Wuhan University
Journal of Natural Sciences

2022, Vol.27 No.6, 539-549

Article ID 1007-1202(2022)06-0539-11
DOI https://doi.org/10.1051/wujns/2022276539

Semantic Segmentation Using DeepLabv3+ Model for Fabric Defect Detection

□ **ZHU Runhu[1], XIN Binjie[2†], DENG Na[1], FAN Mingzhu[1]**

1. School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China;

2. School of Textile and Fashion Technology, Shanghai University of Engineering Science, Shanghai 201620, China

Abstract: Currently, numerous automatic fabric defect detection algorithms have been proposed. Traditional machine vision algorithms that set separate parameters for different textures and defects rely on the manual design of corresponding features to complete the detection. To overcome the limitations of traditional algorithms, deep learning-based correlative algorithms can extract more complex image features and perform better in image classification and object detection. A pixel-level defect segmentation methodology using DeepLabv3+, a classical semantic segmentation network, is proposed in this paper. Based on ResNet-18, ResNet-50 and Mobilenetv2, three DeepLabv3+ networks are constructed, which are trained and tested from data sets produced by capturing or publicizing images. The experimental results show that the performance of three DeepLabv3+ networks is close to one another on the four indicators proposed (Precision, Recall, F1-score and Accuracy), proving them to achieve defect detection and semantic segmentation, which provide new ideas and technical support for fabric defect detection.

Key words: fabric defect detection; semantic segmentation; deep learning; DeepLabv3+

CLC number: TP 399

Received date: 2022-09-05

Foundation item: Supported by the National Natural Science Foundation of China (61876106) and Shanghai Local Capacity-Building Project (19030501200)

Biography: ZHU Runhu, male, Master candidate, research direction: fabric defect detection. E-mail: zhurunhufj@163.com

† To whom correspondence to be addressed. E-mail: xinbj@sues.edu.cn

0 Introduction

The textile industry is significant to China's economic and social development. Research has shown that defects typically lead to a 45%-60% decrease in the price of fabric[1], so fabric defect detection is an essential process in textile production. Currently, the textile industry still focuses on manual defect detection, whose accuracy is affected by subjective factors and lacks consistency[2]. With the development of computer vision and related technologies in recent years, numerous automatic fabric defect detection algorithms have been proposed to reduce the detection cost, improve detection efficiency, and further overcome the shortcomings of false detection[3]. These algorithms are divided into traditional compute vision algorithms and deep learning-based algorithms, of which the former has always the following disadvantages: Spatial domain statistics-based methods have a poor overall image analysis effect and are susceptible to noise interference[4, 5]; The frequency domain-based methods combine the general and local information of the image, but the detection effect of complex textures is poor[6,7]; Model-based algorithms can describe fabric textures well, but the calculation volume is large and the detection rate of more minor defects is low[8,9].

Compared with the traditional algorithm, the deep learning method based on Convolutional Neural Network (CNN) can extract the complex features of images better, thus achieving better results in image classification and target detection. Therefore, many neural network models are introduced into fabric defect detection

to overcome the limitations of traditional detection. Zhu *et al*[10] proposed a deep learning model for edge computing, reducing data transmission latency. By modifying the structure of DenseNet, it is more suitable for resource-constrained scenarios and optimizes cross-loss functions to better evaluate the proposed model; Xie *et al*[11] proposed a fabric defect detection method based on improved RefineDet, which improved defect location accuracy through the entire convolution channel attention block; Hu *et al*[12] proposed a fabric defect detection method based on Deep Convolution Generative Adversarial Network (DCGAN) and introduced a new encoder component to form a reconstruction network. The residual map was generated based on the original image and reconstruction in the testing stage. The residual map and the likelihood map generated by the model were then synthesized together to form an enhanced fusion map for defect segmentation; Jun *et al*[13] proposed a deep convolution neural network (DCNN) to improve the detection accuracy of the model by combining local defect prediction and global defect recognition; Elemmi *et al*[14] proposed a model based on MobileNet and Deep Residual Network for classifying defective and non-defective fabric images, which used morphological and feedback selection feature reduction algorithms to obtain significant features during image analysis.

Semantic segmentation combines object classification, object detection and image segmentation, which overcomes the limitation that the network cannot accurately recognize the target contour. It assigns specific labels to different image regions and eventually obtains segmented images with pixel-level semantic annotations. In fabric defect detection, semantic segmentation can provide reliable feature information for images, which is of great significance for processing subsequent visual tasks. Liu *et al*[15] detected fabric defects with single backgrounds based on an improved U-Net network, but it was not suitable for complex backgrounds; Liu *et al*[16] proposed a fabric defect detection framework based on the Generic Adversary Network (GAN). Through a multi-level GAN network, existing fabric defects can be automatically adapted to different textures, thus data sets can be formed by synthesizing defects on unblemished samples to solve the problems of data set scarcity and high annotation cost. The semantic segmentation network DeepLabv3 can detect defects in different textures by training the semantic segmentation network based on existing defect samples and the GAN network. This semantic segmentation network can detect multi-scale defects and can be fine-tuned to adapt the newly generated sample. Still, the effect of detecting large-area defects was not practical.

The research based on deep learning improves the universality of the model and makes up for the shortcomings of traditional algorithms[17]. Semantic segmentation combines many advantages of deep learning in defect detection and has made certain academic achievements after introducing fabric inspection[15,16]. However, there has been no relevant research on the performance effect of DeepLabv3+ architecture with different backbone networks in this field. While filling the research gap, it is noted that large and well-annotated data sets are required to train and test the model, so the pre-trained DCNN is used as the backbone in this paper.

Based on semantic segmentation and fabric defect detection, the main research contents of this paper are as follows: 1) The classical semantic segmentation network DeepLabv3+ is applied to fabric defect detection to realize pixel-level defect segmentation. 2) Image acquisition and web-based public image produce data sets for the training and evaluating models. 3) To further evaluate the performance differences of DeepLabv3+ models based on different backbone networks, Mobilenetv2, ResNet-18, and ResNet-50 feature extraction networks are used to build models for comparative experiments. The impact of different networks on the segmentation effect is also analyzed in detail.

The rest of this paper is as follows: In Section 1, the DeepLabv3+ network model and three different backbone network models are introduced; In Section 2, the production of data sets, the evaluation criteria of models and the specific data analysis of experiments are presented; In Section 3, the main conclusions are reviewed. Future research trends for semantic segmentation networks in fabric defect detection and other fields are also discussed.

1 Methodology

DeepLabv3+, one of the most effective models for semantic segmentation tasks, absorbs the advantages of Depthwise Separable Convolution (DSConv), Atrous Spatial Pyramid Pooling (ASPP), and Encoder-Decoder structure in the Deeplab series algorithms, which achieves 89.0% and 82.1% performance on the PASCAL VOC 2012 and Cityscape test sets[18], respectively. Based

on the DeepLabv3+ semantic segmentation network, a pixel-level defect segmentation algorithm is proposed in this paper, the architecture of which is shown in Fig. 1. The data set is formed according to captured images and publicly available images for training or testing the model.

The input of the model is the color fabric image and the output is the segmentation result with the pixel-level semantic label mask. ResNet-18, ResNet-50, and Mobilenetv2 are three DCNN that can be used as backbone networks to construct DeepLabv3+ semantic segmentation networks.

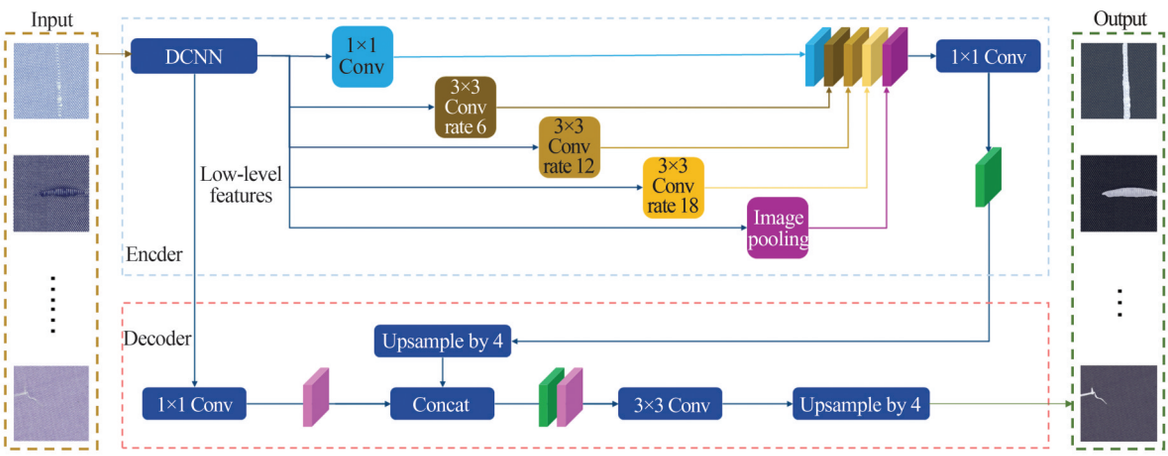

Fig. 1 The overall architecture of the proposed method

1.1 Model Architecture Details

DeepLabv1[19] has the atrous convolution with a larger convolution core or a greater receptive field, which aims to solve problems such as the loss of detailed information in downsampled. It uses a convolution layer to replace the fully connected layer of VGG16 and processes the details of segmentation results with the fully connected Conditional Random Field (CRF). DeepIabv2[20] changes the feature extraction network from VGG16 to ResNet and proposes an Atrous Spatial Pyramidal Pooling (ASPP) module. Multiscale feature fusion is achieved by cascading convolution layers with different atrous rates and the segmentation results are still processed by fully connected CRF. DeepLabv3[21] improved the ASPP module (ASPP+) by cascading or parallel layers of Batch Normalization (BN) and atrous convolution with different sampling rates, which achieves better results without CRF; In 2018, Chen *et al*[18] proposed a faster efficient semantic segmentation network DeepLabv3+ based on DeepLabv1-v3, which used the Xception model as a feature extraction network and retained the ASPP+ module to solve the target multi-scale problem. The classical Encoder-Decoder structure was also adopted. Specifically, the encoding network used DeepLabv3 to obtain rich semantic information and the decoding network obtained clear object boundaries. The use of Depthwise Separable Convolution reduced network parameters and greatly improved network speed. Table 1 summarizes the improvement process of the DeepLab series model.

Table 1 Improvement process from DeepLabv1 to DeepLabv3+

Module	DeepLabv1	DeepIabv2	DeepLabv3	DeepLabv3+
Feature extraction	VGG16	ResNet	ResNet+	Xception
Atrous convolution	√	√	√	√
CRF	√	√	×	×
ASPP	×	ASPP	ASPP+	ASPP+
Encoder-Decoder	×	×	×	√

Note: √ and × indicate that this network owns or does not own the module, respectively

Compared with conventional convolution,the greatest advantage of Depthwise Separable Convolution is high computational efficiency, which consists of two processes: Depthwise Convolution and Pointwise Convolution[22]. In Depthwise Convolution, one convolution kernel is responsible for one channel, while one channel is also convolved by only one convolution kernel. The number of channels output in this process is consistent with the number of channels input. Pointwise Convolution recovers lost cross-channel information, thus Fig. 2 compares different convolutions by the input of a three-channel image and the output of a four-channel feature map.

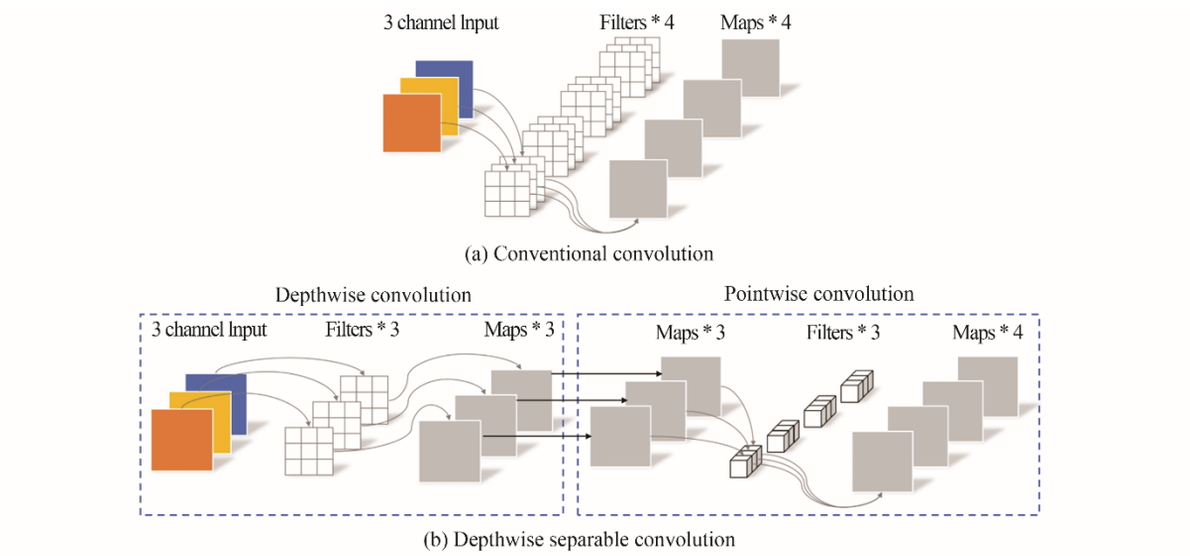

Fig. 2 Workflow comparison between two convolution methods

Atrous convolution controls the receptive field by filling 0 between two adjacent values in the convolution kernel, which can extract multi-scale information without changing the feature map size[23]. The Atrous Spatial Pyramid Pooling used in Deeplabv3 and Deeplabv3+ networks to extract semantic information at different resolutions consists of a 1×1 convolution layer, three 3×3 atrous convolutions and a global average pooling layer. Figure 3 shows a sample of Atrous Spatial Pyramid Pooling with rate 1,2,3.

DeepLabv3+ introduces the Encoder-Decoder architecture to improve network speed. In the Encoder section, the input image is extracted by a deep convolution backbone network. After that, the multiscale features are extracted by four downsamplings through parallel convolution layers, three atrous convolution layers with different rates and pooling layers. The number of channels in the feature layer is adjusted to 256 by splicing the multi-scale features and connecting them to a 1×1 convolution layer. In the Decoder section, the number of channels is adjusted to 48 through two downsamplings using a 1×1 convolution. After a module with the upsampling rate of 4, the output is the same size as the input. The loss function uses Cross Entropy Loss, which is the most widely used in semantic segmentation. The prediction value of pixels is compared with the target value of pixels one by one and then the average value of all pixels is obtained, which is defined by Eq. (1).

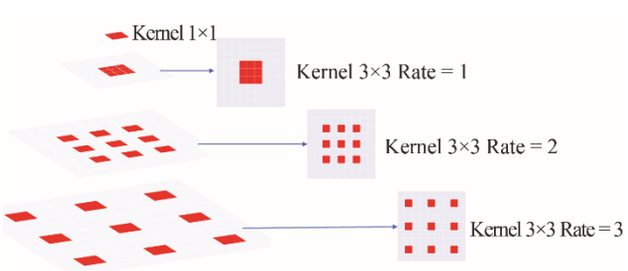

Fig. 3 Atrous Spatial Pyramid Pooling (rate = 1, 2, 3)

$$\text{Loss} = -\sum_{C=1}^{n} y_C \log\left(p_C\right) \tag{1}$$

where n represents the number of categories. y_C is 1 if the prediction is the same as the true value, or 0 if it is not; p_C indicates the prediction probability that the observed sample belongs to category C.

1.2 ResNet

The more layers of deep learning networks are, the more vulnerable their performance is to gradient disappearance, gradient explosion and degradation. He *et al*[24] proposed ResNet, which solved the gradient problem by

data preprocessing and alleviated the degradation problem by introducing the residual structure. The ResNet consists of three parts: The first part extracts the global features of the input image through a 7×7 convolution layer and a 3×3 maximum pool layer; The second part stacks multiple Resner-blocks with different specifications to learn global features further; The third part further processes the residual module data through a global average pooling layer and a fixed output full connection (FC) layer. Then, output results are converted by the softmax function. The network architecture of ResNet is shown in Table 2.

Table 2 The network architecture of ResNet

Layer name	Output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix}3\times3,\ 64\\3\times3,\ 64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,\ 64\\3\times3,\ 64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,\ 64\\3\times3,\ 64\\1\times1,\ 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,\ 64\\3\times3,\ 64\\1\times1,\ 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,\ 64\\3\times3,\ 64\\1\times1,\ 256\end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix}3\times3,\ 128\\3\times3,\ 128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,\ 128\\3\times3,\ 128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,\ 128\\3\times3,\ 128\\1\times1,\ 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,\ 128\\3\times3,\ 128\\1\times1,\ 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,\ 128\\3\times3,\ 128\\1\times1,\ 512\end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix}3\times3,\ 256\\3\times3,\ 256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,\ 256\\3\times3,\ 256\end{bmatrix}\times6$	$\begin{bmatrix}1\times1,\ 256\\3\times3,\ 256\\1\times1,\ 1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1,\ 256\\3\times3,\ 256\\1\times1,\ 1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,\ 512\\3\times3,\ 512\\1\times1,\ 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,\ 512\\3\times3,\ 512\\1\times1,\ 2048\end{bmatrix}\times3$
Others	1×1	Average pool, 1000-d FC, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ResNet-18 corresponds to the "18-layer" in Table 2, consisting of 17 convolution layers and one Full FC layer. The Resner-block is the BasicBlock architecture. The Conv Block that changes the dimensions of network data is selected when the input and output dimensions are different. Otherwise, the Identity Block that increases the number of network layers is selected. Basicblock contains two 3×3 convolution layers. The first connects a Batch Normalization (BatchNorm) layer and a ReLU activation function. The second only connects a BatchNorm layer. ResNet-50 corresponds to the "50-layer" in Table 2, consisting of 49 convolution layers and one FC layer. For ResNet with more than 50 layers, the Resner-block is the Bottleneck architecture, which is structured in a similar way to the BasicBlock. To reduce network parameters, the dimension of residual data is decreased to extract features and then increased to restore. The structure of two Resner-blocks is shown in Fig. 4.

1.3 Mobilenetv2

MobileNetv2 is a lightweight neural network improved on MobileNetv1, which follows the v1 version's deep separable convolution structure. It adds Linear Bottleneck and Inverted Residuals structures. The activation function is also changed from ReLU to ReLU6 to effectively reduce the loss of low-dimensional feature information[25]. Table 3 lists the parameters included in the MobileNetv2 network. t is the extension factor; c is the depth of the output characteristic matrix; n is the number of cycles of the Inverted Residual; s is the first step of each block and all convolution kernel size is 3×3; k is the depth of the input feature matrix.

Depthwise convolution layer extracting features is limited by input feature dimensions. With the classical residual structure in ResNet, fewer features are extracted after a 1×1 Pointwise Convolution and Depthwise Convolution. Therefore, MobileNetv2 first expands the feature map channel through the 1×1 Pointwise Convolution to enrich the number of features and improve accuracy. 3×3 Depthwise Convolution followed extracts features, then 1×1 Pointwise Convolution decreases dimen-

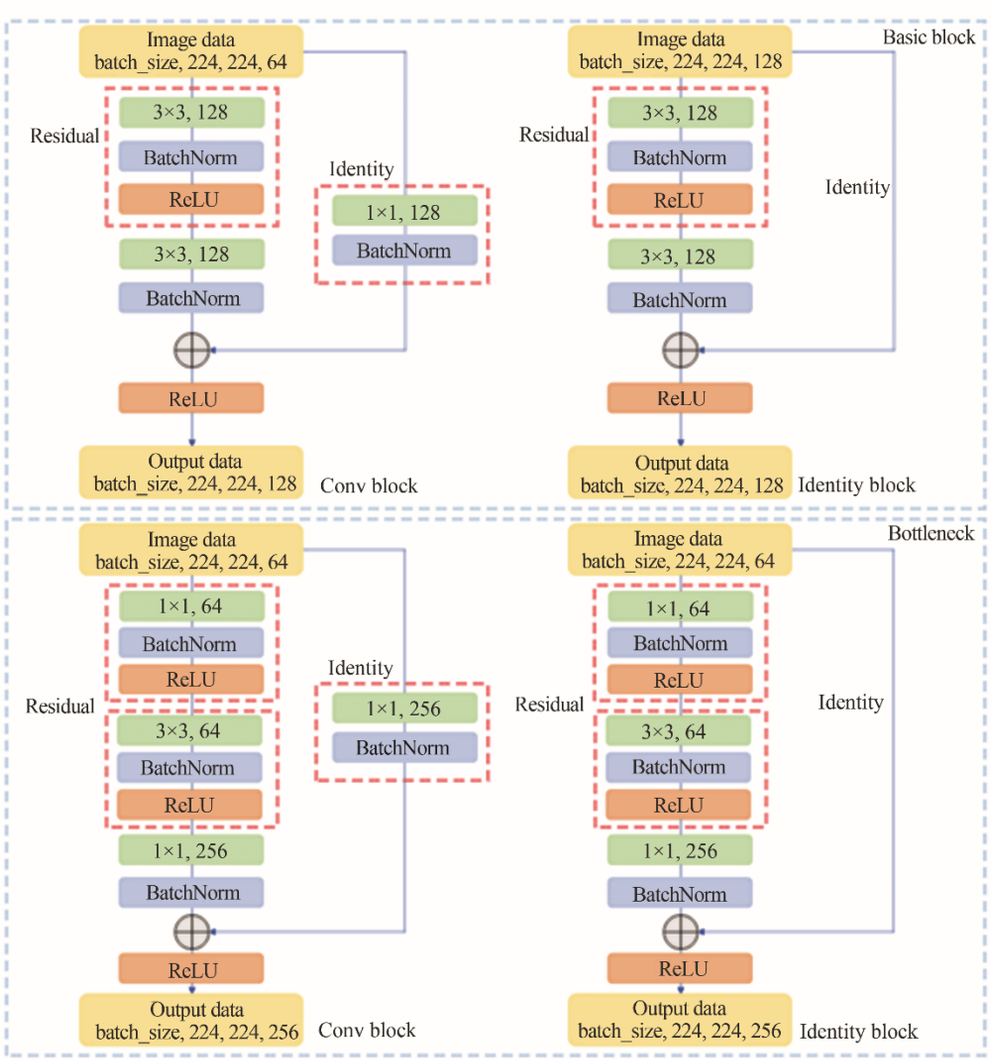

Fig. 4 The architectures of BasicBlock and Bottleneck

Table 3 MobileNetv2 Network parameters

Input	Operator	*t*	*c*	*n*	*s*
$224^2\times3$	conv2d	—	32	1	2
$112^2\times32$	bottleneck	1	16	1	1
$112^2\times16$	bottleneck	6	24	2	2
$56^2\times24$	bottleneck	6	32	3	2
$28^2\times32$	bottleneck	6	64	4	2
$14^2\times64$	bottleneck	6	96	3	1
$14^2\times96$	bottleneck	6	160	3	2
$7^2\times160$	bottleneck	6	320	1	1
$7^2\times320$	conv2d 1×1	—	1 280	1	1
$7^2\times1280$	avgpool 7×7	—	—	1	—
1×1×1280	conv2d 1×1	—	*k*	—	—

sion. This process is reversed with the order of the residuals, as shown in Fig. 5. Shortcut branching occurs only when the step is one and the input and output dimensions are the same.

2 Experiments and Results

Based on the classic semantic segmentation network DeepLabv3+, a pixel-level defect segmentation algorithm is designed and implemented in this paper. To further evaluate the performance difference between ResNet-18, ResNet-50, and Mobilenetv2 as backbone networks, the data sets are established to train and test three different networks. Five evaluation metrics (Precision, Recall, *F*1-score, Accuracy, and Reference time) are proposed to quantitatively analyze the segmentation

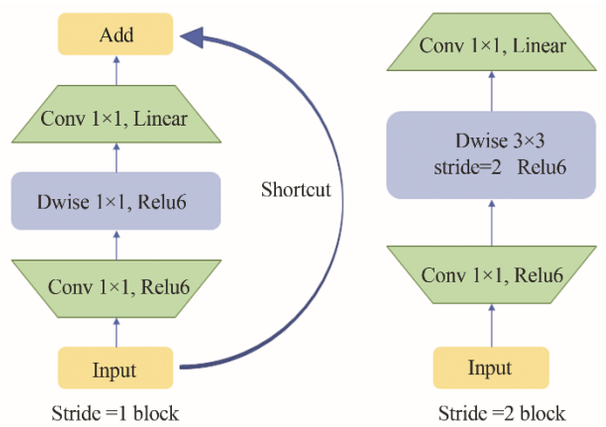

Fig. 5 The architecture of inverted residual

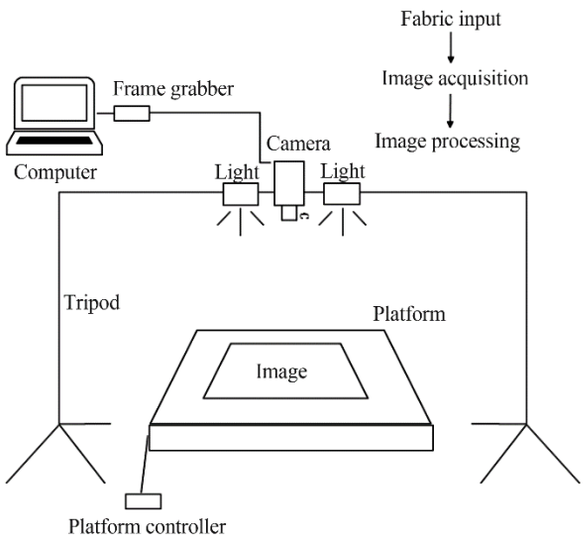

Fig. 6 Image acquisition system for fabric defect detection

results of test sets.

2.1 Experimental Setup

Figure 6 shows that a digital image acquisition system designed to establish the data set for experiments consists of a personal computer, frame grabber, camera, light source, tripod and platform controlled by the motor. The training model requires a large amount of data, so selecting the fabric defect image published on the network is necessary to expand the data set. Three thousand images from two sources contain four defects: Containing Yarn, Knot, Oil Stain and Cracked Ends. The resolution of all images in the data set is uniformly adjusted to 300×300 by the "imresize" function. The data set is divided into the training set and the test set according to the number of images 4∶1, which are 2 400 and 600, respectively. All images in the data set are manually labelled with the ground-truth through the "Image Labeler". In the network training, more samples are provided by data augmentation operations.

In this study, ResNet-18, ResNet-50 and Mobilenetv2 pre-trained by large data sets are used to construct three different DeepLabv3+ semantic segmentation networks, respectively. All experiments are based on MATLAB@2022a platform functions such as "Deep Learning Toolbox", "Computer Vision Toolbox", etc. The hardware used in the experiment and the parameter settings are shown in Table 4.

Table 4 Experimental environment and parameter settings

Experimental environment		Parameter setting	
Operating system	Windows10	Batch size	8
GPU	NVIDIA GTX 3070 8G	Epoch	100
CPU	AMD 5800H 3.2GHz	Initial learning rate	0.000 1
RAM	16 GB	Weight decay	0.000 5
Platform	MATLAB@2022a	Gradient descent algorithm	Adam

2.2 Evaluation Indicators

In the experiment, the input image pixels are divided into two categories, defect and background, to generate the output predicted image. Therefore, the related concepts in the confusion matrix are introduced as basic indicators: TP (True Positivity) indicates defect pixels successfully detected; FP (False Positivity) indicates defect pixels that have not been successfully detected; TN (True Negative) indicates background pixels successfully detected; FN (False Negative) indicates background pixels that cannot be successfully detected.

"Precision" indicates the proportion of all positive predictions that are correctly predicted. "Recall" indicates the proportion of all actual positive predictions that

are correct. $F1$-score represents the harmonic average of "Precision" and "Recall", which can balance the impact of "Precision" and "Recall" for a more comprehensive evaluation. "Accuracy" represents the proportion of the correct number of pixels in the prediction category to the total number of pixels. These four evaluating indicators are given by Eqs. (2)-(5).

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{3}$$

$$F1\text{ - score} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} = \frac{2\text{TP}}{2\text{TP}+\text{FP}+\text{FN}} \tag{4}$$

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \tag{5}$$

2.3 Results and Discussions

In this paper, three DeepLabv3+ semantic segmentation networks constructed by different backbone networks are trained and evaluated for the performance on the test set, respectively. Figure 7 shows the sample segmentation results including four kinds of defects, where (a) for Containing Yarn, (b) for Knot, (c) for Oil Stain and (d) for Cracked Ends. From the perspective of visual evaluation, the performance of the three networks is similar in some samples, such as B2-B4 and C2-C4 in Fig. 7(d); When the background is simple, the defects can be described well based on ResNet-50, such as A4 in Fig. 7(a) and A4 in Fig. 7(c); When the shape of fabric defects or image background is slightly complex, the performance based on Mobilenetv2 is more prominent, such as C2 in Fig. 7(a), A2 in Fig. 7(b); In some samples, the description based on ResNet-18 is closer to the natural shape of the fabric, such as C3 in Fig. 7(c).

Figure 8 shows each evaluation indicator value and inference time of DeepLabv3+ semantic segmentation networks constructed by three different backbone networks. The values of three networks under the four evaluation indicators are relatively close, but the inference time is quite different. Mobilenetv2 as the backbone network has the best evaluation on Precision, Recall and $F1$-score.

The highest accuracy is achieved based on ResNet-

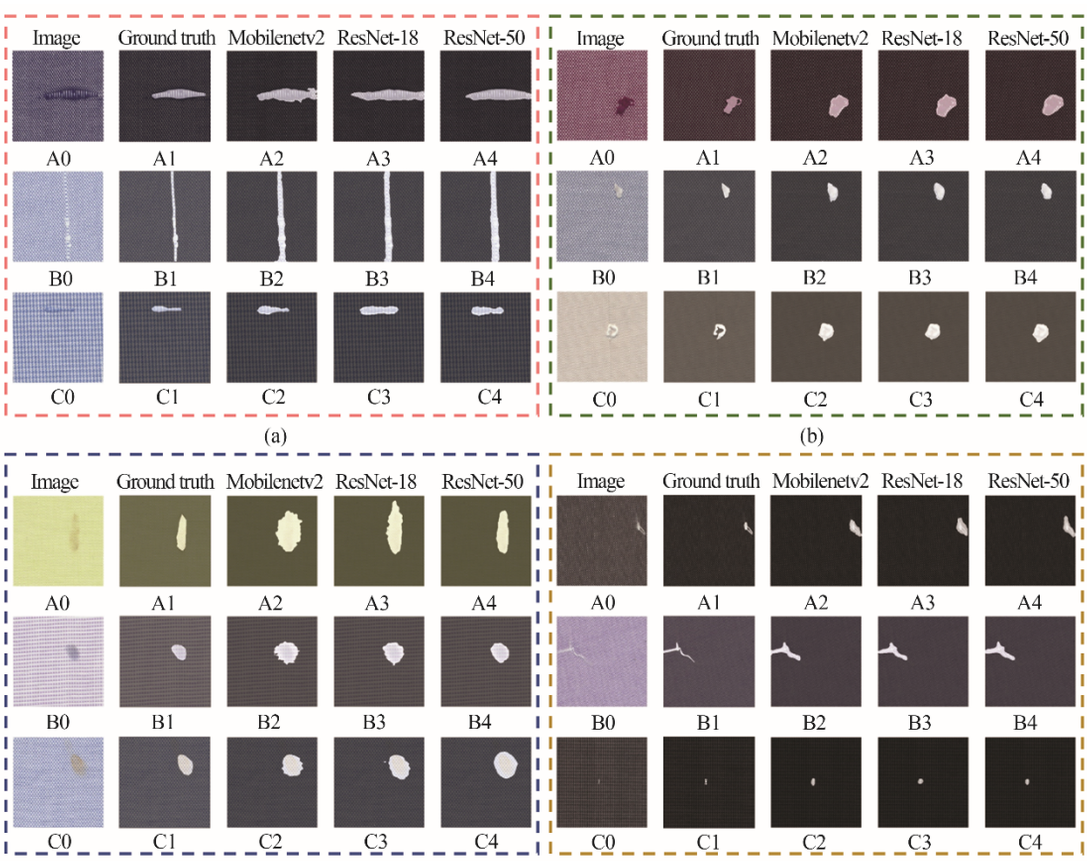

Fig. 7 Sample images
(a) Containing Yarn; (b) Knot; (c) Oil Stain (d) Cracked Ends

50. The inference time based on ResNet-18 is the shortest. In general, the performance of the three networks is similar and all accuracy rates are above 0.96, which shows that the pixel-level segmentation of defects can be achieved. However, ResNet-18 has fewer layers and Mobilenetv2 is also a lightweight network, so the inference time based on both is shorter than that based on ResNet-50.

Table 5 shows the performance comparison of some common algorithms for fabric defect detection. Algorithms ①-③ proposed in this paper are highlighted by bold fonts, and algorithm ③ has the highest accuracy. Algorithm ④ uses the Gray-Level Co-occurrence Matrix (GLCM) to extract image features and segment defects by *K*-means clustering. Algorithm ⑤ calculates the optimal Gabor filter based on the Genetic Algorithm (GA) to detect defects. Algorithms ④ and ⑤ are traditional algorithms and algorithms ⑥-⑧ are common deep learning segmentation networks. The results show that the detection accuracy of the proposed algorithms in this paper is higher than that of other common fabric defect detection algorithms with research and application value.

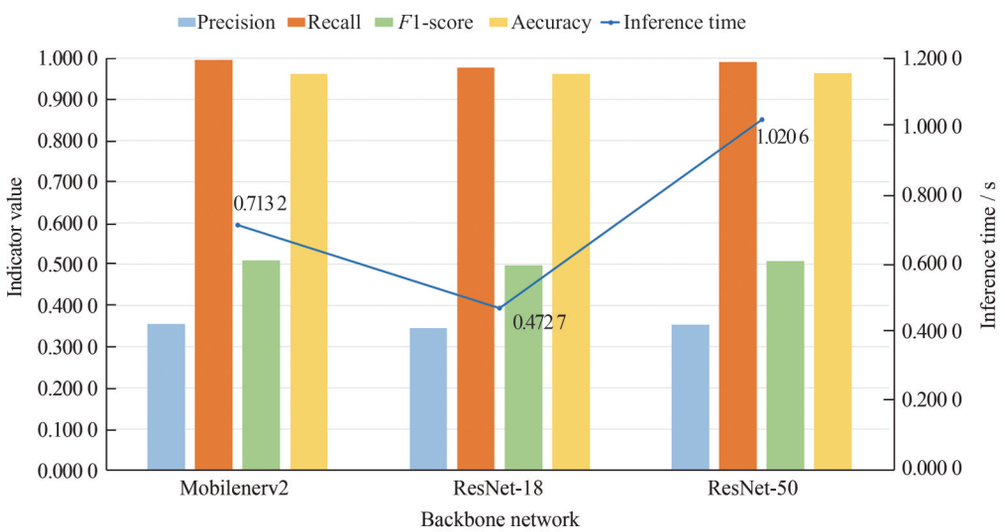

Fig. 8 Indicator values and inference time of different backbone networks in DeepLabv3+
The unit of inference time represents the time required to process each image

Table 5 Detection accuracy comparison of the proposed algorithm

Serial Number	Method	Accuracy
①	**DeepLabv3+_ Mobilenetv2**	**0.961 7**
②	**DeepLabv3+_ ResNet-18**	**0.961 5**
③	**DeepLabv3+_ ResNet-50**	**0.963 2**
④	GLCM + K-means	0.874 3
⑤	Gabor + GA	0.927 4
⑥	SegNet	0.938 6
⑦	UNet	0.941 5
⑧	FCN	0.957 7

3 Conclusion

This paper studies the classical semantic segmentation network DeepLabv3+ and discusses its feasibility in fabric defect detection. Thus, a pixel-level defect segmentation method based on DeepLabv3+ is proposed. To further evaluate network performance, the DeepLabv3+ are constructed by three different backbone net-

works, Mobilenetv2, ResNet-18, and ResNet-50, respectively. To meet the demand of the experimental data set, a digital image acquisition system is designed and constructed. Based on the collected images and network public images, a data set is constructed for model training and verification. The experimental results show that the segmentation network based on Mobilenetv2 has the highest Accuracy, Recall and *F*1 score values, which are 0.354 6, 0.995 9 and 0.509 2, respectively. The accuracy based on ResNet-50 is the highest at 0.963 2. The inference time based on ResNet-18 is the fastest, 0.472 7 s (processing one image). The performance of three DeepLabv3+ semantic segmentation networks on the four proposed segmentation evaluation indicators is relatively close and the accuracies each are over 96%, so the pixel-level segmentation based on DeepLabv3+ is feasible.

Since all data images are manually labelled, there is inevitably an accuracy error. At the same time, hardware and other devices have limitations, which further affect the training and prediction of the model. In the future, optimizing the quantity and quality of data sets will become the focus of work, and these improvements will be more conducive to model fitting and result verification. The framework of Deeplabv3+ will also be improved in addition to researching the backbone networks. Modules such as the attention mechanism will be introduced to improve the detection accuracy and ability to describe the shape of defects, so as to adapt to smaller defects or more complex texture backgrounds.

References

[1] Stojanovic R, Mitropulos P, Koulamas C, *et al*. Real-time vision-based system for textile fabric inspection [J]. *Real-Time Imaging*, 2001, **7**(6): 507-518.

[2] Xia D, Jiang G M, Li Y Y, *et al*. Warp-knitted fabric defect segmentation based on non-subsampled Contourlet transform [J]. *The Journal of the Textile Institute*, 2017, **108**(2): 239-245.

[3] Ngan H Y T, Pang G K H, Yung N H C. Automated fabric defect detection—A review [J]. *Image and Vision Computing*, 2011, **29**(7): 442-458.

[4] Zhang Y F, Bresee R R. Fabric defect detection and classification using image analysis [J]. *Textile Research Journal*, 1995, **65**(1): 1-9.

[5] Alper Selver M, Avşar V, Özdemir H. Textural fabric defect detection using statistical texture transformations and gradient search [J]. *The Journal of the Textile Institute*, 2014, **105**(9): 998-1007.

[6] Yang X Z, Pang G K H, Yung N H C. Discriminative fabric defect detection using adaptive wavelets [J]. *Optical Engineering*, 2002, **41**: 3116-3126.

[7] Chan C H, Pang G K H. Fabric defect detection by Fourier analysis [J]. *IEEE Transactions on Industry Applications*, 2000, **36**(5): 1267-1276.

[8] Cohen F S, Fan Z, Attali S. Automated inspection of textile fabrics using textural models [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1991, **13**(8): 803-808.

[9] Bu H G, Huang X B, Wang J, *et al*. Detection of fabric defects by auto-regressive spectral analysis and support vector data description [J]. *Textile Research Journal*, 2010, **80**(7): 579-589.

[10] Zhu Z W, Han G J, Jia G Y, *et al*. Modified DenseNet for automatic fabric defect detection with edge computing for minimizing latency [J]. *IEEE Internet of Things Journal*, 2020, **7**(10): 9623-9636.

[11] Xie H, Wu Z. A robust fabric defect detection method based on improved RefineDet [J]. *Sensors* (*Basel*, *Switzerland*), 2020, **20**(15): E4260.

[12] Hu G H, Huang J F, Wang Q H, *et al*. Unsupervised fabric defect detection based on a deep convolutional generative adversarial network [J]. *Textile Research Journal*, 2020, **90**(3/4): 247-270.

[13] Jun X, Wang J G, Zhou J, *et al*. Fabric defect detection based on a deep convolutional neural network using a two-stage strategy [J]. *Textile Research Journal*, 2021, **91**(1/2): 130-142.

[14] Elemmi M C, Anami B S, Malvade N N. Defective and non-defective classif ication of fabric images using shallow and deep networks [J]. *International Journal of Intelligent Systems*, 2022, **37**(3): 2293-2318.

[15] Liu R Q, Li M H, Shi J C, *et al*. Fabric defect detection method based on improved U-net [J]. *Journal of Physics*: *Conference Series*, 2021, **1948**(1): 012160.

[16] Liu J H, Wang C Y, Su H, *et al*. Multistage GAN for fabric defect detection [J]. *IEEE Transactions on Image Processing*, 2020, **29**: 3388-3400.

[17] Jing J F, Zhuo D, Zhang H H, *et al*. Fabric defect detection using the improved YOLOv3 model [J]. *Journal of Engineered Fibers and Fabrics*, 2020, **15**: 155892502090826.

[18] Chen L C, Zhu Y K, Papandreou G, *et al*. Encoder-decoder with atrous separable convolution for semantic image segmentation [C]// *Computer Vision-ECCV* 2018, 2018: 801-808.

[19] Chen L C, Papandreou G, Kokkinos I, *et al*. Semantic image segmentation with deep convolutional nets and fully connected CRFs [EB/OL]. [2022-10-22]. *https://www.semanticscholarorg/reader/*39*ad*6*c*911f3351*a*3*b*390130*a*6*e*4265355*b*4*d*593*b*.

[20] Chen L C, Papandreou G, Kokkinos I, *et al*. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018, **40**(4): 834-848.

[21] Chen L C, Papandreou G, Schroff F, *et al*. Rethinking atrous convolution for semantic image segmentation [EB/OL]. [2022-09-07]. *https://www. semanticscholar. org/reader/ee*4*a*012*a*4b12*d*11*d*7*ab*8*c*0*e*79*c*61*e*807927*a*163*c*.

[22] Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, **39**(4): 640-651.

[23] Dumoulin V, Visin F. A guide to convolution arithmetic for deep learning [EB/OL]. [2022-09-24]. *https://www.semanticscholar. org/reader/f*19284*f*6*ab*802*c*8*a*1*fcde*076*fcb*3*fba*195*a*71723.

[24] He K M, Zhang X Y, Ren S Q, *et al*. Deep residual learning for image recognition [C]// 2016 *IEEE Conference on Computer Vision and Pattern Recognition*. New York: IEEE, 2016: 770-778.

[25] Sandler M, Howard A, Zhu M L, *et al*. MobileNetV2: Inverted residuals and linear bottlenecks [C]// 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. New York: IEEE, 2018: 4510-4520.

□