doi:10.3772/j.issn.1006-6748.2020.02.009

RGB and LBP-texture deep nonlinearly fusion features for fabric retrieval①

Shen Fei(沈　飞)*, Wei Mengwan*, Liu Jiajun*, Zeng Huanqiang**, Zhu Jianqing②*
(*College of Engineering, Huaqiao University, Quanzhou 362021, P. R. China)
(**School of Information Science and Engineering, Huaqiao University, Xiamen 361021, P. R. China)

Abstract

Fabric retrieval is very challenging since problems like viewpoint variations, illumination changes, blots, and poor image qualities are usually encountered in fabric images. In this work, a novel deep feature nonlinear fusion network (DFNFN) is proposed to nonlinearly fuse features learned from RGB and texture images for improving fabric retrieval. Texture images are obtained by using local binary pattern texture (LBP-Texture) features to describe RGB fabric images. The DFNFN firstly applies two feature learning branches to deal with RGB images and the corresponding LBP-Texture images simultaneously. Each branch contains the same convolutional neural network (CNN) architecture but independently learning parameters. Then, a nonlinear fusion module (NFM) is designed to concatenate the features produced by the two branches and nonlinearly fuse the concatenated features via a convolutional layer followed with a rectified linear unit (ReLU). The NFM is flexible since it can be embedded in different depths of the DFNFN to find the best fusion position. Consequently, DFNFN can optimally fuse features learned from RGB and LBP-Texture images to boost the retrieval accuracy. Extensive experiments on the Fabric 1.0 dataset show that the proposed method is superior to many state-of-the-art methods.

Key words: fabric retrieval, feature fusion, convolutional neural network(CNN)

0 Introduction

Taking a query fabric image as an input, fabric retrieval aims to search from a gallery fabric image dataset and return a list of most similar fabric images. Hence, fabric retrieval plays an important role in improving the production efficiency in textile industries. However, fabric retrieval is very challenging because fabric images usually contain illumination changes, blots, complex textures, and poor image qualities.

As an image retrieval problem, the fabric image representation is essential for fabric retrieval. Generally, feature representation methods can be mainly divided into 2 types: hand-crafted features and deep-learned features. For hand-crafted features, many well-known hand-crafted features (e.g., histogram of orientation gradient (HOG)[1], local binary pattern (LBP)[2] and speed-up robust feature (SURF)[3]) are designed based on the knowledge and experience of researchers. For deep-learned features, there are also a wide range of famous deep networks, such as AlexNet[4], VGG-Net[5], GoogleNet[6], ResNet[7], DenseNet[8].

Mainstream academic researchers[5,9] generally believe that hand-crafted features (e.g., HOG[1], LBP[2], and SURF[3]) allow for a better understanding of shallowing image representation methods. In this light, hand-crafted features are possible to be replaced by deep-learned features using shallow neural networks. However, deep-learned features are automatically obtained by using deep networks trained from RGB images. Human cannot intuitively understand the mean of deep-learned features and lack efficient guidance to improve deep-learned features except deepening a network's depth. Therefore, there is a great requirement for finding a method to combine hand-crafted and deep-learned features efficiently.

Intuitively, texture features are significant for fabric images, thus, fusing texture features with deep features learned from RGB images are expected to improve the fabric retrieval performance. However, texture features are considered as low-level features, while deep-

① Supported by the National Natural Science Foundation of China (No. 61976098, 61871434, 61802136, 61876178), Open Foundation of Key Laboratory of Security Prevention Technology and Risk Assessment, People's Public Security University of China (No. 18AFKF11), Science and Technology Bureau of Quanzhou (No. 2018C115R) and the Subsidized Project for Postgraduates' Innovative Fund in Scientific Research of Huaqiao University (No. 18014084008).

② To whom correspondence should be addressed. E-mail: jqzhu@hqu.edu.cn

Received on June 8, 2019

learned features resulting from RGB images are high-level features. If they are directly concatenated, there is a huge risk of losing the balance of hand-crafted and deep-learned features which is harmful to boost the fabric retrieval performance. For that, a novel deep feature nonlinear fusion network (DFNFN) is proposed to optimally fuse texture features and deep-learned features. More specifically, DFNFN simultaneously processes a RGB image and the corresponding texture image through two feature learning branches. Each branch contains the same convolutional neural network (CNN) architecture but independently learning parameters. It is worth noting that texture images are obtained by using local binary pattern texture (LBP-Texture)[2] features to describe RGB fabric images. In addition, a nonlinear fusion module (NFM) is designed to concatenate features generated by the two feature learning branches and the concatenated features are further nonlinearly fused through a convolution layer followed by a rectified linear unit (ReLU). NFM is so flexible that can be embedded in the DFNFN at different depths to find the best fusion position. Therefore, DFNFN can optimally fuse features learned from RGB images and texture images to improve the fabric retrieval accuracy. Extensive experiments show that the proposed DFNFN method is superior to multiple state-of-the-art methods on the Fabric 1.0[10] dataset.

The rest of this paper is organized as follows. Section 1 introduces the related work. Section 2 describes the proposed fabric retrieval method in detail. Section 3 presents experiment and analysis results to show the superiority of the proposed method. Section 4 is the conclusion of this work.

1 Related work

1.1 Hand-crafted feature

As a fabric retrieval task, the most important thing is to design a suitable feature descriptor for representing fabric images. Many hand-crafted feature descriptors (e. g., HOG[1], LBP[2], and SURF[3]) have been proposed to represent images. For example, the combination of LBP[2] histograms and support vector machine (SVM) classifiers is commonly used to fabric recognition[11] and fabric classification[12]. Jing et al.[13] proposed a multi-scale texture feature extraction method to merge LBP-Texture features[2] of different scales for fabric retrieval. Besides LBP-Texture features[2], Suciati et al.[14] combined fractal-based texture features and HSV color features to exploit the salience information of fabric images. Yao et al.[15] fused GIST[16] and SURF[3] features for fabric retrieval.

1.2 Deep-learned feature

With the rapid development of deep learning algorithms, more and more attention has been paid to deep-learned feature based fabric retrieval works. For example, based on VGGNet[5], Liu et al.[17] simultaneously learned attribute features and landmark features for fabric retrieval. Xiang et al.[18] used GoogleNet[6] as a feature extractor, then converted the learned features into binary codes to improve the accuracy of fabric retrieval. Bell et al.[19] concatenated the features produced by different depth convolutional layers of a deep network to improve the feature representation ability. In addition, some ultra-deep networks, i. e., ResNet[4] and DenseNet[8], are sensitive to learn more discriminate features for fabric retrieval, according to Refs[20,21].

1.3 Fusion of hand-crafted and deep-learned features

Recently, the fusion of hand-crafted features and deep-learned features has achieved excellent performance in the field of image retrieval. Overall, there are 2 methods of fusing hand-crafted and deep-learned features, i. e., serial fusion and parallel fusion.

For the serial fusion method[22,23], RGB images are firstly transformed into some types of hand-crafted features, and then the hand-crafted features are input into a deep network for learning deep-learned features. For example, Ref. [22] firstly extracts LBP-Texture[2] images of RGB images and then trains a CNN from LBP-Texture[2] images. Hand-crafted features hold some robustness properties, e. g., LBP-Texture[2] is robust to illumination changes and SIFT[24] is invariant to scale variations. However, they inevitably lose some intrinsic information of RGB images. For instance, LBP-Texture loses color information, while SIFT[24] cannot take spatial information. The intrinsic information losing is harmful to the subsequently deep feature learning, and the accuracy performance is restricted.

For the parallel fusion approach[25], the network contains 2 feature learning branches to learn the hand-crafted features and deep-learned features simultaneously. Specifically, Ref. [25] applied VGGNet[5] and weighted HOG[25] to obtain the deep-learned features and hand-crafted features, respectively. Then, they integrated deep-learned features and hand-crafted features via a fully connected layer to acquire a strong feature descriptor for pedestrian gender classification. This parallel fusion approach has a huge risk of losing the balance of hand-crafted and deep-learned features since the hand-crafted feature is a type of shallow features.

2 Proposed method

2.1 Overview

Fig. 1 shows the framework of the proposed DFNFN. It can be seen that the DFNFN contains two inputs and one output. The 2 inputs are RGB fabric images and the corresponding LBP-Texture images, respectively. Both RGB fabric images and the corresponding LBP-Texture images are processed with 2 feature learning branches. The 2 branches are merged to form single output by using the proposed NFM. The details of DFNFN will be described in the following subsections.

2.2 Nonlinear fusion module

2.2.1 Feature extraction on RGB images

As shown in Fig. 1, the feature extraction on RGB images is completed by the RGB feature learning branch. For ease of description, a convolutional (Conv) layer, a batch normalization (BN) layer and a ReLU are sequentially packaged to construct a CBR block. Then, 3 CBR blocks (i. e., CBR1-CBR3) and 3 max-pooling layers (i. e., MP1-MP3) are packaged in turn to build the RGB feature learning branch.

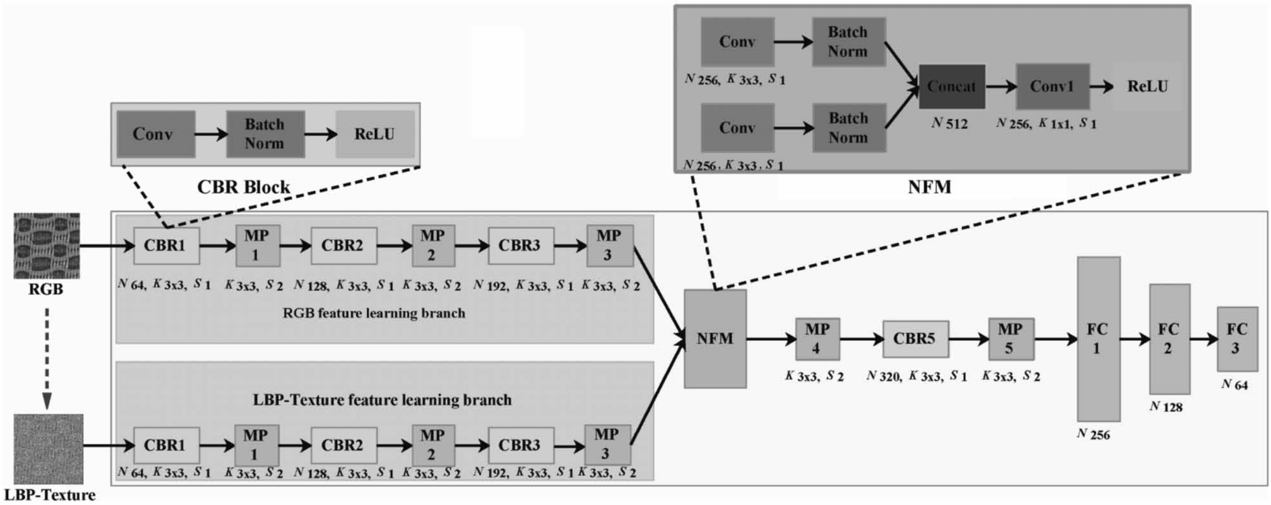


Fig. 1 The framework of the proposed deep feature nonlinear fusion network

MP, NFM, and FC represent max-pooling, nonlinear fusion module, and fully connected layers, respectively. For those capital letters lying under each component, N is the channel number of the resulted features; K means a filter size if the component is a convolution layer, while K represents a pooling window size once the component is a pooling layer; S denotes a stride size working on the component.

2.2.2 Feature extraction on LBP images

As shown in Fig. 1, LBP-Texture images are fed into the LBP-Texture feature learning branch. Following Ref. [26], an RGB image is firstly gray processed and then the 3 × 3 sized LBP[26] encoding operation is applied to transform the gray image into a LBP-Texture image. As shown in Fig. 2, the 3 × 3 sized LBP[26] encoding operation is implemented between each spatial position and its 8 neighbors. That is, the pixel value of a spatial position is applied as a threshold, and if a neighbor's pixel value is smaller than the threshold, the corresponding code is 0, otherwise, it is 1. In this way, 8-bit binary codes are obtained at each position. Those 8-bit binary codes are further transformed into decimal representations by using Eq. (1). As a result, the LBP-Texture image is acquired, and its pixel values are ranged in [0, 255].

$$LBP_{3\times3}(g_c) = \sum_{i=0}^{8} l(g_i > g_c) \times 2^i \qquad (1)$$

where, g_c is the gray value of a spatial position and g_i represents the 8-neighbor of g_c. l is an indicator function, which is equal to 1 if $g_i > g_c$, otherwise, it is equal to 0.

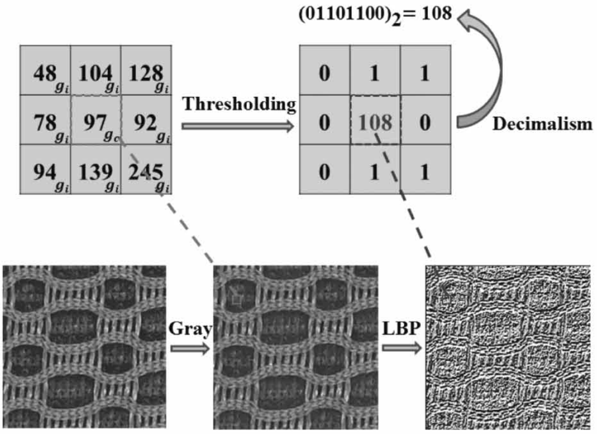


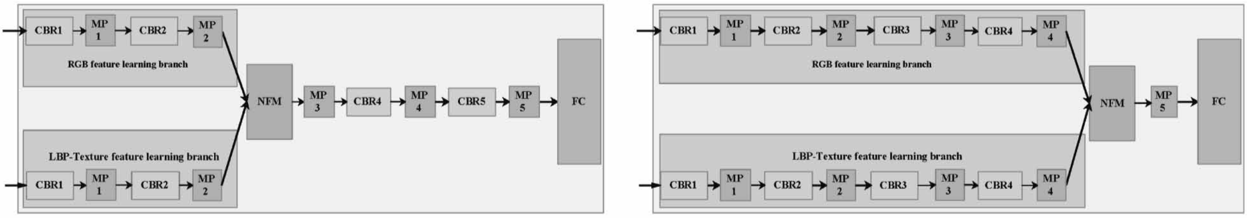
Fig. 2 The diagram of transforming an RGB image into a LBP-texture image

The architecture of the LBP-Texture feature learning branch is kept the same with that of the RGB feature learning branch to reduce model complexities. However, it should be noted that parameters of the 2 branches are independently learned because they are learned on the different input sources. Similar to Ref. [22], this paper learns features on LBP-Texture images, yet this paper designs a novel network that also learns on RGB images. Therefore, the proposed method can receive more information to train a stronger network.

2.2.3 Nonlinear fusion

The RGB feature learning branch and the LBP-Texture feature learning branch are merged into a single branch via the proposed NFM to realize the nonlinear fusion, as shown in Fig. 1. Firstly, for both RGB and LBP-Texture feature learning branches, NFM applies Conv and BN layers to deal with the generated features. Secondly, NFM applies a concatenation (Concat) layer to concatenate features resulting from the 2 branches according to the channel dimension. Moreover, the Concat layer is followed by a bottleneck layer (i. e. , the combination of a conventional layer using 1 ×1 sized filters and a ReLU) to nonlinearly compress the concatenated features. Finally, at the end of NFM, there are MP4, CBR5, MP5 and 3 fully connected layers (i. e. , FC1-FC3). Among these fully connected layers, FC1 and FC2 are followed with batch normalization layers and ReLU activation functions.

Last but not least, it should be pointed out that the NFM is flexible, which means that it can be embedded at different depths of the DFNFN to find the best nonlinear fusion position. As shown in Fig. 3(a), the NFM is embedded between MP2 and MP3 to replace CBR3, while in Fig. 3(b) the NFM is embedded between MP4 and MP5 to replace CBR4. This flexibility allows the proposed DFNFN is more general than the parallel fusion way[25], since the parallel fusion way is just a special case that embeds the NFM at the deepest position.



(a) NFM is embedded between MP2 and MP3 of the deep network (b) NFM is inserted between the MP4 and MP5 layer of the deep network

Fig. 3 The proposed NFM is flexible to be embedded at different depths

2.3 Network configuration

The parameter configuration of the DFNFN is shown in Fig. 1. The channel numbers of CBR1, CBR2, CBR3, CBR5 are 64, 128, 192, 256 and 320, respectively. Except for Conv1 in the NFM, the resting convolutional layers apply 3 × 3 sized filters since smaller sized filters are beneficial to learn local fabric image details. All convolutional layers apply 1-pixel strides to retain fabric image details as much as possible. Moreover, all pooling layers use 3 ×3 sized max-pooling windows and 2-pixel strides.

2.4 Brief description of fabric retrieval implementation

After training the proposed DFNFN network shown in Fig. 2, the fabric retrieval is implemented as follows. Firstly, both probe and gallery images are resized into a unified size, i. e. , 224 ×224, and the applied image resizing method is the commonly-used bilinear interpolation algorithm[27]. Secondly, using the proposed DFNFN network, same dimensional deep features (i. e. , output vectors of the full connection layer (i. e. , FC3) shown in Fig. 1) of probe and gallery images can be extracted. Thirdly, the Euclidean distance between a pair of the probe and gallery images described with the learned deep features is employed to measure the similarity. Finally, based on the similarity measurement, gallery images are ranked to produce candidates for the probe image.

CMC[28] curve shows identification accuracy rates at different ranks. The average precision (AP) is the area under the precision-recall curve of a query, and MAP[29] is the mean value of APs. Hence, the MAP can evaluate the overall fabric retrieval performance.

3 Experimental results and analysis

3.1 Dataset

To validate the superiority of the proposed approach, extensive experiments are conducted on Fabric 1.0[10]. Fabric 1.0 contains 46 656 fabric images of 972 subjects from a garment factory in Quanzhou, China. The class label of each subject is annotated by workers of the factory. Both front and back sides of each subject have been recorded. In order to better simulate actual application scenarios, each side of a subject has been rotated randomly 12 times between 0 and 360 degrees. Moreover, to reduce redundant, each side of a subject is randomly cropped at 2 different spatial positions, and each cropped patch is a 256 × 256 sized RGB image. As a result, each subject contains 48 cropped images. Ref. [10] described the details of Fabric 1.0.

The Fabric 1.0 is separated into non-overlapped training and testing sets for evaluating performance. To be more specific, the half of Fabric 1.0 is applied as the training set and the rest half is used as the testing set. Consequently, both training and testing sets include 23 328 images of 486 subjects. Furthermore, on the testing set, the half of each subject's images is applied to construct a probe subset and the other half is used to form a gallery subset. Therefore, both the probe and gallery subsets of the testing set contains 11 664 images of 486 subjects.

3.2 Implementation

The hardware is an ASUS workstation, containing E5-2650 v2 CPU, 128 GB Memory, and a 12 GB Memory NVIDIA TITAN GPU. The deep learning tool is TensorFlow[30]. The training set of Fabric 1.0 is divided into 2 parts: Part I for training and Part II for validation. The ratio between Part I and Part II is 0.7 : 0.3. During the training process, all images are randomly cropped to 224 × 224 sized patches. The horizontal flip and randomly rotating operations are applied for the data augmentation. Each mini-batch is composed of 64 images that are randomly selected from the Part I of training set. The AlexNet [7] is applied as the backbone network, which is pre-trained on ImageNet[31] for accelerating the training process. The L_2 regularization and dropout techniques are applied to prevent an over-fitting risk, and their hyper-parameters are set as 0.01 and 0.5, respectively. The Adam[32] optimizer is applied to optimize the model, with a learning rate of 10^{-4} and a momentum value of 0.9. At last, the training is finished by using 500 epochs.

For making a clear experiment presentation, in the following experiments, the feature learned by using the proposed NFM is denoted as the nonlinear fusion feature (NFF). Moreover, discarding NFM of Fig. 1, RGB-DF represents deep-learned features produced by only using the RGB feature learning branch, while LBP-DF represents deep-learned features produced by only using the LBP-Texture feature learning branch. In addition, LBP-DF + RGB-DF is the case that directly concatenates LBP-DF and RGB-DF.

3.3 Comparison with state-of-the-art methods

In order to show the superiority of the proposed NFF, it is compared with the state-of-the-art methods, including LBP[2], BoF[33], SIFT + FV[34], SIFT + VLAD[35], LeNet[9], AlexNet[4], and VGGNet[5]. From Table 1, it can be found that the proposed NFF acquires the best performance. Specifically, it outperforms the strongest hand-crafted method, i. e., SIFT + VLAD[35], by a 3.52% higher rank-1 identification rate. Moreover, the proposed NFF also defeats the most powerful deep learning method, i. e., VGGNet[5], by an approximately 7% higher rank-1 identification rate.

To make an intuitive comparison, Fig. 4 shows the top-10 ranked retrieval results of 2 query images. From Fig. 4, it can be found that the proposed NFF works better than multiple state-of-the-art methods, even fabric images are rotated and deflected.

For each query, its top-10 ranked images resulting from LBP, LeNet, AlexNet, VGGNet and NFF are shown, respectively. True matched images are marked with ticks, while false matched images are labeled with crosses.

Table 1 The performance comparison among the proposed method and state-of-the-art methods on Fabric 1.0

Method	Rank = 1	Rank = 5	MAP
HOG[1]	50.89%	63.47%	44.63%
LBP[2]	55.47%	72.62%	45.12%
SIFT + BoF[33]	62.28%	79.74%	46.33%
SIFT + FV[34]	89.86%	91.51%	48.59%
SIFT + VLAD[35]	94.87%	96.54%	50.88%
LeNet[9]	83.59%	94.55%	48.57%
AlexNet[4]	89.58%	98.99%	50.42%
VGGNet[5]	91.14%	98.43%	50.82%
NFF (proposed)	98.39%	99.73%	72.44%

3.4 Analysis

3.4.1 Role of nonlinear fusion module

From Fig. 5, one can find that the proposed NFF consistently outperforms LBP-DF, RGB-DF and LBP-DF + RGB-DF. For rank-1 identification rates, the proposed NFF defeats the LBP-DF and the RGB-DF by 26.93% and 14.59%, respectively. Moreover, NFF beats the LBP-DF and the RGB-DF by 26.87% and 25.08% larger MAPs, respectively. Moreover, one can see that although LBP-DF + RGB-DF concatenating LBP-DF and RGB-DF obtains large improvements, it is still obviously defeated by the proposed NFF. To be more

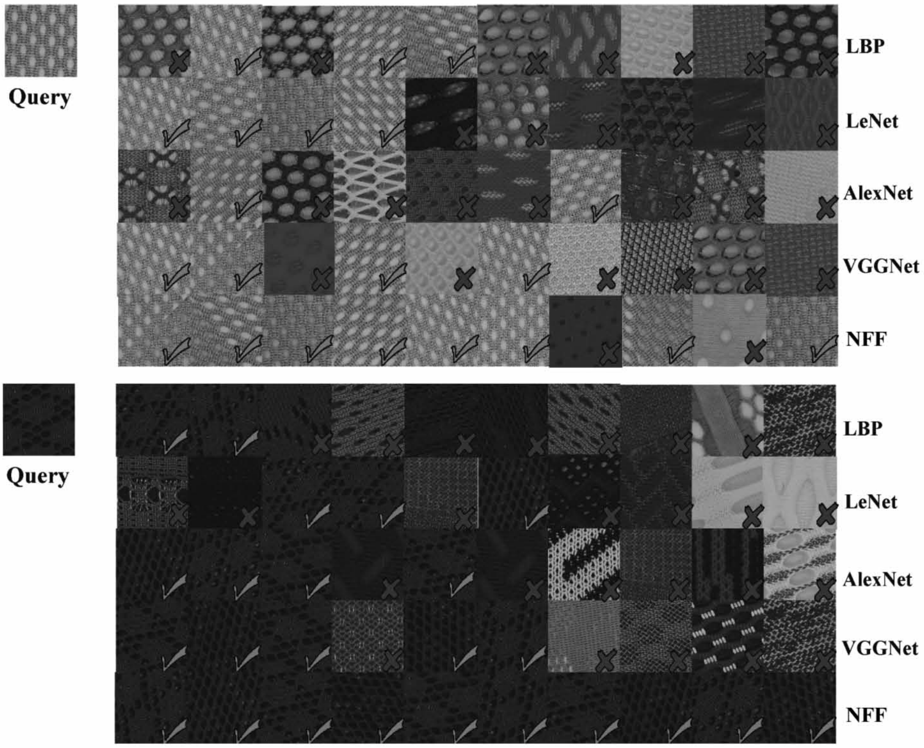


Fig. 4 Examples of retrieval results on Fabric 1.0

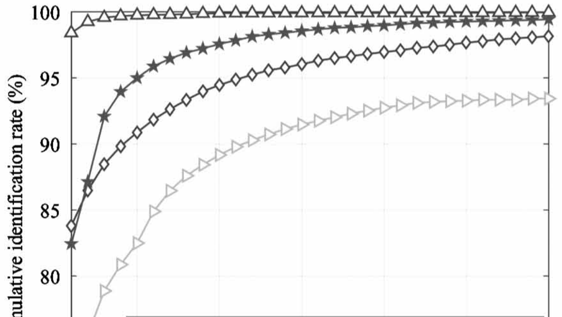


Fig. 5 The CMC curve comparison of LBP-DF, RGB-DF, LBP-DF + RGB-DF and NFF cases on Fabric 1.0 (R1 and R5 represent rank-1 and rank-5 identification rates, respectively)

specific, the rank-1 identification rate of NFF is about 16% higher than that of LBP-DF + RGB-DF.

3.4.2 Comparison of different fusion approaches

In this subsection, different fusion methods are compared to show the superiority of the proposed NFM. Both the serial fusion method[22] and the parallel fusion method[25] are evaluated. For the serial fusion method[22], RGB images are firstly transformed into LBP-Texture images and then the LBP-Texture images are fed to train the deep network. Hence, this serial fusion way is the same as the above-mentioned LBP-DL case. For the parallel fusion method[25], the deep-learned features and hand-crafted features are extracted by the AlexNet and the LBP operation. Then, they are fused via a fully connected layer (i. e., FC1). It should be pointed out that in Ref. [25] the fusion occurs between deep-learned features and weighted HOG features, but in this paper, the weighted HOG features are replaced with LBP-Texture features for making a fair comparison.

As shown in Table 2, the serial fusion method is inferior to the parallel fusion method. This is because the LBP-Texture images containing local illumination variation information but losing color information, which is harmful to the subsequently deep feature learning. In addition, one can see that the proposed NFF method defeats both the serial fusion way and the parallel fusion way, which illustrates the proposed method is an effective fusion method.

Table 2 The comparison of different fusion methods on Fabric 1.0

Method	Rank = 1	Rank = 5	MAP
NFF	98.39%	99.73%	72.44%
Serial fusion[22]	71.46%	82.48%	45.57%
Parallel fusion[25]	95.32%	98.86%	62.44%

In order to explore the contribution of the proposed NFM more clearly, cases embedding NFM from shallow to deep are evaluated and the corresponding results are shown in Fig. 6. For ease of description, CBR1 denotes the case that replaces the original CBR1 block by inserting NFM at the CBR1 position of the deep network, and the rest cases are named the same way. From Fig. 6, it can be observed that the performance fluctuates when NFM is embedded into different positions. The best result is generated by the CBR4 case that embeds NFM to replace the CBR4 block. For example, the CBR4 case acquires a highest MAP (i. e., 72.44%), which is 21.19%, 12.66%, 7.67% and 3.17% higher than CBR1, CBR2, CBR3 and CBR5 cases, respectively. These results show that the optimal feature fusion does not always occur in the deepest position and thus the existing parallel fusion method[25] that forcibly fuses deep-learned and hand-crafted features at the deepest position is not the optimal feature fusion solution.

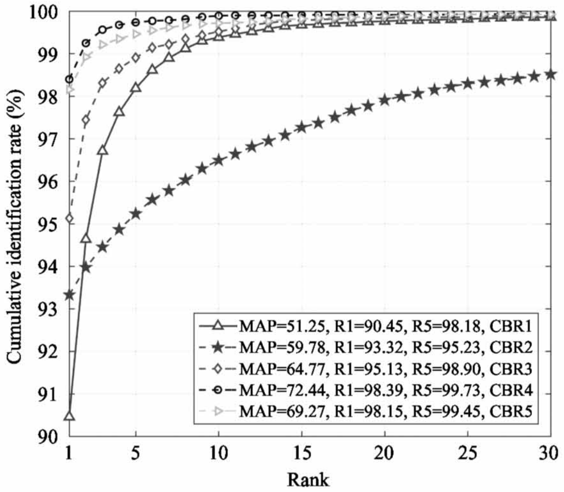


Fig. 6 The CMC curve comparison of cases embedding NFM at different positions

3.4.3 Visual analysis

As shown in Fig. 7, the RGB-DF case and the LBP-DF case exhibit different activation maps and their activation maps can compensate each other to some extent. This illustrates that there is a complementarity between RGB-DF and LBP-DF. Hence, the RGB-DF + LBP-DF case concatenating of RGB-DF and LBP-DF can boost the fabric retrieval performance. In addition, one can find that the proposed NFF method is almost a union of RGB-DF and LBP-DF, which effectively fuses LBP-DF and the RGB-DF to acquire a larger improvement in the fabric retrieval performance.

4 Conclusions

In this work, a novel DFNFN is proposed to nonlinearly fuse features learned from RGB and texture image for fabric retrieval. Texture images are obtained by using LBP-Texture features to describe RGB fabric images. The DFNFN applies 2 feature learning branches to process RGB images and corresponding LBP-Texture images simultaneously. Each branch contains the same convolutional neural network architecture yet independently learning parameters. The features produced by the 2 branches are nonlinearly fused via a special designing NFM. The NFM is so flexible that can be embedded in different depths to find the optimal fusion position of the feature learned from RGB and LBP-Texture images. Consequently, the DFNFN can optimally combine features learned from RGB and LBP-Texture images to promote the fabric retrieval accuracy. Extensive experiments on the Fabric 1.0 dataset show that the proposed method can outperform many state-of-the-art methods.

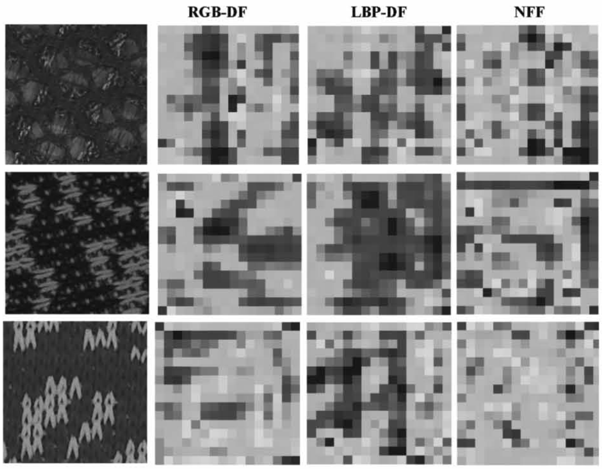


Fig. 7 Activation map (i. e., output of CBR5) visualizations for RGB-DF, LBP-DF and NFF

Reference

[1] Xiao Z, Nie X, Zhang F, et al. Recognition for woven fabric pattern based on gradient histogram[J]. *The Journal of the Textile Institute*. 2014, 105(7): 744-752

[2] Tajeripour F, Kabir E, Sheikhi A. Fabric defect detection using modified local binary patterns[J]. *EURASIP Journal on Advances in Signal Processing*, 2008, doi: 10.1155/2008/783898

[3] Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF)[J]. *Computer Vision and Image Understanding*, 2008, 110(3): 346-359

[4] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in Neural Information Processing Systems, Lake Tahoe, USA, 2012: 1097-1105

[5] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[C]//IEEE Conference on International Conference on Learning Representations, San Diego, Chile, 2015: 1-12

[6] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//IEEE Conference on Computer Vision and

Pattern Recognition, Boston, USA,2015: 1-9

[7] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]// IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016: 770-778

[8] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, USA, 2017: 4700-4708

[9] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. *Proceedings of the IEEE*, 1998, 86(11): 2278-2324

[10] Shen F, Lin L, Wei M, et al. A large benchmark for fabric image retrieval[C]//IEEE Conference on Image, Vision and Computing, Xiamen, China, 2019, doi: 10.1109/ICIVC47709.2019.8981065

[11] Salem Y B, Nasri S. Automatic recognition of woven fabrics based on texture and using SVM[J]. *Signal Image and Video Processing*, 2010, 4(4): 429-434

[12] Ben Salem Y, Nasri S. Automatic classification of woven fabrics using multi-class support vector machine[J]. *Research Journal of Textile and Apparel*, 2009, 13(2): 28-36

[13] Jing J, Li Q, Li P, et al. A new method of printed fabric image retrieval based on color moments and gist feature description[J]. *Textile Research Journal*, 2016, 86 (11): 1137-1150

[14] Suciati N, Herumurti D, Wijaya A Y. Fractal-based texture and HSV color features for fabric image retrieval[C] // IEEE International Conference on Control System, Computing and Engineering, Batu Ferringhi, Malaysia, 2015: 178-182

[15] Yao L, Ke H. Robust image retrieval for lacy and embroidered fabric[J]. *Textile Research Journal*, 2019, 89 (13): 2616-2625

[16] Oliva A, Torralba A. Modeling the shape of the scene: a holistic representation of the spatial envelope[J]. *International Journal of Computer Vision*, 2001, 42(3): 145-175

[17] Liu Z, Luo P, Qiu S, et al. Deep fashion: powering robust clothes recognition and retrieval with rich annotations [C]//IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016: 1096-1104

[18] Xiang J, Zhang N, Pan R, et al. Fabric image retrieval system using hierarchical search based on deep convolutional neural network[J]. *IEEE Access*, 2019, 7: 35405-35417

[19] Bell S, Lawrence Zitnick C, Bala K, et al. Inside-outside net: detecting objects in context with skip pooling and recurrent neural networks[C]//IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016: 2874-2883

[20] Deng D, Wang R, Wu H, et al. Learning deep similarity models with focus ranking for fabric image retrieval[J]. *Image and Vision Computing*, 2018, 70: 11-20

[21] Yue-Hei Ng J, Yang F, Davis L S. Exploiting local features from deep networks for image retrieval[C]//IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, USA, 2015: 53-61

[22] Zhang H, Qu Z, Yuan L, et al. A face recognition method based on LBP feature for CNN[C]//IEEE Conference on Advanced Information Technology, Electronic and Automation Control, Chongqing, China, 2017: 544-547

[23] Levi G, Hassner T. Emotion recognition in the wild via convolutional neural networks and mapped binary patterns [C]// ACM International Conference on Multimodal Interaction, Brisbane, Australia, 2015: 503-510

[24] Bei L, Haizhen Z. An algorithm of fabric image mosaic based on SIFT feature matching[C]//IEEE International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 2009: 435-438

[25] Cai L, Zhu J, Zeng H, et al. HOG-assisted deep feature learning for pedestrian gender recognition[J]. *Journal of the Franklin Institute*, 2018, 355(4): 1991-2008

[26] Ojala T, Pietikainen M, Harwood D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions[C]// IEEE International Conference on Pattern Recognition, Jerusalem, Israel, 1994: 582-585

[27] Seiler M C, Seiler F A. Numerical recipes in C: the art of scientific computing[J]. *Risk Analysis*, 1989, 9(3): 415-416

[28] Gray D, Brennan S, Tao H, Evaluating appearance models for recognition, reacquisition, and tracking[C]// IEEE International Workshop on Performance Evaluation for Tracking and Surveillance, Rio de Janeiro, Brazil, 2007: 1-7

[29] Voorhees E M. Variations in relevance judgments and the measurement of retrieval effectiveness[J]. *Information Processing and Management*, 2000, 36(5): 697-716

[30] Abadi M, Barham P, Chen J, et al. Tensorflow: a system for large-scale machine learning[C]//USENIX Symposium on Operating Systems Design and Implementation, Savannah, USA, 2016: 265-283

[31] Deng J, Dong W, Socher R, et al. A large-scale hierarchical image database[C]// IEEE Conference on Computer Vision and Pattern Recognition, Savannah, USA, 2009: 248-255

[32] Kingma D P, Ba J. Adam: a method for stochastic optimization[J]. *arXiv*:1412.6980, 2014

[33] Jiang Y G, Ngo C W, Yang J. Towards optimal bag-of-features for object categorization and semantic video retrieval[C]// ACM International Conference on Image and Video Retrieval, Amsterdam, Netherlands, 2007: 494-501

[34] Perronnin F, Sánchez J, Mensink T. Improving the fisher kernel for large-scale image classification[C]//European Conference on Computer Vision, Heraklion, Greece, 2010: 143-156

[35] Jégou H, Douze M, Schmid C, et al. Aggregating local descriptors into a compact image representation[C]// IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA, 2010: 3304-3311

Shen Fei, born in 1995. He is currently pursuing the M. S. degree in computer science and technology from the College of Engineering, Huaqiao University, China. His research interests include computer vision and pattern recognition with a focus on the applications of deep learning.