2020.
- [26] M. Eldessouki, "Computer vision and its application in detecting fabric defects," in *Applications of Computer Vision in Fashion and Textiles*. Cambridge, U.K.: Woodhead Publishing, 2018, pp. 61–101.
- [27] R. M. Alguliyev, R. M. Aliguliyev, and F. J. Abdullayeva, "Privacy-preserving deep learning algorithm for big personal data analysis," *J. Ind. Inf. Integr.*, vol. 15, pp. 1–14, Sep. 2019.
- [28] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, p. 27, Dec. 2019.
- [29] C. S. C. Tsang, H. Y. T. Ngan, and G. K. H. Pang, "Fabric inspection based on the Elo rating method," *Pattern Recognit.*, vol. 51, pp. 378–394, Mar. 2016.
- [30] K. Zhang, Y. Yan, P. Li, J. Jing, X. Liu, and Z. Wang, "Fabric defect detection using saliency metric for color dissimilarity and positional aggregation," *IEEE Access*, vol. 6, pp. 49170–49181, 2018.
- [31] Y. Li, H. Li, and H. Wang, "Pixel-wise crack detection using deep local pattern predictor for robot application," *Sensors*, vol. 18, no. 9, p. 3042, Sep. 2018.
- [32] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation," *Neurocomputing*, vol. 338, pp. 139–153, Apr. 2019.
- [33] F. J. de la Calle Herrero, D. F. Garcia, and R. Usamentiaga, "Inspection system for rail surfaces using differential images," *IEEE Trans. Ind. Appl.*, vol. 54, no. 5, pp. 4948–4957, Sep. 2018.
- [34] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2999–3007.



ZHEN WANG received the B.S. degree in automation from Xi'an Polytechnic University, Xi'an, China, in 2018, where he is currently pursuing the M.S. degree with the School of Electronics and Information. His main research interests include computer vision and deep learning.



JUNFENG JING is currently a Professor with the School of Electronics and Information, Xi'an Polytechnic University, Xi'an, China. His main research interests include artificial intelligence, machine vision, image processing, and pattern recognition.

...