Fabric Retrieval Based on Multi-Task Learning

Jun Xiang, Ning Zhang, Ruru Pan, and Weidong Gao

***Abstract*— Due to the potential values in many areas such as e-commerce and inventory management, fabric image retrieval, which is a special case in Content Based Image Retrieval (CBIR), has recently become a research hotspot. It is also a challenging issue with serval obstacles: variety and complexity of fabric appearance, high requirements for retrieval accuracy. To address this issue, this paper proposes a novel approach for fabric image retrieval based on multi-task learning and deep hashing. According to the cognitive system of fabric, a multi-classification-task learning model with uncertainty loss and constraint is presented to learn fabric image representation. Then we adopt an unsupervised deep network to encode the extracted features into 128-bits hashing codes. Further, the hashing codes are regarded as the index of fabrics image for image retrieval. To evaluate the proposed approach, we expanded and upgraded the dataset WFID, which was built in our previous research specifically for fabric image retrieval. The experimental results show that the proposed approach outperforms the state-of-the-art.**

***Index Terms*— Multi-task learning, fabric retrieval, CBIR, deep hashing, image representation.**

I. Introduction

IMAGE retrieval methods have gone through significant development in the last decade, starting with descriptors based on local-features, first organized in bag-of-words [1] and further expanded by spatial verification [2], hamming embedding [3], and query expansion [4]. Fabric image retrieval, as a special case of generic image retrieval, is a meaningful issue, due to its potential values in many areas such as textile product design, e-commerce, and inventory management.

With the improvement of people's living standards, consumer demands for goods are no longer limited to its practical performance, but tends to be beautiful and diversified. Therefore, the "small- batch and multi-variety" has increasingly become a new production mode for the textile industry. Under this mode of production, companies have accumulated a large amount of historical production data. Image retrieval plays a very important role in product search work. For example, in the textile order production, the processing company will analyze the process parameters of the samples provided by the customer, and then find the visually identical or similar products from the historically produced products. It is very time-consuming and laborious to complete this work manually, and image retrieval can solve this situation.

Given a query, a retrieval system is expected to output a list of images that are similar or related to the query. Traditional keyword-based image retrieval (KBIR) technology is now very mature and widely used in the textile industry. However, the image data in the KBIR dataset is generally labeled by a human. This process couldn't be applied to the retrieval system with a large-scale database. With the reason that the KBIR cannot solve the subjectivity of labeling personal content perception and description, it is more difficult to adapt to the emergence of a large amount of new data. This paper mainly studies fabric image retrieval based on visual content, which is known as Content Based Image Retrieval (CBIR). It remains a challenging task because of the diversity of fabric appearance.

The goal of image retrieval is to find a collection from an image database, where the images have a high similarity to the input query. So, the key to the task is how to quantify the similarity between images, based on human visual perception of the images. Humans visually distinguish two completely different objects mainly through some scattered features, which are difficult to summarize and classify. However, for fabric images with fine texture, distinguishing them often requires analysis from multiple fixed scales and dimensions, such as coarse textures, fine textures, color composition, etc. As shown in Fig 1, the two fabrics are very similar visually, whether viewed from the perspective of coarse texture or color composition (such as Feature 2). However, due to the difference in Feature 1 (fine texture), these two fabrics have completely different properties and uses, and belong to different categories. At present, learning-based image retrieval technologies are mainly driven by a single-classification task to learn image representations, which makes it difficult for these methods to accurately describe fabric images with fine textures. To address this problem, this paper proposes a multi-task learning based framework for fabric image retrieval.

Technically speaking, there are two key components involved in the general method of image retrieval: (1) design a robust feature extraction algorithm for representing images; (2) choose a suitable distance or similarity measure. Recently, a significant breakthrough has been achieved on image analysis by moving from the early low-level feature based hand-crafted algorithms to deep learning based end-to-end framework.

Manuscript received May 11, 2020; revised October 6, 2020 and December 1, 2020; accepted December 5, 2020. Date of publication December 29, 2020; date of current version January 7, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB0309200, in part by the Fundamental Research Funds for the Central Universities under Grant JUSRP52007A, in part by the Postgraduate Research and Practice Innovation Program of Jiangnan University under Grant JNKY19028, and in part by the National Natural Science Foundation of China under Grant 61976105. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Husrev T. Sencar. *(Corresponding author: Weidong Gao.)*

Jun Xiang and Ning Zhang are with the Key Laboratory of Eco-textiles, Ministry of Education, Jiangnan University, Wuxi 214122, China (e-mail: skyjun12@163.com; 15251635269@163.com).

Ruru Pan is with the College of Textile Science and Engineering, Jiangnan University, Wuxi 214122, China (e-mail: prrsw@163.com).

Weidong Gao is with the College of Textile and Engineering, Jiangnan University, Wuxi 214122, China (e-mail: gaowd3@163.com).

Digital Object Identifier 10.1109/TIP.2020.3043877

Even though, the most challenging task is still associating the low-level features based on pixel-level information to high-level semantic features from human perception, which is called "semantic gap". To narrow the "semantic gap", traditional methods usually employ hand-crafted feature descriptors, such as SIFT [5], GIST [6], Bag of Words [1], VLAD [7], and Fisher Vector [8]. Although having achieved certain success in CBIR, these low-level feature based methods depend heavily on feature extraction engineering, which leads to their limitations. It has been demonstrated that the convolutional neural network (CNN) has a very good performance in visual feature description [9]–[12]. Therefore, the most efficient retrieval frameworks in recent years are based on CNNs. Our previous works [9], [13], [14] separately explored the use of hand-crafted features and learning-based features to achieve fabric image retrieval, and has experimentally demonstrated the superiority of learning-based features. However, since the methods based on deep learning requires a lot of computational cost and the dimension of learning-based features is generally high, the retrieval time is more than the methods based on hand-crafted features. There are many researches [15]–[17] that use learning-based coding methods to encode features extracted from deep deep networks. To improve retrieval efficiency without losing the effect of fabric representation, this study adds a latent layer between the classifier and the FC layer of the CNN network to learn the binary coding for representing the fabric.

To the best of our knowledge, there are not public fabric image datasets available for research on image retrieval. We build a large-scale fabric image retrieval dataset including 82,073 fabric images, of which 33,645 images are labeled from four dimensions:n coarse texture, fine texture, color composition and, style. The labeled images in the dataset are used for training and testing the proposed multi-task model. And the whole datasets are applied to verify the performance of the proposed retrieval system.

The rest of this paper is organized as follows. Section 2 review the work related to this study, including multi-task learning and image retrieval. The framework and specific steps of the proposed method are presented in section 3. Section 4 introduces the experimental methods and discusses the results. Finally, section 5 concludes the whole paper.

II. Related Works

According to different technical components, the two topics most to this study are multi-task learning and image retrieval. The two topics will be reviewed in this section.

A. Multi-Task Learning

Multi-task learning (MTL) aims to improve the prediction accuracy and learning efficiency of each task when compared to training a separate model for each task. Caruana [18] has summarized the goal of multi-task learning succinctly:nnnn "MTL improves generalization by leveraging the domain-specific information contained in the training signals of related tasks". MTL can be regarded as an approach of

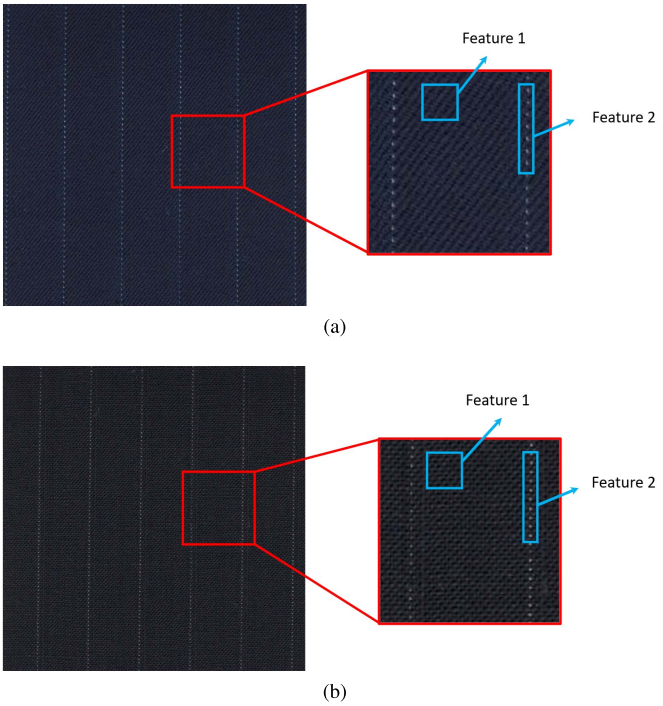


Fig. 1. Examples of similar fabrics in different categories. (a) Striped fabric with twill; (b) Striped fabric with plain.

inductive knowledge transfer, which improves the generalizations ability by sharing domain information between complementary tasks. It does this by using shared representations to learn multiple tasks – what is learned from one task can help learn other related tasks.

Fine-tuning can be considered as a basic example of MTL, where the different learning tasks can be regarded as a pre-training step. However, the purpose of this study is to learn different types of fabric image representations simultaneously, driven by labeled data. Generally, there are two main types of multi-task learning frameworks using deep neural networks:n soft parameter sharing and hard parameter sharing. Each task in the hard parameter has its own model with its own parameter in the soft parameter. As shown in Figure 2(a), soft parameter sharing encourages the model to learn the relevance of each task by adding constraints in the latent layers [19], [20]. However, the soft parameter sharing requires some assumptions to be made in the advance, and the model is easily overfitting. Therefore, this paper proposes using another MTL framework– hard parameter sharing, which greatly reduces the risk of overfitting by sharing some latent layers (while each task has its own independent output) can be seen in Figure 2(b).

In computer vision, there are many examples of methods for MTL, most of which focus on the semantic task, such as classification, semantic segmentation and object detection. Teichmann [21] proposed a MultiNet for detection, classification, and semantic segmentation. Cross-Stitch networks [22] explore methods to combine multi-task neural activations. Uhrig *et al.* [23] learn semantic and instance segmentations under a classification setting. Alex [24] presented a multi-task learning framework using uncertainty to weight losses for scene geometry and semantics. The tasks in this study are

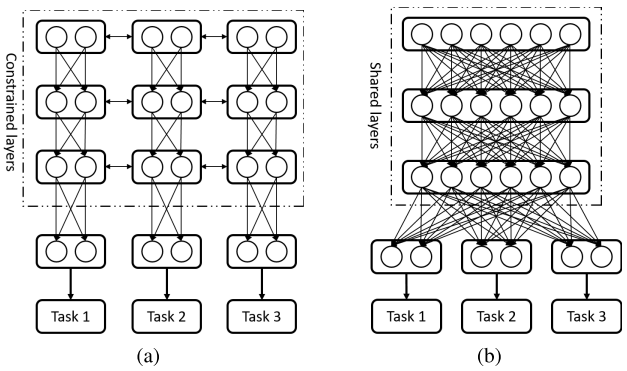


Fig. 2. Two common MTL frameworks (a) Hard parameter sharing (b) Soft parameter sharing.

all classification tasks, which are relatively simple compared to other tasks. We first apply a pre-trained ResNet-50 [25] to represent the fabric image, and then build a multi-task learning framework using a soft parameter sharing mechanism.

B. Fabric Image Retrieval

For fabric retrieval, not only focus on macro features such as color, pattern, and shape, but also focus on micro features such as detailed texture. However, for printed and jacquard fabrics, the pattern is important than its detailed texture, which has led many works [26]–[29] to focus on the retrieval or analysis of their surface patterns. It is for the above reasons that the commonly used image retrieval algorithms often fail to achieve ideal performance on fabric dataset. Technically speaking, clothing retrieval uses more local features to represent images, while fabric retrieval uses the more global features. Moreover, fabric retrieval has strict requirements for search results.

There are three low-level features in the fabric image: color, shape, and texture. Most studies described fabrics from these three aspects, and generally combined these features to retrieval images. Jing *et al.* [30] proposed a method to combine weighted color histogram with image segmentation for pattern fabric retrieval. In a follow-up study [26], Jing presented an algorithm based on the color moment and Gist feature descriptor for printed fabric. Both methods combine the color and shape (all are hand-crafted features) of the fabric to describe the fabric image. Zhang *et al.* [31] proposed a multi-scale and rotation invariant local binary pattern (MRI-LBP) feature which is a texture descriptor for lace fabric image retrieval. Li *et al.* [32] presented a content-based lace fabric image retrieval system using texture (Haralick features) and shape feature. In [33], an image retrieval framework combining color moments and coding features (extracted by perceptual hashing algorithm) is proposed. Nanik *et al.* [34] developed a fabric retrieval system using a combination of fractal-based texture features and HSV features. Cao *et al.* presented a fast fabric retrieval method based on SURF K-means and LSH algorithm. Recently, Zhang *et al.* [13], [14] studied the retrieval of wool fabrics using color and texture features separately, and proposed two methods: a method based on Fourier transform and local binary pattern (Part I) and a method based on dominant colors (DCs) and color moments (CMs) (Part II). Li *et al.* [35] proposed a method with a shape based local affine invariant texture representation for fabric image retrieval. Although having achieved certain success in fabric retrieval, these methods based on hand-crafted features depend heavily on feature extraction engineering which leads to their limitations.

Significant breakthrough has been achieved on image analysis by moving from the early hand-crafted feature based algorithms to deep learning based framework. Therefore, many researchers in the textile industry have tried to apply deep learning techniques to achieve the task of fabric retrieval. Deng *et al.* [36] presented a novel embedding method called focus ranking, which can be easily integrated into CNN to jointly learn image representation and metrics in the context of fine-grained fabric image retrieval. Cai *et al.* [37] apply a triplet convolutional neural network termed Triplet-CNN to learn image representation under the criterion of similarity metric. Xiang *et al.* [9] proposed a deep learning based method which employed a hierarchical search strategy. And the idea of this method is that the binary codes and features for representing the fabric image can be learned by a deep CNN when the data labels are available. Wang *et al.* [38] applied a pre-trained CNN model with the center loss to describe yarn-dyed fabric pattern for fabric retrieval and achieved a good performance. Shen *et al.* [39] introduced a large benchmark for fabric image retrieval and presented the reported baseline performance (contain a non-deep learning method and three deep learning methods). There are two main applications of deep learning in fabric retrieval: image representation learning and similarity matric learning.

In summary, there are two main problems in current researches on fabric image retrieval: 1) many methods [13], [14], [26], [30], [31], [33] are just for a specific type of fabric, making them less adaptable; 2) the existing methods only represent the fabric from one or two dimensions, which is difficult to fully describe the visual characteristics of the fabric.

III. Multi-Task Learning for Image Representation

A. Homoscedastic Uncertainty Loss

Multi-task learning, which is prevalent in many deep learning problems, concerns the problem of optimizing a model with respect to multiple objectives. However, this opens the question of how to combine the losses from each individual task into a single overall loss. Kendall *et al.* [24] have observed that the majority of prior work use a simple sum of the individual loss functions: *nn*

$$L_{total} = \sum_i w_i L_i \tag{1}$$

where the weights, w_i, are fixed hyper-parameters determined through a grid search. Generally, the search for the weights may be very expensive, and may not find the optimal values because it is limited in resolution. The weights might also have varying optimal values at different epochs during training, and so any value of fixed weights may not be optimal for the entire training procedure.

We model the classification task using a Softmax function: n

$$p(y|f^{\mathbf{W}}(x)) = Softmax(f^{\mathbf{W}}(x)) \tag{2}$$

where we regard the output of Softmax as a probability vector from which we sample to get the predicted class. To incorporate heteroscedastic uncertainty in this model we first scale the output of the network as follows: n

$$p(y|f^{\mathbf{W}}(x), \sigma) = Softmax\left(\left(\frac{1}{\sigma^2} f^{\mathbf{W}}(x)\right)\right) \tag{3}$$

Thus the Softmax has the form: n

$$\begin{aligned} p(y = c|f^{\mathbf{W}}(x), \sigma) &= Softmax\left(\frac{1}{\sigma^2} f^{\mathbf{W}}(x)\right) \\ &= \frac{\exp\left(f_c^{\mathbf{W}}(x)\right)}{\sum_{c'} \exp\left(\frac{f_{c'}^{\mathbf{W}}(x)}{\sigma^2}\right)} \end{aligned} \tag{4}$$

$$\begin{aligned} \log(p(y = c|f^{\mathbf{W}}(x), \sigma)) &= \frac{1}{\sigma^2} f_c^{\mathbf{W}}(x) \\ &\quad - \log \sum_{c'} \exp\left(\frac{f_{c'}^{\mathbf{W}}(x)}{\sigma^2}\right) \end{aligned} \tag{5}$$

$p(y|f^{\mathbf{W}}(x), \sigma)$ denotes the probability that the input x belongs to c. c' is a variable used to represent each class. For classification task, we often use the cross-entropy loss function:

$$\log L_{CE}(y = c, \mathbf{W}) = \log\left(\sum_{c'} \exp\left(f_{c'}^{\mathbf{W}}(x)\right)\right) - f_c^{\mathbf{W}}(x) \tag{6}$$

We can rewrite equation (5) in terms of L_{CE}:

$$\begin{aligned} \log(p(y = c|f^{\mathbf{W}}(x), \sigma)) &= -\frac{1}{\sigma^2} log L_{CE}(y = c, \mathbf{W}) \\ &\quad - \log \frac{\sum_{c'} \exp\left(\frac{f_{c'}^{\mathbf{W}}(x)}{\sigma^2}\right)}{\sum_{c'} \exp\left(f_{c'}^{\mathbf{W}}(x)\right)^{\frac{1}{\sigma^2}}} \end{aligned} \tag{7}$$

Kendall *et al.* [24] have noted that: n

$$\exp\left(f_{c'}^{\mathbf{W}}(x)\right)^{\frac{1}{\sigma^2}} \approx \frac{1}{\sigma} \exp\left(\frac{f_{c'}^{\mathbf{W}}(x)}{\sigma^2}\right) \tag{8}$$

Then, the equation (7) can be simplified to the NLL: n

$$-\log\left(p(y = c|f^{\mathbf{W}}(x), \sigma)\right) \approx \frac{1}{\sigma^2} L_{CE} + \log\sigma \tag{9}$$

If our model has multiple classification outputs, then we can combine them as follows: n

$$\begin{aligned} &p(y_1, y2|f^{\mathbf{W}}(x), \sigma_1, \sigma_2) \\ &= Softmax\left(y_1; \frac{1}{\sigma_1} f^{\mathbf{W}}(x)\right) \cdot Softmax\left(y_2; \frac{1}{\sigma^2} f^{\mathbf{W}}(x)\right) \end{aligned} \tag{10}$$

$$\begin{aligned} &\log p(y_1, y2|f^{\mathbf{W}}(x), \sigma_1, \sigma_2) \\ &= \log Softmax\left(y_1; \frac{1}{\sigma_1} f^{\mathbf{W}}(x)\right) \\ &\quad + \log Softmax\left(y_2; \frac{1}{\sigma^2} f^{\mathbf{W}}(x)\right) \end{aligned} \tag{11}$$

Algorithm 1 Multi-Task Learning

Input: Training set: $\mathbf{X}$, Label: $\mathbf{Y}$, learning rate: η, epochs: n
Output: Parameters: $\{\mathbf{W}, s_1, s_2\}$(assume there are two tasks)
Step1 (initialization):
1: initialize $\mathbf{W}$;
2: initialize $s_1 = 0.5, s_2 = 0.5$;
Step2 (Optimization by back propagation):
3: **for** $i = 1$ to n **do**
4: Obtain the gradients according (14), (15), (16);
5: Update $\{\mathbf{W}, s_1, s_2\}$ according (17), (18), (19);
6: Calculate $L(\mathbf{W}, s_1, s_2)$ using (13)
7: **end for**

And the total loss can be denoted: n

$$\begin{aligned} L_{total}(\mathbf{W}, \sigma_1, \sigma_2) &= \frac{1}{\sigma_1^2} L_1(\mathbf{W}) + \frac{1}{\sigma_2^2} L_2(\mathbf{W}) \\ &\quad + \log\sigma_1 + \log\sigma_2 \end{aligned} \tag{12}$$

where L_1, L_2 are the cross entropy losses of each

$$L(\mathbf{W}, s_1, s_2) = e^{-s_1} L_1(\mathbf{W}) + e^{-s_2} L_2(\mathbf{W}) + \frac{1}{2}(s_1 + s_2) \tag{13}$$

To solve this optimization problem, we employ the gradient-based method to learn parameters $\{\mathbf{W}, \mathbf{s_1}, \mathbf{s_2}\}$.The gradient of the objective function in (13) with respect to different parameters are computed as follows: n

$$\frac{\partial L}{\partial s_1} = \frac{1}{2} - e^{-s_1} L_1(\mathbf{W}) \tag{14}$$

$$\frac{\partial L}{\partial s_2} = \frac{1}{2} - e^{-s_2} L_2(\mathbf{W}) \tag{15}$$

$$\frac{\partial L}{\partial \mathbf{W}} = e_{-s_1} \frac{\partial L_1}{\partial \mathbf{W}} + e^{-s_2} \frac{\partial L_2}{\partial \mathbf{W}} \tag{16}$$

The parameters are updated by the following gradient descent algorithm until convergence.

$$s_1 = s_1 - \eta \frac{\partial L}{\partial s_1} \tag{17}$$

$$s_2 = s_2 - \eta \frac{\partial L}{\partial s_2} \tag{18}$$

$$\mathbf{W} = \mathbf{W} - \eta \frac{\partial L}{\partial \mathbf{W}} \tag{19}$$

where η represents the learning rate. Algorithm 1 summarizes the detailed procedure of multi-task learning.

B. Network Architectures

Recently, many deep neural networks have been developed to achieve computer vision tasks such as image classification, object detection. Common network architectures related to the task in study includes AlexNet [40], VGGNet [41], ResNet [25], DenseNet [42], GoogLeNet [43]. Here, we employ ResNet-50 as the stem model in the multi-task learning framework. The reasons are as follows:n

1) The residual network makes it easier for the network to learn identity mapping at certain layers. Performing identity mapping at certain layers is a constructive

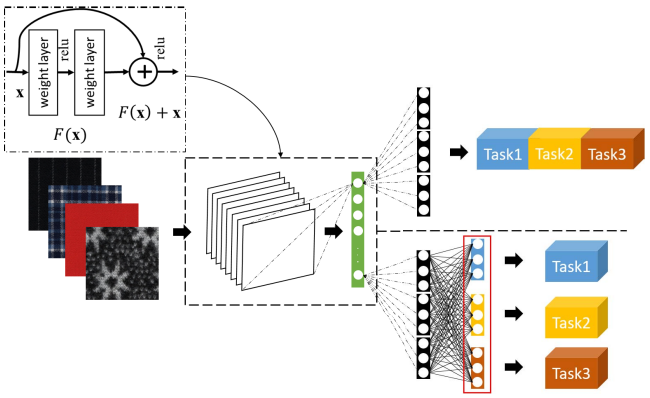


Fig. 3. The framework of multi-task learning. The added tasks are formed by arranging and combining several independent tasks.

solution, so that the performance of deeper models is at least not less than that of shallower models.

2) The residual network behave like ensembles of relatively shallow networks. Residual networks avoid short-lived gradient problems by introducing short paths that can carry gradients over a very deep network.
3) The residual network makes it easier for information to flow between layers, including providing feature reuse during forwarding propagation and mitigating the disappearance of gradient signals during backward propagation.

Concerning on the time and computational cost, the authors modify the building block by using a stack of 3 layers instead of 2. The detailed network architectures can be checked in [25].

For the multiple task learning framework, we propose a new approach to learning the relationship between tasks, as shown in Fig 3, and the approach is only effective for the classification task. For n-classification-task learning, we assume that t_i represents the number of categories in the ith task, then we add another classification task with N categories in the framework. The added task can also be regarded as a constraint that guides the correlation between shared weight learning tasks. By analyzing the data, we find that the number of samples in some categories in the added task is small or not. Our purpose is to back-propagate the joint distribution information of the training-set to the shared weight structure of the model so that it can learn the correlation between different tasks.

$$N = \prod_{i}^{n} t_i \tag{20}$$

In this study, the corresponding multi-task learning model has a total of 4 tasks, and the number of categories corresponding to each task is t1, t2, t3, t4. N represents the number of categories of the added task. The detailed configuration of the network is shown in Table 1. We use the layer before the fully connected layer in res-50 to extract the features of the fabric image with a dimension of 2048. Each fc2_x layer takes as input the output of the fc1 layer. Then all tasks use softmax as the classifier.

TABLE I
DETAILED NETWORK ARCHITECTURES OF MULTI-TASK LEARNING MODEL

layer name	input size	output size
Res Stem	224×224×3	1×1 ×2048
fc1	1×2048	1×1024
fc2_1	1×1024	1×512
softmax_t1	1×512	1×t_1
fc2_2	1×1024	1×512
softmax_t2	1×512	1×t_2
fc2_3	1×1024	1×512
softmax_t3	1×512	1×t_3
fc2_4	1×1024	1×512
softmax_t4	1×512	1×t_4
fc2_1	1×1024	1×512
softmax_t1	1×512	1×t_1

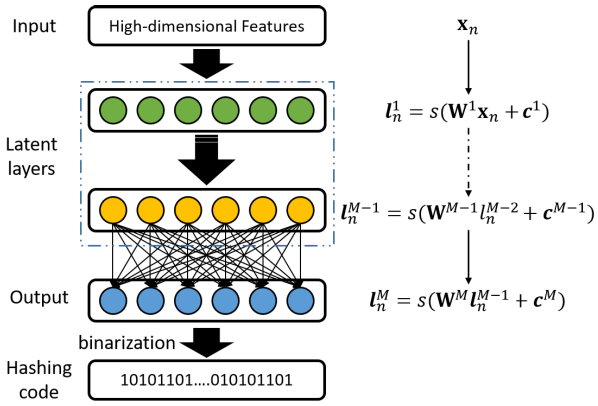


Fig. 4. Network for deep hashing. In this study, we use a 5-layer neural network with the number of nodes {512, 512, 256, 128, 128}.

IV. DEEP HASHING FOR FEATURE AGGREGATION

The function of the multi-task learning model is to extract features for fabric image representation. Generally, thousands or more features can be extracted from a single image. The high-dimensional feature directly used for retrieval will greatly increase the computational cost, thereby reducing the retrieval efficiency. To achieve compact image representation, the high-dimensional features are quantized to a fixed-length vector which is called hashing code.

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2 \cdots \mathbf{x}_N\} \in \mathbb{R}^{d\times N}$ be the training-set which contains N samples, where $\mathbf{x}_n\mathbb{R}^d$ is the nth sample (high-dimensional features) in $\mathbf{X}$. We using a learning-based hashing method (represented by Equation 21) to seek a hash function to map and quantize each sample into a compact binary vector. In Equation (14), $\mathbf{b}_n$ is the binary bit of $\mathbf{x}_n$, $\mathbf{W}$ is the projection matric, and $sgn(\upsilon)$ returns 1 if $\upsilon > 0$ and -1 otherwise.

$$\mathbf{b}_n = sgn(\mathbf{W}^T\mathbf{x}_n) \tag{21}$$

For a given sample $\mathbf{x}_n$, the binary vector $\mathbf{b}_n$ is obtained by a network which contains $M + 1$ fully connected layers

of nonlinear transformations.Assume there are p^m units in the mth layer, where $m = 1, 2 \cdots M$. For the given input $\mathbf{x}_n \in \mathbb{R}^d$, the output of the first layer is:nn $\boldsymbol{l}_n^1 = s(\mathbf{W}^1\mathbf{x}_n + \mathbf{c}^1) \in \mathbb{R}^{p^1}$ where $\mathbf{W}^1$ is the projection matrix (weight) to be learned at the first layer, $\mathbf{c}^1$ is the bias, and $s(\bullet)$ is a nonlinear activation function. Then, the output of first layer is considered as the input for the next layer. Therefore, the output of mth layer can be represented by: $\boldsymbol{l}_n^m = s(\mathbf{W}^m\boldsymbol{l}_n^{m-1} + \mathbf{c}^m) \in \mathbb{R}^{p^m}$, and the output at the top layer of the network is:n $\boldsymbol{l}_n^M = s(\mathbf{W}^M\boldsymbol{l}_n^{M-1} + \mathbf{c}^M) \in \mathbb{R}^{p^M}$.

Let $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2 \cdots \mathbf{b}_N\} \in \{-1, +1\}^{p^M \times N}$ be the matrix representation of the binary codes vectors, and $\mathbf{H}^m = \{\mathbf{h}_1^m, \mathbf{h}_2^m \cdots \mathbf{h}_N^m, \} \in \mathbb{R}^{p^m \times N}$ represents the output of the mth layer of the network. Then the parameters of the network is learned through the following convex optimization problem:

$$\min_{\mathbf{W},\mathbf{c}} L = L_0 - \beta_1 L_1 + \beta_2 L_2 + \beta_3 L_3 \tag{22}$$

$$L_0 = \frac{1}{2}\left\|\mathbf{B} - \mathbf{H}^M\right\|_F^2 \tag{23}$$

$$L_1 = \frac{1}{2N}\mathrm{tr}\left((\mathbf{H}^M\mathbf{H}^M)^T\right) \tag{24}$$

$$L_2 = \frac{1}{2}\sum_{m=1}^{M}\left\|\mathbf{W}^m(\mathbf{W}^m)^T - \mathbf{I}\right\|_F^2 \tag{25}$$

$$L_3 = \frac{1}{2}\left(\left\|\mathbf{W}^M\right\|_F^2 + \left\|\mathbf{c}^M\right\|_F^2\right) \tag{26}$$

The optimization problem consists of 4 parts: L_0, L_1, L_2, and L_3. L_0 denotes the quantization loss between the binary codes and the original real-valued vectors. L_1 aims to maximize the variance of the learned binary codes to ensure balanced bits. The third term L_2 enforces a loose orthogonality constraint on those projection matrices to maximize the independence of each transformation. The last term L_3 is a regularizer that controls the scale of the parameters. β_1, β_2, and β_3 are three parameters used to balance the effects of different terms. To solve this convex optimization problem, we apply the stochastic gradient descent (SGD) method to optimize the parameters $\{\mathbf{W}^m, \mathbf{c}^m\}_{m=1}^M$. This method is unsupervised learning that does not require labeled data for driving. The gradient of the objective function in (22) with respect to different parameters are computed as follows:

$$\frac{\partial L}{\partial \mathbf{W}^m} = \triangle^m\ (\mathbf{H}^{m-1})^T + \beta_2\mathbf{W}^m\left(\mathbf{W}^m(\mathbf{W}^m)^T - \mathbf{I}\right) + \beta_3\mathbf{W}^m \tag{27}$$

$$\frac{\partial L}{\partial \mathbf{c}^m} = \triangle^m + \beta\mathbf{c}^m \tag{28}$$

where

$$\triangle^m = \left((\mathbf{W}_1^{\mathbf{m}+\mathbf{1}^T}\ \triangle^{m+1})\right) \odot s'(\mathbf{Z}^m) \tag{29}$$

$$\triangle^M = \left(-(\mathbf{B} - \mathbf{H}^M) - \beta_1\mathbf{H}^M\right) \odot s'(\mathbf{Z}^M) \tag{30}$$

Here, $\odot$ denotes element-wise multiplication, and$\mathbf{Z}^m = \mathbf{W}^m\mathbf{H}^{m-1} + \mathbf{c}^m$.The parameters are updated by using following gradient descent algorithm until convergence.

$$\mathbf{W}^m = \mathbf{W}_m - \eta\frac{\partial L}{\partial \mathbf{W}^m} \tag{31}$$

$$\mathbf{c}^m = \mathbf{c}_m - \eta\frac{\partial L}{\partial \mathbf{c}^m} \tag{32}$$

Algorithm 2 Deep Hashing

Input: Training set: $\mathbf{X}$, network layer number : M, learning rate: η, epochs: n
Output: Parameters: $\{\mathbf{W}^m, {}^m\}_{m=1}^M$
Step1 (initialization):
initialize $\mathbf{W}^1$ by getting the top p^1 eigenvectors from the covariance matrix;
initialize $\{\mathbf{W}^m\}_{m=2}^M = \mathbf{I}^{p_{m-1}\times p_m}$;
initialize $\{\mathbf{c}^m\}_{m=1}^M = \mathbf{1}^{p_m\times 1}$.
Step2 (Optimization by back propagation):
for $i = 1$ to n **do**
  set $\mathbf{H}^0 = \mathbf{X}$;
  for all m such that $m \in \{M, M-1\cdots 1\}$ **do**
    Obtain the gradients according (26), (27);
    Update $\{\mathbf{W}^m, \mathbf{c}^m\}$ according (30), (31);
    Calculate L using (22)
  end for
end for

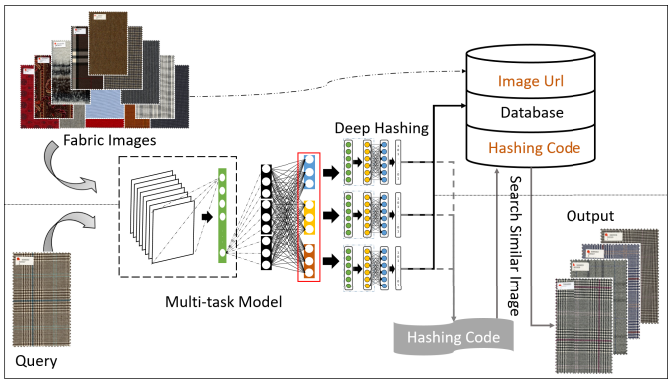


Fig. 5. The proposed retrieval system. The system is divided into two parts, the upper half of the figure is the offline module, the lower half is the online module.

Algorithm 2 summarizes the detailed procedure of deep hashing. In this study, we use four 5-layer neural networks with the number of nodes {512, 512, 256, 128, 128} to respectively encode the extracted high-dimensional features from fc2_1, fc2_2, fc2_3, fc2_4 (Section III. B).

V. The Framework of Retrieval System

Like most CBIR systems, after training the image representation models (multi-task learning model and deep hashing model), we built the fabric image retrieval framework shown in Figure 5. The framework can be divided into two parts: offline module and an online module.

In the offline module, the purpose is to convert the images in the database into hashing codes. For each fabric image, we first input it into the trained multi-task learning model, and extract the last fully connected layer under each task as

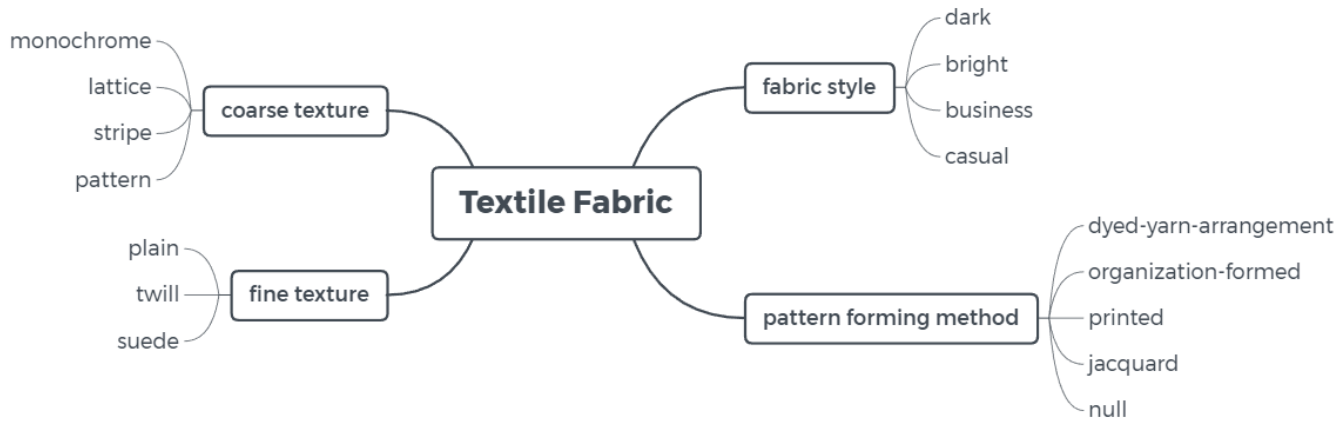


Fig. 6. The structure of the fabric label in this study. We first divide it into four categories based on the appearance of the fabric: Lattice Fabrics. Monochrome Fabrics, Pattern Fabrics, and Stripe Fabrics. The fine texture of all fabrics is simply divided into plain, twill, and suede. The fabric style of monochrome fabrics is divided into dark and bright. The styles of other types of fabrics are divided into casual and business. There is no pattern on plain fabric, so the pattern forming method is null. The patterns in stripe fabrics and lattice fabrics are formed by dyed-yarn-arrangement or organization-formed. However, the pattern fabrics are divided into printed and jacquard.

the image representations. Then, the image representations are input to the deep hashing network for dimensionally reduction and encoding. Finally, the generated hashing code and the URL of the related image are simultaneously stored in the database for retrieval.

For a given query image, we also input it to the multi-task learning model and deep hashing model, and finally output the hashing codes in the online module. Then, the output codes will be compared with the code in the database. The similarity between different codes is measured by Hamming distance which can be denoted by Equation (33).

$$S(x, y) = 1 - \frac{\sum_i^I x_i \oplus y_i}{I} \tag{33}$$

where $S(x, y)$ indicates the similarity between the two codes (x and y), I is the length of the input codes, and $\oplus$ is XOR operator. The similarity between similar codes is close to 1, and the similarity between different codes is close to 0.

Since the framework outputs multiple strings of hashing codes, multiple similarities can be calculated. We use a weighted summation to calculate the total similarity. In the paper, we directly add these similarities to obtain the total similarity.

VI. Experiments

A. Dataset and Implementation

To the best of our knowledge, there are no public datasets for the fabric image retrieval problem. Due to the potential value of fabric image retrieval in many applications, e.g., large-scale fabric searching and online shopping, we expanded and upgraded the dataset WFID built in the previous research [9]. The updated content has three main aspects: (1) the number of fabric samples in the dataset has been expanded to 82,073; (2) 33,645 samples in the dataset have been labeled from four different dimensions for training image representation model; (3) validation set made by experts for evaluating retrieval methods (contains 1029 sets of data, each of which is an image and the 20 most relevant images in the dataset). In particular, it is stated that the annotations between the related images are exactly the same, and their corresponding fabric manufacturing processes are also similar. The original fabric samples are provided by Jiangsu Sunshine Group.[1] The images were captured in a red, green, blue (RGB) model using a scanner (Canon 9000F Mark II). The light source of the scanner was a white light-emitting diode (LED), which can guarantee a stable capture environment. And the resolution was set to 200 dpi.

Humans visually distinguish two completely different objects mainly through some scattered features, which are difficult to summarize and classify. However, for fabric images with fine texture, distinguishing them often requires analysis from multiple fixed scales and dimensions. This study annotates images from four common dimensions of people's recognition of fabrics [44], [45]: coarse texture (classifications: monochrome fabric, lattice fabric, stripe fabric and pattern fabric), fine texture (classifications: plain, twill, and suede), fabric style (classifications: dark, bright, business, and casual), and pattern forming method (dyed-yarn-arrangement, organization-formed, printed, jacquard and null). It is stated that the fabric style in this study only refers to the human visual evaluation of the fabric. The structure of the fabric label is shown in Fig 6. And Fig 7 presents four labeled samples in the dataset. With regard to the classification of coarse texture, monochrome indicates that the fabric is woven with only one color of yarns; lattice refer to a fabric with two or more color yarns in the warp and weft; stripe fabric refers to a fabric in which one of the warp yarn or the weft yarn contains two or more colors of yarns; and the patterns on the surface of pattern fabric are generally more complicated. The classification of the fine texture is based on the interweaving method of the warp and weft of the fabric. The suede fabric refers to the fabric whose structure cannot be distinguished visually. The four tasks involved in this study

[1] http://www.china-sunshine.com/

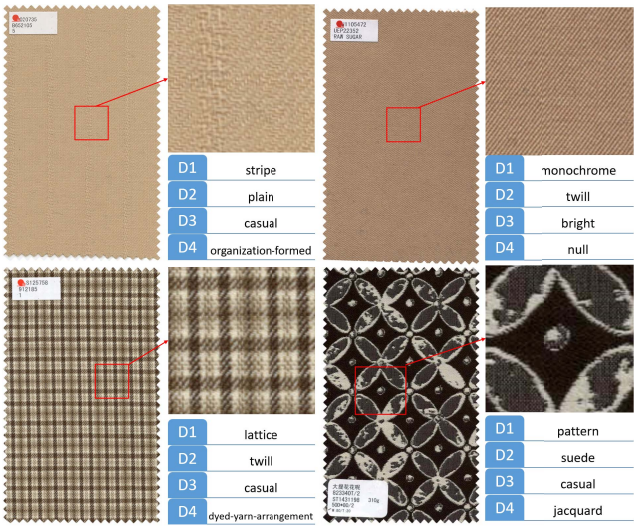


Fig. 7. Four labeled samples in the dataset.

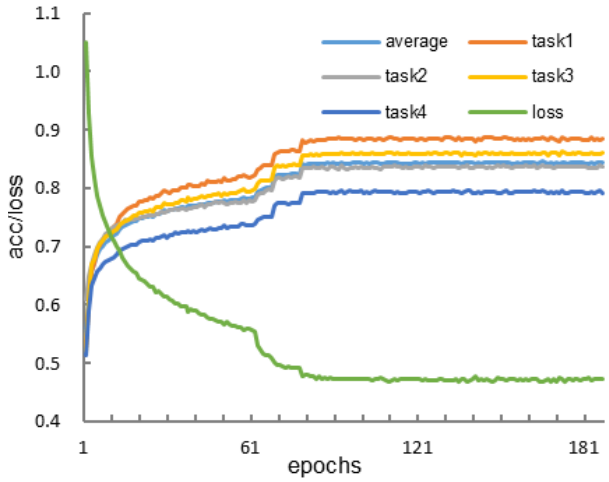


Fig. 8. The curves during training.

are all classification tasks. The goal of task1 is to learn the coarse-texture reprensentation of fabric images. And task2 is to learn the fine-texture representation. Task3 classifies the fabric based on the fabric style. Task4 recognizes the fabric category according to the forming method.

According to the framework of the retrieval system shown in Fig 5, we first train the multi-task learning model using the labeled data. It is stated that we divide the labeled data into two parts: Training-set (80%, for training multi-task learning model) and Testing-set (20%, for testing the multi-task learning model). There are 4 tasks in the multi-task model. To make the parameters of the model have a good initialization state, we use the model pre-trained on the ImageNet to initialize the previous convolutional layers. The hyperparameters during training are as follows: batch_size = 32, weight_decay = 4e-5, learning_rate = 1e-3, learning_rate_decay = 0.1 (every 10 epochs after 60 epochs), and optimizer = SGD. Then we use the output features (with a shape of 1×1024) of each task as the input to train the deep hashing network. After the deep hashing, an image is finally encoded into a binary code with a size of 128. All models are built based on the deep learning framework called Pytorch. In addition, the hardware environment of the model training and retrieval system is as follows: CPUE5-2623 v4 @ 2.60GHz, RAM32G, GPUNVIDIA TITAN XP (11G).

Firstly, we verify the performance of the proposed model in the representation of the fabric image through three experiments, namely: 1) visualize the high-dimensional features extracted by the proposed multi-task learning model using T-SNE; 2) compare the individual model to multi-task learning models using a naïve weighted loss or the task uncertainty weighting with a constraint which we proposed in this study; 3) compare to other state-of-the-art methods in all four tasks. Then we represent the performance of retrieval on fabric dataset, including the precision-recall curves and some retrieval samples. Finally, we compare the proposed retrieval method with other methods (including methods based on manual features and methods based on deep hashing).

B. The Performance of Image Representation

During the training process of the multi-task learning model, the accuracy of each task, the average accuracy of all tasks, and the total loss curve on the testing-set are shown in Fig 8. Similar to the training process of common deep learning models, the proposed model's accuracy indicators show a trend: first rise sharply, then slowly rise, and finally tend to coverage. However, the total loss shows a completely opposite trend. It is worth pointing out that the curves appear several "jump-like" increase or decrease after 60th epoch, which demonstrates that a proper learning_rate_decay for training can help improve the performance of the model. After 90 iterations of training, the model basically converged, and the accuracies and total loss tended to be flat. The final average accuracy reached $84.4\% \pm 0.004$, while the total loss decreased to 0.47 ± 0.02.

To demonstrate the effectiveness of the proposed multi-task learning model, we first visualize the extracted high-dimensional feature using T-SNE [46]. The visualization results are shown in Fig 9. In light of the good separation for the classification task of coarse texture, we should emphasize here that T-SNE is unsupervised, that is, the method received labels and the colors were added after transformation. Since the classification task of coarse texture is the easiest to distinguish visually, the model has the best separation effect for this task. Due to the existence of the "fuzzy zone" between categories of other tasks that are difficult to distinguish, there are cross-links between certain categories. In addition, the reason for these cross-links is also the excessive loss of features during dimensionality reduction. Overall, the proposed multi-task learning model performs well on fabric image representation.

To illustrate the superiority of the proposed multi-task learning model, we conducted comparative experiments. Firstly, we compare the individual models to multi-task learning models using a naïve weighted loss or the task uncertainty weighting with the constraint which we proposed in this study. The experimental results are shown in Table 2. When using fixed weights, adding the constraint proposed in this paper can improve the performance of the model. However,

TABLE II

QUANTITATIVE IMPROVEMENT WHEN LEARNING FOUR TASKS WITH THE PROPOSED LOSS. EXPERIMENTS WERE CONDUCTED ON THE WFID. EXPERIMENTAL RESULTS ARE SHOWN FROM THE VALIDATION SET. WE OBSERVE AN IMPROVEMENT IN PERFORMANCE WHEN TRAINING WITH UNCERTAINTY LOSS AND OUR CONSTRAINT, OVER BOTH SINGLE-TASK MODELS, FIXED WEIGHTED LOSS AND ONLY WITH UNCERTAINTY

Loss	**Task Weights**					**Avg_Acc**	**Task1**	**Task2**	**Task3**	**Task4**
	task1	task2	task3	task4	constraint					
Fixed weighting	0.25	0.25	0.25	0.25	0	81.0%	85.2%	80.3%	82.4%	76.0%
Fixed weighting	0.2	0.2	0.2	0.2	0.2	82.7%	86.7%	81.5%	84.3%	78.2%
Task1 only	1	0	0	0	0	82.9%	87.1%	~	~	~
Task2 only	0	1	0	0	0		~	82.0%	~	~
Task3 only	0	0	1	0	0		~	~	84.5%	~
Task4 only	0	0	0	1	0		~	~	~	78.0%
Uncertainty	✓	✓	✓	✓	✓	83.1%	86.5%	82.8%	84.6%	78.1%
Uncertainty with constraint	✓	✓	✓	✓	✓	84.5%	88.6%	83.9%	86.1%	79.6%

TABLE III

COMPARISON EXPERIMENT RESULTS WITH THE STATE-OF-THE-ART METHODS

Method	**Task1**	**Task2**	**Task3**	**Task4**	**Avg_acc**
DRN[47]	87.9%	81.2%	85.3%	77.4%	83.0%
FAFS[48]	87.3%	82.7%	84.5%	78.3%	83.2%
SCN[22]	88.5%	84.1%	85.7%	77.9%	83.8%
Proposed	88.6%	83.9%	86.1%	79.6%	84.5%

training a multi-task learning model with fixed weights is not as good as training models for each task individually. When we use uncertainty loss for the multi-task learning model, the classification results are improved to 83.1%. Simultaneously, we also demonstrated that uncertainty loss can improve the learning ability of the multi-task model. Finally, Table 1 clearly illustrates the benefit of multi-task learning using uncertainty loss and proposed constraint, which obtains significantly better-performing results than other models.

We also compare to a number of other state of the art methods in all four tasks, including Deep Relationship Networks (DRN) [47], Full-Adaptive Feature Sharing (FAFS) [48] and Cross-Stitch Networks (SCN) [22]. The experimental results shown in Table 3 demonstrate that the multi-task learning methods perform better than individual models. This is because the correlation between the tasks involved in this paper can improve the learning of all tasks. Therefore, grasping the correlation between tasks is the key to multi-task learning. The results once again prove that the added constraint has a certain improvement in the model learning process.

C. The Performance of Fabric Image Retrieval

CBIR systems have typically been evaluated based on mean average precision (mAP), which is computed by sorting images in descending order of relevance per query and averaging AP of individual queries. In addition to mAP, we also employ a modified version of Precision and Recall by considering all query images at the same time, which are given by

$$\text{Precision} = \frac{\sum_q \left|\mathcal{R}_q^{\text{TP}}\right|}{\sum_q \left|\mathcal{R}_q\right|} \tag{34}$$

$$\text{Recall} = \sum_q \left|\mathcal{R}_q^{\text{TP}}\right| \tag{35}$$

where $\mathcal{R}_q$ is a set of retrieved images for query q given a threshold, and $\mathcal{R}_q^{\text{TP}}$ ($\subseteq \mathcal{R}_q$) denotes a set of true positives. Here,

Apart from other image retrievals, fabric retrieval requires higher accuracy of retrieval results. Generally, fabrics with the same label are not necessarily similar fabrics. However, we hold the view that similar fabrics have similar forming processes, so The experts judge whether two fabric images are similarly based on their annotations and process parameters (including fabric structure, yarn arrangement, yarn color, color yarn composition, etc.). So the retrieval task is to search out fabrics with the same label and similar appearance from the database.

To analyze the benefit of the proposed method for fabric image retrieval on WFID, we compare our retrieval framework with its variations: hash-32, hash-64, hash-128, hash-256, and hash-512. Figure 10 presents the Precision-Recall curves of the proposed retrieval framework with different dimensions of hashing. The curves demonstrate that high-dimensional hashing will lead to better results. However, considering the computational cost, we propose to use a 128-bits hashing as the fabric image index, so our later comparative experiments also use 128-bits hashing in the retrieval system. And Figure 11 presents some retrieval results on WFID.

To fairly compare the performance of the hashing method and demonstrate the effectiveness of the proposed hashing method more convincingly, we conduct ablation experiments: 1) directly use the features extracted from the fc layer of the

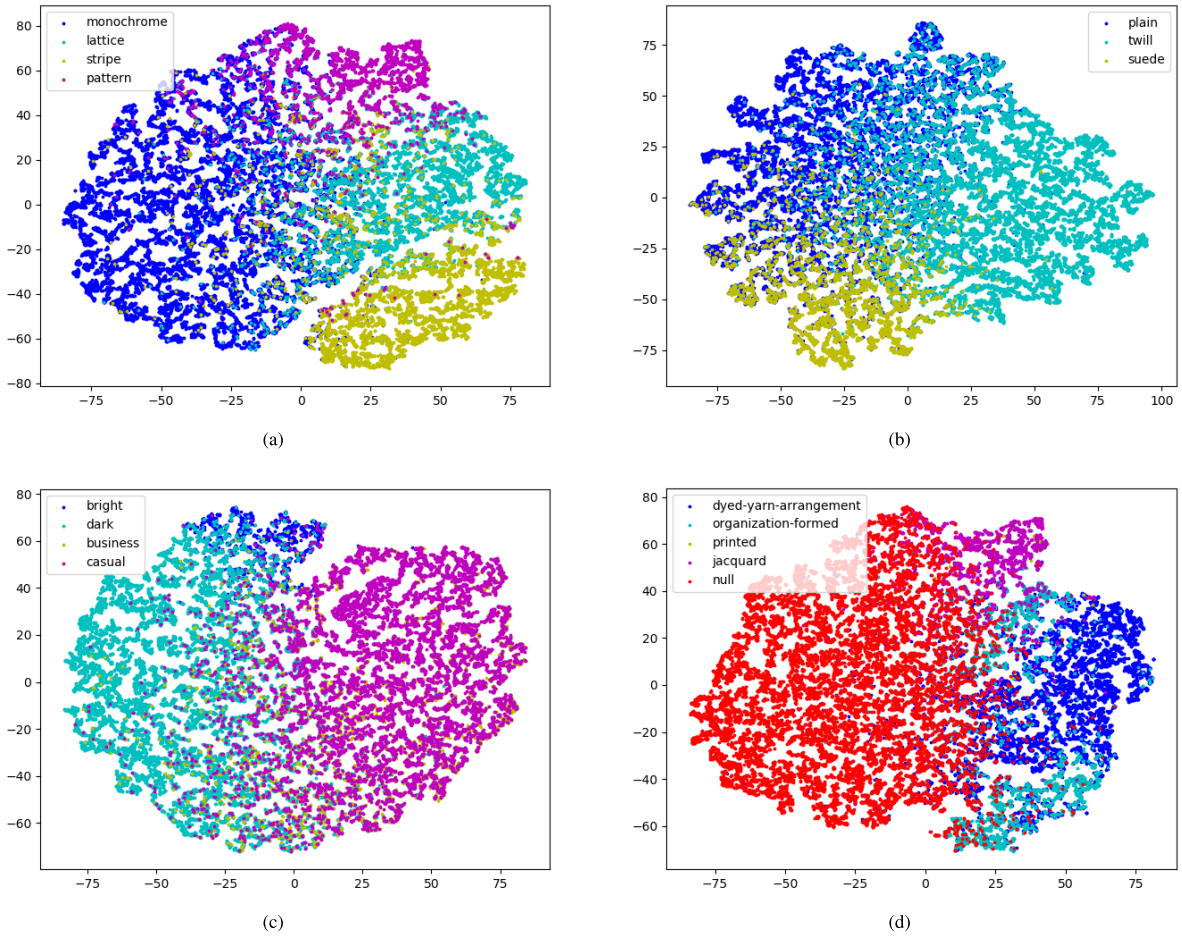


Fig. 9. The visualization results of multi-task learning model (a) The classification task of coarse texture (b) The classification task of fine texture (c) The classification task of fabric style (d) The classification task of pattern forming method.

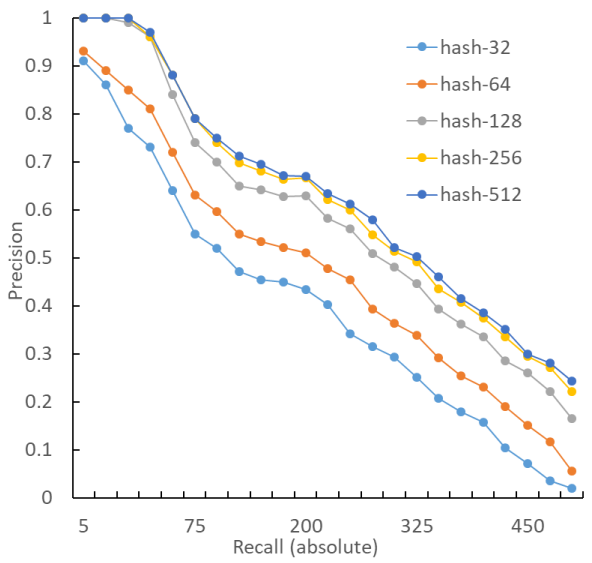


Fig. 10. Precision-Recall curves for the fabric image retrieval on WFID. Here, we regard the samples with exactly the same label and query as positive, otherwise, negative.

multi-task model for retrieval; 2) use the code encoded by the proposed hashing method for retrieval. The results are presented in Tabel 4. Using high-dimensional features extracted from the fc layer as an image index for image retrieval can achieve better performance. However, the computational cost and time cost of these methods are too high to be adopted. The proposed hashing method does not lose much information, since the mAP# Top 20 only drops by about 0.02. In addition, the hashing method greatly reduced retrieval time.

TABLE IV

ABLATION EXPERIMENTS. WE APPLY RESNET-50(BEFORE FC LAYER) AS THE STEM OF THE FOUR MULTI-TASK LEARNING MODEL. ALL EXPERIMENTS ARE CONDUCTED ON THE SAME WORKSTATION

Method		dimension	mAP#Top 20	time/s
without hashing	DRN	4×512	0.927	14.5
	FAFS	2048	0.905	13.7
	SCN	4×512	0.925	14.9
	proposed	4×512	0.941	14.6
with hashing	DRN	4×128	0.908	2.6
	FAFS	512	0.879	2.5
	SCN	4×128	0.913	2.8
	proposed	4×128	0.927	2.6

The proposed retrieval method is compared to several recent retrieval methods. Although there are various research outcomes related to image retrieval, we believe that the following

Fig. 11. Some retrieval examples. We show the top-4 results that are relevant to the query. Not only are the patterns similar between the result and the query, but the fine texture and style are also very similar.

methods are either relevant to our algorithm or most critical to evaluation due to their good performance.

1) Deep Supervised Hashing (DSH) [49]: This is a recent hashing method to learn compact binary codes for highly efficient image retrieval. The key technology of this method is the loss function which punishes similar images mapped to different binary codes and punishes dissimilar images mapped to close binary codes when their Hamming distance falls below the margin threshold. We define that images with identical annotations are similar images.

2) Deep Cauchy Hashing (DCH) [50]: A novel deep hashing model that generates compact and concentrated binary hash codes to enable efficient and effective Hamming space retrieval. The main idea of this method is its pairwise cross-entropy loss based on Cauchy distribution, which penalizes significantly on similar image pairs with Hamming distance larger than the given Hamming radius threshold. When implementing this method, we also define fabric images with identical annotations as similar image pairs.

3) Deep Visual-Semantic Quantization (DVSQ) [51]: This is the first approach to learn deep quantization models from labeled image data as well as the sematic information underlying the general text-domain. The method uses AlexNet to extract a 4096-dimensional deep fc7 feature for each image and adopts the skip-gram model to construct the semantic space. When implementing this method, we treat all annotations as semantic information of the image.

4) Learning Deep Similarity Model (LDSM) [36]: This a novel embedding method termed focus ranking that can be easily unified into a CNN for jointly learning image representations and metrics in the context of fine-grained fabric image retrieval. When implementing this method, we just use the annotations in the coarse texture dimension to training the model.

5) Hierarchical Search Based on Deep CNN (HSDC) [9]: This method employs a hierarchical search strategy that includes coarse-level retrieval and fine-level retrieval. Its image representation adopts labeled data to induce learning based on deep sparse networks. When implementing this method, we only used the annotations in the coarse texture dimension, according to the method proposed by the author.

6) Fabric Retrieval Based on Texture Feature (FRT) [13]: This method is proposed for fabric images based on Fourier transform and Local Binary Pattern (texture feature).

7) Fabric Retrieval Based on Color Features (FRC) [14]: This approach adopts dominant colors (DCs) and color moments (CMs) as the image index for fabric.

We present quantitative results to illustrate the performance of the proposed method compared to the methods listed above. Here we adopt the mAP of the top 20 retrieval results to evaluate different methods. The results are shown in Table 5. Both DSH, LDSM and DCH use pairs of similar or dissimilar images to guide learning image representation and coding. This way makes it difficult for the model to learn the knowledge system of the fabric. Their retrieval results are very similar, but it is difficult to meet actual needs. DVSQ performs very well because it makes full use of annotation information. The retrieval process of HSDC is divided into two steps: coarse-level retrieval uses binary codes (hash codes) to search candidate sets, and fine-level retrieval uses fully connected layer features to filter similar images. Therefore, the results are less affected by the size of hashing. In addition, FRT and FRC just take the color or texture into account, so that their accuracy is much lower than other methods. The quantitative results demonstrate that the proposed method outperforms all other techniques significantly.

TABLE V
COMPARISON EXPERIMENT RESULTS WITH THE STATE-OF-THE-ART RETRIEVAL METHODS

Methods	mAP#Top 20				
	32bits	64bits	28bits	256bits	512bits
Proposed	0.875	0.892	0.947	0.958	0.965
DSH [49]	0.821	0.834	0.851	0.891	0.912
DCH [50]	0.815	0.834	0.845	0.897	0.913
DVSQ [51]	0.842	0.861	0.882	0.912	0.931
LDSM [36]	0.715	0.791	0.812	0.832	0.851
HSDC [9]	0.827	0.831	0.837	0.839	0.842
FRT [13]	0.721				
FRC [14]	0.716				

VII. CONCLUSION

In this paper, a novel method for fabric image retrieval based on a multi-task learning model was presented. The model aligns its objective with the human perception of fabric, namely, coarse texture, fine texture, fabric style, and pattern forming method. According to the classification of fabric, a multi-classification-task learning model with uncertainty loss and constraint was proposed to learn fabric image representation. The comparison experiments verified the effectiveness of the proposed method. Then we applied an unsupervised deep network to encode the extracted features into 128-bits hashing codes. Finally, the hashing codes was regarded as the index of fabric image for image retrieval. The experimental results demonstrated that our method outperforms than the state-of-the-art.

ACKNOWLEDGMENT

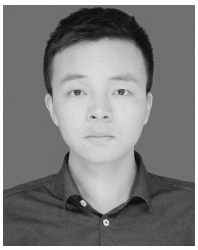
The authors would like to thank Jiangsu Sunshine Group for providing data for this research.

REFERENCES

[1] Sivic and Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, p. 1470.
[2] Y.-H. Kuo, K.-T. Chen, C.-H. Chiang, and W. H. Hsu, "Query expansion for hash-based image object retrieval," in *Proc. 17 ACM Int. Conf. Multimedia MM*, 2009, pp. 65–74.
[3] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2008, pp. 304–317.
[4] H. Xie, Y. Zhang, J. Tan, L. Guo, and J. Li, "Contextual query expansion for image retrieval," *IEEE Trans. Multimedia*, vol. 16, no. 4, pp. 1104–1114, Jun. 2014.
[5] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2. Sep. 1999, pp. 1150–1157.
[6] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.
[7] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3304–3311.
[8] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.
[9] J. Xiang, N. Zhang, R. Pan, and W. Gao, "Fabric image retrieval system using hierarchical search based on deep convolutional neural network," *IEEE Access*, vol. 7, pp. 35405–35417, 2019.
[10] T. Zhao *et al.*, "Embedding visual hierarchy with deep networks for large-scale visual recognition," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 4740–4755, Oct. 2018.
[11] L. Wang, X. Qian, Y. Zhang, J. Shen, and X. Cao, "Enhancing sketch-based image retrieval by CNN semantic re-ranking," *IEEE Trans. Cybern.*, vol. 50, no. 7, pp. 3330–3342, Jul. 2020.
[12] J. Fan, N. Zhou, J. Peng, and L. Gao, "Hierarchical learning of tree classifiers for large-scale plant species identification," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4172–4184, Nov. 2015.
[13] N. Zhang, J. Xiang, L. Wang, W. Gao, and R. Pan, "Image retrieval of wool fabric. Part I: Based on low-level texture features," *Textile Res. J.*, vol. 89, nos. 19–20, pp. 4195–4207, Oct. 2019.
[14] N. Zhang, J. Xiang, L. Wang, N. Xiong, W. Gao, and R. Pan, "Image retrieval of wool fabric. Part II: Based on low-level color features," *Textile Res. J.*, vol. 90, nos. 7–8, pp. 797–808, Apr. 2020.
[15] L. Xie, J. Shen, J. Han, L. Zhu, and L. Shao, "Dynamic multi-view hashing for online image retrieval," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 3133–3139.
[16] L. Zhu, J. Shen, L. Xie, and Z. Cheng, "Unsupervised visual hashing with semantic assistant for content-based image retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 2, pp. 472–486, Feb. 2017.
[17] H. Zhai, S. Lai, H. Jin, X. Qian, and T. Mei, "Deep transfer hashing for image retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Apr. 29, 2020, doi: 10.1109/TCSVT.2020.2991171.
[18] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.
[19] L. Duong, T. Cohn, S. Bird, and P. Cook, "Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process. (Short Papers)*, vol. 2, Jul. 2015, pp. 845–850.
[20] Y. Yang and T. Hospedales, "Deep multi-task representation learning: A tensor factorisation approach," 2016, *arXiv:1605.06391*. [Online]. Available: http://arxiv.org/abs/1605.06391
[21] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time joint semantic reasoning for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1013–1020.
[22] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3994–4003.
[23] J. Uhrig, M. Cordts, U. Franke, and T. Brox, "Pixel-level encoding and depth layering for instance-level semantic labeling," in *Proc. German Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2016, pp. 14–25.
[24] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7482–7491.
[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[26] J. Jing, Q. Li, P. Li, and L. Zhang, "A new method of printed fabric image retrieval based on color moments and gist feature description," *Textile Res. J.*, vol. 86, no. 11, pp. 1137–1150, 2015.
[27] J.-L. Liu and B.-Q. Zuo, "Segmentation of pattern of printed fabric based on genetic algorithm," *Comput. Eng. Des.*, vol. 15, pp. 3965–3967, Aug. 2008.
[28] C.-F. J. Kuo and C.-Y. Shih, "Printed fabric computerized automatic color separating system," *Textile Res. J.*, vol. 81, no. 7, pp. 706–713, 2011.
[29] H. Y. T. Ngan, G. K. H. Pang, S. P. Yung, and M. K. Ng, "Defect detection on patterned jacquard fabric," in *Proc. 32nd Appl. Imag. Pattern Recognit. Workshop*, Oct. 2003, pp. 163–168.
[30] J. Jing, "Patterned fabric image retrieval using color and space features," *J. Fiber Bioeng. Informat.*, vol. 8, no. 3, pp. 603–614, Jun. 2015.
[31] L. Zhang, X. Liu, Z. Lu, F. Liu, and R. Hong, "Lace fabric image retrieval based on multi-scale and rotation invariant LBP," in *Proc. 7th Int. Conf. Internet Multimedia Comput. Service ICIMCS*, 2015, pp. 1–5.
[32] Y. Li, H. Luo, G. Jiang, and H. Cong, "Content-based lace fabric image retrieval system using texture and shape features," *J. Textile Inst.*, vol. 110, no. 6, pp. 911–915, Jun. 2019.
[33] Z. Li, J. Xiang, L. Wang, N. Zhang, R. Pan, and W. Gao, "Yarn-dyed fabric image retrieval using colour moments and the perceptual hash algorithm," *Fibres Textiles Eastern Eur.*, vol. 137, no. 5, pp. 39–46, Aug. 2019.
[34] N. Suciati, D. Herumurti, and A. Y. Wijaya, "Fractal-based texture and HSV color features for fabric image retrieval," in *Proc. IEEE Int. Conf. Control Syst., Comput. Eng. (ICCSCE)*, Nov. 2015, pp. 178–182.

[35] Y. Li, J. Zhang, M. Chen, H. Lei, G. Luo, and Y. Huang, "Shape based local affine invariant texture characteristics for fabric image retrieval," *Multimedia Tools Appl.*, vol. 78, no. 11, pp. 15433–15453, Jun. 2019.
[36] D. Deng, R. Wang, H. Wu, H. He, Q. Li, and X. Luo, "Learning deep similarity models with focus ranking for fabric image retrieval," *Image Vis. Comput.*, vol. 70, pp. 11–20, Feb. 2018.
[37] Z. Cai, W. Gao, Z. Yu, J. Huang, and Z. Cai, "Feature extraction with triplet convolutional neural network for content-based image retrieval," in *Proc. 12th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Jun. 2017, pp. 337–342.
[38] X. Wang, G. Wu, and Y. Zhong, "Fabric identification using convolutional neural network," in *Proc. Int. Conf. Artif. Intell. Textile Apparel*. Cham, Switzerland: Springer, 2018, pp. 93–100.
[39] F. Shen *et al.*, "A large benchmark for fabric image retrieval," in *Proc. IEEE 4th Int. Conf. Image, Vis. Comput. (ICIVC)*, Jul. 2019, pp. 247–251.
[40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[41] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[42] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.
[43] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[44] I. Emery, "The primary structures of fabrics, an illustrated classification," AATA, Washington, DC, USA, Tech. Rep. 25-1937, 1980.
[45] A. Song, Y. Han, H. Hu, and J. Li, "A novel texture sensor for fabric texture measurement and classification," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 7, pp. 1739–1747, Jul. 2014.
[46] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
[47] M. Long, Z. Cao, J. Wang, and P. S. Yu, "Learning multiple tasks with multilinear relationship networks," 2015, *arXiv:1506.02117*. [Online]. Available: http://arxiv.org/abs/1506.02117
[48] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, and R. Feris, "Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5334–5343.
[49] H. Liu, R. Wang, S. Shan, and X. Chen, "Deep supervised hashing for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2064–2072.
[50] Y. Cao, M. Long, B. Liu, and J. Wang, "Deep cauchy hashing for Hamming space retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1229–1237.
[51] Y. Cao, M. Long, J. Wang, and S. Liu, "Deep visual-semantic quantization for efficient image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1328–1337.

Jun Xiang received the master's degree in textile engineering from Jiangnan University, Wuxi, China, where he is currently pursuing the Ph.D. degree in textile science and engineering with the College of Textile Science and Engineering. He is interested in image analysis, textile measurement, machine learning, and intelligent manufacturing.



Ning Zhang received the B.S. degree in textile engineering from Jiangnan University, Wuxi, China, in 2016, where he is currently pursuing the Ph.D. degree with the College of Textile Science and Engineering. His current research interests are in interactive machine learning, deep learning, and its application in textile and garment industry.

Ruru Pan received the B.S. and Ph.D. degrees in textile engineering from Jiangnan University, Wuxi, China, in 2005 and 2010, respectively. He is currently an Associate Professor with the College of Textile Science and Engineering, Jiangnan University, Wuxi, China. His current research interests include digital textile technology and digital image processing of textile.

Weidong Gao received the B.S. and M.S. degrees in textile engineering from the Wuxi Institute of Light Industry, Wuxi, China, in 1982 and 1985, respectively, and the Ph.D. degree in textile engineering from Donghua University, Shanghai, China, in 2011. He is currently a Full Professor with the College of Textile Science and Engineering, Jiangnan University, Wuxi, China. His current research interests include intelligent textile technology, intelligent weaving, and digital image processing of textile.