http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (7): 1387-1393, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.1387.1393

Gender Classification Based on Relaxed Pixel-Pattern-Based Texture Feature

[1]Bo Chunjuan, [2]Zhang Junxing, [1]Liu Changhong and [1]Sun Yanhui
[1]College of Electromechanical and Information Engineering,
[2]College of Information and Communication Engineering, Dalian Nationalities University,
Dalian, 116600, China

Abstract: Gender classification has been investigated from both psychological and computational perspectives used for face recognition, identification, etc. Any interesting methods may benefit some real applications with not bad performance but developing a more accurate and robust computational algorithm for gender classification remains a challenging open problem. This study presents a novel face image gender classification algorithm based on Support Vector Machine (SVM) and the proposed Relaxed Pixel-Pattern-Based Texture Feature (RPPBTF). For extracting the RPPBTF, the image basis functions are adopted which are obtained by Principal Component Analysis (PCA) as a set of templates for obtaining texture patterns. Compared to the traditional PPBTF method, the RPPBTF algorithm uses a soft operator rather than a hard operator for extracting texture features. This modification makes that the RPPBTF method not only maintains the advantage of the original PPBTF method (e.g., insensitive to illumination variations) but also extracts much texture information without much computational load. Experimental results and discussions demonstrate that the proposed method is better than other popular features for the gender classification task.

Key words: Gender classification, texture feature, PCA, PPBTF, RPPBTF

INTRODUCTION

Face image gender classification is an attractive and challenging issue in the field of computer vision. A successful gender classification method has a wide range of applications, including human-computer interface, human identification, security, education and telecommunication, etc.

Earlier, the gender classification works were based on neural network and raw pixel features. Three typical neural networks were trained to identify gender with different pixels, resolutions and parts of face images which are two-layered neural network SEXNET (Golomb *et al*., 1991), multi-layered neural network (Tamura *et al*., 1996) and linear neural network (Edelman *et al*., 1998). However, all these methods require large amount of training samples. In addition, their recognition speeds are quite slow. After that, the researchers adopted different machine learning algorithms to solve the gender classification problem. Support Vector Machine (SVM) was used to classify gender with low-resolution 12×12 "thumbnail" faces (Moghaddam and Yang, 2000). A gender classification method was developed by using a boosting method (Baluja and Rowley, 2007).

Although, these raw pixel-based methods achieved not bad performance, they were sensitive to many challenging factors (e.g., illumination, pose and skin variations). Thus, the texture-based methods attracted many attentions. The LBP (local binary patterns) feature was applied to extract the texture feature efficiently and then conducted to gender recognition (Shan, 2012). To overcome the disturbance of illumination variations, 2D Gabor transform (Lee, 1996) was adopted to extract the face features and a SVM classifier was trained to achieve good recognition results on a relative large-scale and low-resolution face database (Wang *et al*., 2008). Pixel-Pattern-Based Texture Feature (PPBTF) was boosted to select the most discriminate features subset and a SVM classifier was adopted for final recognition (Lu *et al*., 2008). Compared to Gabor, PPBTF is very fast and retains enough facial information in a compact representation.

In this study, a novel texture feature, Relaxed Pixel-Pattern-Based Texture Feature (RPPBTF) is presented for gender classification. Figure 1 illustrates the flow chart of the gender classification system in this work. First, some training face images with binary labels are collected and PCA templates are trained based on image patches from these image samples. Second, based on the trained PCA templates, texture features are extracted by using the proposed RPPBTF algorithm. Then a SVM

Corresponding Author: Zhang Junxing, College of Information and Communication Engineering, Dalian Nationalities University, Dalian, 116600, China

classifier with the extracted features and their corresponding labels are learned. For a given test image, its RPPBTF description is extracted firstly; and then the trained SVM is used for classification. Compared to the original PPBTF, the proposed RPPBTF uses a soft operator for extracting texture features which alleviates information loss without introducing much computational load. In addition, many experiments are conducted on the FERET (Phillips *et al*., 2000) face database to compare the results of different algorithms and discuss the effects of different parameters (i.e., the number of templates and the size of blocks). The experimental results demonstrate that RPPBTF achieves better and more stable performance than other popular features.

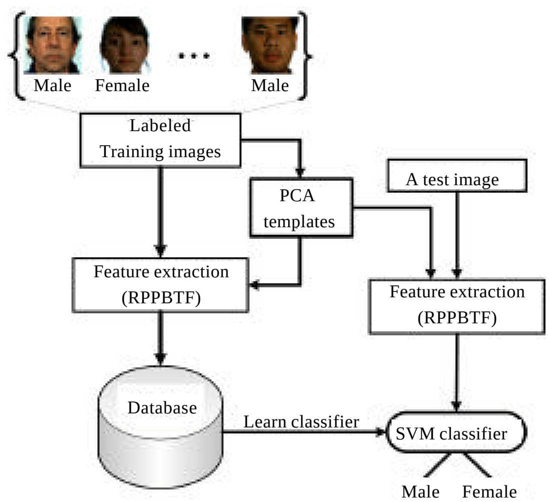

Fig. 1: The flow chart of gender classification

BACKGROUND AND RELATED WORK

Here, some background knowledge is briefly provided and reviewed to make this study self-contained which includes Local Binary Patterns (LBP), Gabor functions and Pixel-Pattern-Based Texture Feature (PPBTF).

Local binary patterns (LBP): Local Binary Pattern (LBP) is a kind of simple yet efficient operator for gray scale and rotation invariant texture representation. There are two main advantages on the LBP features. One is very robust in terms of gray-scale variations as the operators are by definition invariant against any monotonic transformation of the gray-scale. The other is computational simplicity, since the operator can be realized with a few operations in a small neighborhood and a lookup table which facilitates a very straightforward and efficient implementation. The LBP operator $LBP_{8,1}$ produces 2^P different output values, corresponding to the different binary patterns that can be formed by the P pixels in the neighbor set with the radius R($LBP_{8,1}$ is the most common operator). In addition, the concept of "Uniform" was introduced which is corresponding to the binary coding string that changes between "0" and "1" less than twice (Ojala *et al*., 2002). Figure 2 illustrates the examples of the LBP operator. Figure 2a-d shows some "Uniform" patterns; while Fig. 2f demonstrates a "Non-uniform" case. As mentioned in (Ojala *et al*., 2002), the "Uniform" patterns were recognized to be a fundamental property of texture as they provide a vast majority of local texture patterns in examined textures, corresponding to texture microstructures such as edges.

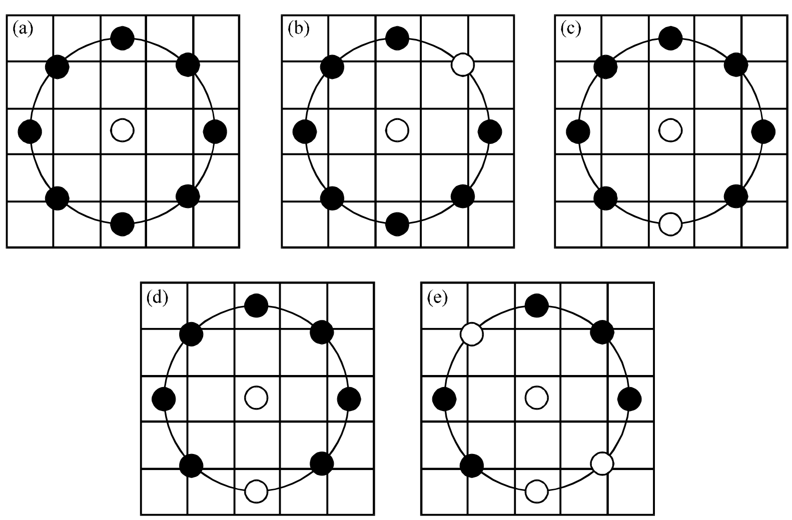

Fig. 2(a-f): Samples of the LBP operators

Usually, the LBP feature histogram can be adopted to represent the texture features. For face recognition (Ahonen *et al*., 2004) and gender classification problems, the face image is first divided into small regions from which LBP features are extracted and concatenated into a single, spatially enhanced feature histogram. This block-based LBP histogram represents the face image efficiently, as it considers both shape and texture information.

Gabor functions: The Gabor functions are demonstrated to simulate the responses of the visual cortex and widely used for image understanding, recognition, etc. Gabor is a complex sine function modulated by Gaussian function which extracts local frequency feature within a specific region. The 2D Gabor core function is defined as the product of an elliptical Gaussian envelope and a complex plane wave:

$$\varphi_{u,v}(x,y)=\frac{k^2}{\sigma^2}\exp\left(-\frac{k^2(x^2+y^2)}{2\sigma^2}\right)\cdot\left[\exp^{i(k\cdot(x,y)^T)}-\exp\left(-\frac{\sigma^2}{2}\right)\right] \quad (1)$$

where, i is a complex operator, (x, y) is a column vector that specifies the vector norm, parameter σ/k decides the size of Gaussian window and let σ equal $\sqrt{2}\pi$, k is the wave vector, different k will consist different wavelet function, k defines as:

$$k_v=\begin{pmatrix}k_x\\k_y\end{pmatrix}=\begin{pmatrix}k_v\cos\theta_u\\k_v\sin\theta_u\end{pmatrix}\text{ and }k_v=2^{\frac{-v+2}{2}}\pi,\ \theta_u=\frac{u\pi}{k}$$

frequency coefficient v determines Gabor filter wavelength (it is also called scale) while direction coefficient u decides the orientation of Gabor core function, k_v is filter central frequency. θ stands for different direction. Usually, 40 Gabor wavelets (also called filter banks) are used for feature extraction which consist of 5 scales (v = 0, 1, 2, 3, 4) and 8 directions (θ = 0, $\pi/8$, $\pi/4$, $3\pi/8$, $\pi/2$, $5\pi/8$, $3\pi/4$, $7\pi/8$).

The Gabor feature extraction is realized by using a convolution computation between face image and each of 40 different Gabor core functions, respectively which is:

$$G=\varphi_{u,v}\times I(x,y)=\iint\varphi_{u,v}I(x,y)dxdy \quad (2)$$

where, I(x, y) is gray-scale image. Since, the convolution results are complex, their modulus can be regarded as Gabor features. Although, the Gabor features achieve very promising performance on image understanding and classification, both computational complexity and feature dimension are very high. For solving this problem, a Gabor (sum) feature was presented which is the sum over scales and directions of Gabor functions based representation (Tao *et al*., 2007). They showed that the Gabor (sum) features achieve slightly worse performance than but significantly less computational costs than the traditional Gabor features.

Pixel-pattern-based texture feature (PPBTF): Principal Component Analysis (PCA) is a classical dimension-reduction method and has many real applications (e.g., face recognition (Turk and Pentland, 1991), visual tracking (Wang *et al*., 2013), image understanding (Heidemann, 2006) and classification (Lu *et al*., 2008). Especially, many researchers have found the theoretical and empirical significance of PCA for feature extraction and pattern matching (Wang *et al*., 2011).

Empirical results that PCA basis functions of natural images can be viewed as fundamental patterns of human's receptive field as they contain edge-like, bar-like, grating-like and noise patterns. Thus, they can be used to extract texture features. Compared to the texture descriptors that are designed mathematically, they are obtained statistically so that they take image prior into consideration. Pixel-Pattern-Based Texture Feature (PPBTF) used the PCA basis function as templates was proposed for pattern matching which are proved quite time saving and free of the influence of illumination (Zeng *et al*., 2004). PPBTF and a SVM classifier were used to design a gender recognition system (Lu *et al*., 2008). Some detailed information can be found in next section, where the difference between PPBTF and RPPBTF is highlighted.

RELAXED PIXEL-PATTERN-BASED TEXTURE FEATURE (RPPBTF)

PCA templates: Both the PPBTF and proposed RPPBTF methods use the PCA basis functions as texture patterns. Figure 3 shows the flowchart of the computation processing of PCA templates. The main processes include four steps:

Step 1: Image patches of S×S pixels are sampled at random positions from K natural gray images (training set), as shown in Fig. 3a

Step 2: For each image, D image patches are selected and then converted them into column vectors. The collected training set is denoted as X = [x_1, x_2,..., $x_{K\times D}$]. Remove the mean:

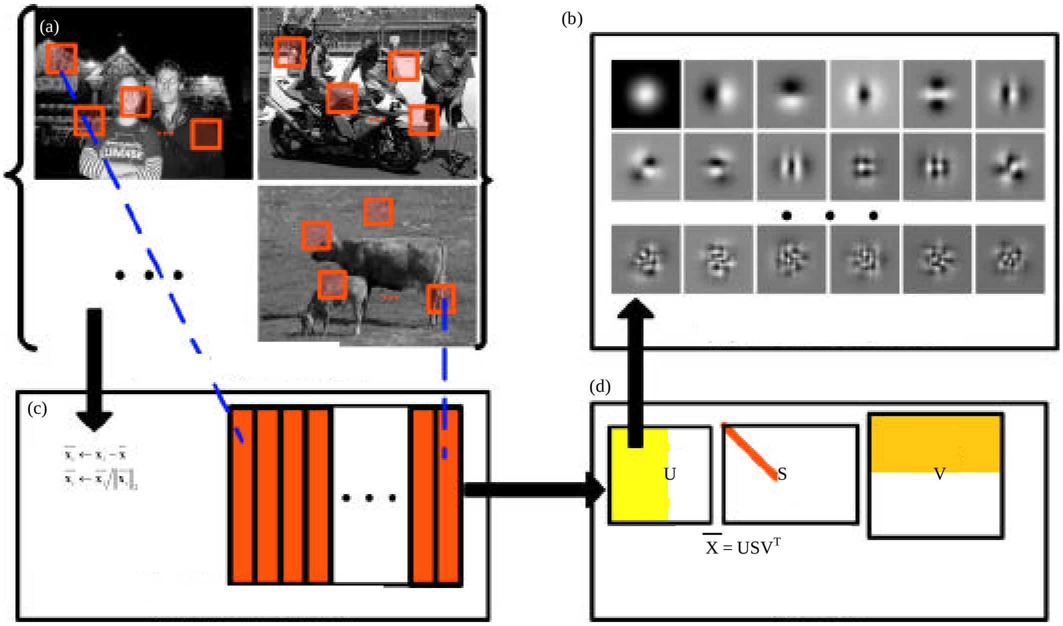

Fig. 3(a-d): The flowchart of the computation processing of PCA templates. The PCA templates are shown in terms of eigen-images (a) Natural images, (b) PCA templates, (c) Training set and (d) PCA

$$\cdot\ \bar{x} = \frac{1}{K \times D} \sum_i x_i$$

from the training set, $\bar{x}_I = \bar{x}_I - \bar{x}$ and then them to obtain the final training set $\bar{x}_I \leftarrow \bar{x}_I / \|\bar{x}_I\|_2$ The processing of collecting training samples is shown in Fig. 3b

Step 3: Apply Principal Component Analysis (PCA) (Turk and Pentland, 1991) to the final training set and then obtain the PCA bases $U = [u_1, u_2, ...]$, as shown in Fig. 3c

Step 4: Finally, extract M PCA templates $[u_2, u_3, ..., u_{M+1}]$, as texture patterns

Figure 3d demonstrates 36 PCA templates, where the first row shows the templates from 1-6, the second row illustrates the templates from 7-12 and the last row shows the last 6 templates. PCA basis functions are sorted in the order of decreasing variances which means that the templates of lower spatial frequencies are located in the front as they account for the main part of the variance. It can be seen from Fig. 3d that the first two rows stand for some common texture patterns (e.g., low-pass, edge, bar patterns and grating- like patterns). In addition, the last row indicates some noise patterns. Thus, M PCA templates $[u_2, u_3, ..., u_{M+1}]$, can be used as texture patterns which include the first M+1 PCA basis functions that except the first basis function as it can be indicated as a Gaussian low-pass filter. Finally, M texture patterns are obtained and denoted as $w_i = u_{i+1}|_{i=1,...,M}$.

The RPPBTF extraction: For a given R×C image, the traditional PPBTF method uses M texture patterns (PCA templates) to build a pattern map P:

$$P(r, c) = \arg\max_m \{w_m, x\} \qquad (3)$$

where, x denotes a neighbor vector that is vectorized from a S×S image patch centering at (r, c). For a given region R, the normalized feature is calculated by histogram statistic:

$$F_{PPBTF} = \left[\overline{f_1}, \overline{f_2}, ..., \overline{f_M}\right]^T$$

$$\overline{f_m} = \frac{\sum_{(r,c)\in R} \delta(P(r,c) = m)}{\sum_{m=1}^{M} \sum_{(r,c)\in R} \delta(P(r,c) = m)},\ m = 1, ..., M \qquad (4)$$

It is obvious that the maximization is a kind of hard operator as it merely retains the maximization feature response and discards the remaining ones. However, the real textures of natural images are very complex. This hard operator loses much useful information which will lead to a low accuracy.

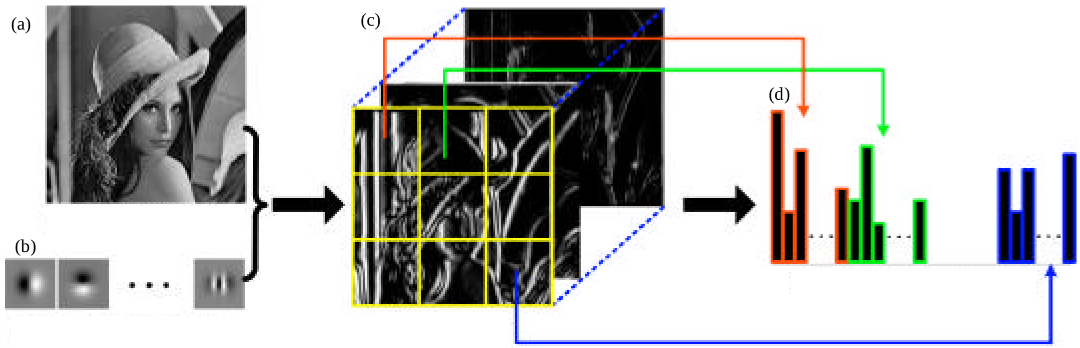

Fig. 4(a-d): The RPPBTF extraction processing (a) Original image, (b) Texture patterns, (c) Feature responses and (d) RPPBTF histogram

Motivated by this observation, the (relaxed pixel-pattern-based texture feature) RPPBTF method is presented. To extract RPPBTF, M texture patterns (PCA templates) are used to filter the original images first and therefore obtain a feature response tensor T (Fig. 4c) which is a three dimensional R×C×M array. To obtain the feature response T (r, c, m), a S×S neighbor block that centers at (r, c) is extracted and converted to a column vector x. The feature response is calculated by:

$$T(r, c, m) = \langle w_m, x \rangle \tag{5}$$

where, r = 1,..., R, c = 1,..., C and m = 1,..., M. The RPPBTF method adopts a soft operator rather than a hard operator. The soft operator normalizes the feature response at each position to make sure the consistency of scale by:

$$\overline{T}(r,c,m) = \frac{T(r,c,m)}{\sum_{m=1}^{M} T(r,c,m)} \tag{6}$$

Then, for a given region R, the histogram feature $F_{RPPBTF} = [\overline{f}_1, \overline{f}_2, ..., \overline{f}_M]^T$ can be obtained by:

$$\overline{f}_m = \frac{\sum_{(r,c)\in R} \overline{T}(r,c,m)}{\sum_{m=1}^{M} \sum_{(r,c)\in R} \overline{T}(r,c,m)}, \; m = 1, ..., M \tag{7}$$

Similar to the methods (Shan, 2012; Ahonen *et al*., 2004) that use LBP features to solve face problems, the face image is first divided into small regions from which the PPBTF or RPPBTF histogram is extracted and concatenated into a single, spatially enhanced feature histogram Fig. 4d. To compute the RPPBTF, the Eq. 7 requires calculating histograms on a 3D array rather than a 2D matrix that is adopted in PPBTF Eq. 4. Thus, RPPBTF needs some additional computational costs compared with PPBTF. Here, experiments will be conducted to show the RPPBTF's performance in terms of both accuracy and speed.

EXPERIMENTS

In these experiments, 661 images of 248 subjects are collected from the FERET (Phillips *et al*., 2000) database which are cropped and resized into 32×32 pixels. 200 images (100 male and 100 female) are selected for training and use the remaining 461 images for testing. For the raw pixel and Gabor(sum) features, the Gaussian kernel Eq. 8 is chosen to train the SVM classifier. For the histogram-based features (LBP, PPBTF and RPPBTF), the exponential chi-square kernel Eq. 9 is chosen to train the SVM classifier. The SVM-KM (Canu *et al*., 2005) toolbox is used to learn the SVM classifier. The penalty parameter and the kernel parameter are determined by cross validation:

$$K(x_i, x_j) = \exp\left(\frac{-\|x_i - x_j\|_2^2}{2\sigma^2}\right) \tag{8}$$

where, σ denotes the variance of the Gaussian kernel.

$$K(x_i, x_j) = \exp\left(\frac{-\chi^2(x_i, x_j)}{t}\right) \tag{9}$$

where, χ^2 stands for the chi-square distance.

$\chi^2(a,b) = \sum_i \frac{(a_i - b_i)^2}{a_i + b_i}$, $a = [a_1, a_2, ..., a_n]^T$, $b = [b_1, b_2, ..., b_n]^T$ and t

Table 1: Comparisons among raw pixel

	Raw	Gabor (sum)	LBP	PPBTF	RPPBTF
Accuracy (%)	83.95	86.77	89.80	86.98	91.11
Feature dimension	1024	1024	1475	250	250
CPU time (sec)	0.0002	0.0238	0.0026	0.0033	0.0078

Edelman *et al.* (1998), Gabor (sum) Tao *et al.* (2007), LBP: Local binary pattern, Ojala *et al.* (2002), PPBTF: Pixel-pattern-based texture feature, Zeng *et al.* (2004) and RPPBTF: Relaxed pixel-pattern-based texture feature, methods

Table 2: Comparisons between PPBTF and RPPBTF methods of different template number M when the block size is 5×5

	M = 3	M = 5	M = 8	M = 10	M = 12	M = 15
PPBTF accuracy (%)	81.34	84.16	85.90	86.98	86.55	88.50
RPPBTF accuracy (%)	84.16	85.90	90.24	91.11	91.97	91.32
Feature dimension	75.00	125.00	200.00	250.00	300.00	375.00

is a parameter that rescales the distance. All experiments are implemented by MATLAB on a standard i5-580 2:67 GHz PC with 2:0 GB RAM.

Comparison between RPPBTF and other features: To demonstrate the effectiveness of RPPBTF, it is compared with raw pixel (Edelman *et al*., 1998), Gabor (sum) (Tao *et al*., 2007), LBP (Ojala *et al*., 2002) and PPBTF features (Zeng *et al*., 2004). For the LBP feature, the $LBP_{8,1}$ descriptor and 5×5 blocks are used to build a spatially enhanced feature histogram. As suggested by Lu *et al.* (2008) and Wang *et al.* (2011), 5×5 image patches are collected to train PCA basis functions. For PPBTF and RPPBTF, 10 PCA templates and 5×5 blocks are used to generate feature histograms. The effects of template number and block size are investigated in this section. Table 1 reports the results of different features.

It can be seen that the proposed RPPBTF achieves not only high accuracy but also low dimension. It also can be seen that the speed of RPPBTF is also satisfied. Since, the raw pixel method does not require any additional feature extraction processing, its speed is very fast but its accuracy is not satisfied. Compared with PPBTF, RPPBTF achieves more accuracy (almost 5% improvement) without much additional computational load.

The discussions on different parameters: Different template number: the template number M is a very important parameter as it is related to the texture patterns. Table 2 shows the results obtained by PPBTF and RPPBTF of templates M when the block size is fixed to 5×5.

It can be seen from this table that the better results are obtained when 10, 12 and 15 templates are used. When M is smaller than 8, the number of PCA templates (patterns) is not able to capture enough texture information. However, if the number of templates is too large, the feature dimension (the computational load) is very high which not benefits practical application. Finally, M = 10 is chosen to balance the accuracy and feature dimension. From Table 2, it also can be seen that the proposed RPPBTF method achieves better performance than the original PPBTF algorithm.

Table 3: Comparison between PPBTF and RPPBTF methods of varied block size (N×N), when 10 PCA templates are used

	N = 2	N = 3	N = 4	N = 6	N = 6
PPBTF accuracy (%)	79.39	84.38	84.60	86.98	83.73
RPPBTF accuracy (%)	82.65	84.60	89.15	91.11	87.64
Feature dimension	40.00	90.00	160.00	250.00	360.00

Varied block size: the block size $_N$ is also an important parameter to be investigated. Table 3 shows recognition accuracies of PPBTF and RPPBTF with varied block size.

It can be seen from Table 3 that N = 5 is a good candidate parameter in the experiments. The potential reason is that for face gender classification, most of the regions are very similar. If N is too large (such as N = 6), the contribution of noise is also very large which results in a low accuracy. But when N is too small, the feature descriptor is not able to describe enough spatial information. In addition, it can be seen that the proposed RPPBTF method performs consistently better than the original PPBTF method.

CONCLUSION

In this study, a novel texture feature, Relaxed Pixel-Pattern-Based Texture Feature (RPPBTF) is proposed for gender classification. Compared to the original PPBTF method, the proposed RPPBTF algorithm adopts a soft operator for extracting texture features which alleviates information loss without introducing much computational load. For achieving gender classification, a pipeline is designed based on the RPPBTF method and the SVM classification technique. Many experiments are conducted on the FERET face database to compare the results of different texture features which demonstrates that the RPPBTF method achieves better performance than other popular features. In addition, the effects of different parameters (i.e., the number of templates and the size of blocks) are empirically discussed. The further works will focus on discussing different kinds of templates (e.g., ICA and NMF) and finding more potential applications.

ACKNOWLEDGMENT

This research is supported by Fundamental Research Funds for Central Universities (Grant No. DC120101131).

REFERENCES

Ahonen, T., A. Hadid and M. Pietikainen, 2004. Face recognition with local binary patterns. Proceedings of the 8th European Conference on Computer Vision, May 11-14, 2004, Prague, Czech Republic, pp: 469-481.

Baluja, S. and H.A. Rowley, 2007. Boosting sex identification performance. Int. J. Comput. Vision, 71: 111-119.

Canu S., Y. Grandvalet, V. Guigue and A. Rakotomamonjy, 2005. SVM and kernel methods matlab toolbox. Perception Systemes et Information, INSA de Rouen, Rouen, France.

Edelman, B., D. Valentin and H. Abdi, 1998. Sex classification of face areas: How well can a linear neural network predict human performance. J. Biol. Syst., 6: 241-264.

Golomb, A., D.T. Lawrence and T.J. Sejnowski, 1991. Sexnet: A Neural Network Identifies Sex from Human Faces. In: Advances in Neural Information Processing Systems 3, Touretzky, D.S. and R. Lippman (Eds.). Morgan Kaufmann, San Mateo, CA., pp: 572-577.

Heidemann, G., 2006. The principal components of natural images revisited. IEEE Trans. Pattern Anal. Mach. Intell., 28: 822-826.

Lee, T.S., 1996. Image representation using 2D gabor wavelets. IEEE Trans. Pattern Anal. Machine Intell., 18: 959-971.

Lu, H.C., Y.J. Huang, Y.W. Chen and D.L. Yang, 2008. Automatic gender recognition based on pixel-pattern-based texture feature. J. Real-Time Image Process., 3: 109-116.

Moghaddam, B. and M.H. Yang, 2000. Gender classification with support vector machines. Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, March 28-30, 2000, Grenoble, France, pp: 306-311.

Ojala, T., M. Pietikainen and T. Maenpaa, 2002. Multiresolution Gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Analysis Machine Intellig., 24: 971-987.

Phillips, P.J., H.J. Moon, S.A. Rizvi and P.J. Rauss, 2000. The FERET evaluation methodology for face recognition algorithms. IEEE Trans. Pattern Anal. Mach. Intell., 22: 1090-1104.

Shan, C.F., 2012. Learning local binary patterns for gender classification on real-world face images. Pattern Recogn. Lett., 33: 431-437.

Tamura, S., H. Kawai and H. Mitsumoto, 1996. Male/female identification from 8×6 very low resolution face images by neural network. Pattern Recogn., 29: 331-335.

Tao, D.C., X.L. Li, X.D. Wu and S.J. Maybank, 2007. General tensor discriminant analysis and Gabor features for gait recognition. IEEE Trans. Pattern Anal. Mach. Intell., 29: 1700-1715.

Turk, M. and A. Pentland, 1991. Eigenfaces for recognition. J. Cognitive Neurosci., 3: 71-86.

Wang, C.X., Y. Liu and Z.Y. Li, 2008. Algorithm research of face image gender classification based on 2-D gabor wavelet transform and SVM. Proceedings of the International Symposium on Computer Science and Computational Technology, December 20-22, 2008, Shanghai, China, pp: 312-315.

Wang, D., H.C. Lu and M.H. Yang, 2013. Online object tracking with sparse prototypes. IEEE Trans. Image Process., 22: 314-325.

Wang, D., H.C. Lu and X.L. Li, 2011. Two dimensional principal components of natural images and its application. Neurocomputing, 74: 2745-2753.

Zeng, XY., Y.W. Chen, Z. Nakao and H.Q. Lu, 2004. Texture representation based on pattern map. Signal Process., 84: 589-599.