



Face recognition using AMVP and WSRC under variable illumination and pose

Kang Wang¹ · Zhenxue Chen¹ · Q. M. Jonathan Wu² · Chengyun Liu¹

Received: 10 February 2017 / Accepted: 27 December 2017 / Published online: 6 January 2018
© The Natural Computing Applications Forum 2018

Abstract

Face recognition with variable illumination and pose is an important and challenging task in computer vision. In order to solve the problem that the accuracy of face recognition reduces with illumination and pose changes, this paper proposes a method via AMVP (AWULBP_MHOG_VGG_PCA) features and WSRC. In the proposed method, we need to extract AWULBP_MHOG and VGG_PCA features, respectively. As for AWULBP_MHOG features, firstly, variable illumination is normalized for face images. And uniform local binary pattern (ULBP) and multiple histogram of oriented gradient (MHOG) features are extracted from each block, which are called ULBP_MHOG features. Then, we use information entropy to obtain adaptively weighted ULBP_MHOG (AWULBP_MHOG) features. As for VGG_PCA features, we use the pre-trained VGG-Face model to extract VGG features from original face images. And PCA is used to reduce the dimension of VGG features to obtain VGG_PCA features. Then, AWULBP_MHOG and VGG_PCA features are combined to form AMVP (AWULBP_MHOG_VGG_PCA) features. Finally, test face images can be classified using weighted sparse represent (WSRC). The comparison experiments with different blocks, classifiers and features have been conducted on the ORL, Yale, Yale B and CMU-PIE databases. Experimental results prove that our method can improve the accuracy effectively for illumination and pose variable face recognition.

Keywords Face recognition · Illumination and pose changes · AWULBP_MHOG_VGG_PCA features · WSRC

1 Introduction

Since the early 1990s, face recognition [1] has gained widespread attention in pattern recognition and machine learning. Besides, face recognition continues to mature, which can be widely applied in security system, authentication, digital surveillance, human–computer interaction

and other public places. Although great results have been achieved under constrained face images [2, 3], there are many difficulties in variable illumination, pose, expression, occlusion, etc., for face images in the real world. As a face image is taken, it is easy to be affected by the illumination and pose, especially in the outdoor environment. Face appearance can alter drastically due to illumination and pose changes. And illumination and pose changes for the same face identity have a bigger impact than changes from variation of face identity [4]. Therefore, illumination and pose variable face recognition is a challenging task.

1.1 Review of works

Variable illumination and pose are the bottleneck of face recognition. Up to now, many works considering them separately have appeared. In terms of the only illumination variable face recognition, approaches can be approximately divided into three types, which are illumination pretreatment techniques, illumination model techniques and

✉ Zhenxue Chen
chenzhenxue@sdu.edu.cn

Kang Wang
kangkang_wanglucky@163.com

Q. M. Jonathan Wu
jwu@uwindsor.ca

Chengyun Liu
liuchengyun@sdu.edu.cn

¹ School of Control Science and Engineering, Shandong University, Jinan 250061, China

² Department of Electrical and Computer Engineering, University of Windsor, Windsor N9B 3P4, Canada

invariant feature extraction techniques. As for illumination pretreatment techniques, histogram equalization (HE) [5] is commonly used. In addition, Tan et al. [6] proposed an illumination pretreatment technique via three steps under local ternary patterns (LTP) operator. But these methods do not fully yield satisfactory results. As for illumination model techniques, Basri et al. [7] used a 9-D linear subspace to approximate varying illumination conditions. The disadvantage of illumination model techniques is that they need multiple face images under different illumination conditions. As for invariant feature extraction techniques, Gudur et al. [8] put forward a technique using Gabor wavelet and PCA, where Gabor wavelet was used to extract local features, which can effectively improve face recognition with illumination changes. But the dimension of Gabor features is large because of many Gabor wavelet kernels used in feature extraction, and thus, the elapsed time is much more. Roy et al. [9] proposed the local gravity face (LG face) for illumination invariant and heterogeneous face recognition. Ramaiah et al. [10] put forward using deep convolutional neural networks (CNN) for face recognition under non-uniform illumination. But they need to tune many parameters.

In terms of the only pose variable face recognition, some methods include multi-view-based approaches [11], pose invariant feature-based approaches [12, 13] and morphable model-based approaches [14–16]. For example, Carlos et al. [17] proposed a method via stereo matching to compute the similarity between two face images. Li et al. [18] used the linear combination of training images to represent test faces, and linear regression coefficients can be obtained as extracted features for face recognition. But the capacity of the linear models is limited, and the non-linear models have lower efficiency in model training or testing. Chen et al. [19] extracted multi-scale local binary pattern (LBP) features from patches about 27 landmarks detected by a face alignment algorithm. Although LBP features from all patches are concatenated to high-dimensional feature vectors as the pose robust features, this method heavily depends on the precision of face alignment. In addition, it is hard to detect dense landmarks in unconstrained face images. Li et al. [20] put forward using a set of 3D displacement fields to generate virtual views of test faces and matching synthesized faces with training faces. Asthana et al. [21] presented the view-based active appearance model to match 3D models with 2D images. Despite their effectiveness, they require complicated computer graphics operations. They are at the cost of expensive computation.

Many papers have considered both illumination and pose changes, which are grouped into 3D-based methods [22] and 2D-based methods. Although 3D methods can achieve better performances, they are complex in

computation because they need to compute additional 3D information or select landmarks manually. As for 2D methods, uLBP [23], HOG [24, 25], etc., robust features are used. Bhele et al. [26] put forward extracting discriminating LBP-HOG vectors for face recognition. In addition, when the deep learning appears, face recognition makes progress. For instance, Parkhi et al. [27] used the deep structure that is the VGG-Face model for face recognition. These methods can improve the accuracy of face recognition with illumination and pose changes due to discriminative features. But the dimension of extracted features is huge, and the elapsed time is large.

In order to improve the accuracy of face recognition with variable illumination and pose, we put forward one method based on AMVP (AWULBP_MHOG_VGG_PCA) features and WSRC. Illumination is normalized firstly, which can greatly reduce the lighting effects. Because of the insensitiveness of the local binary pattern (LBP) and histogram of oriented gradient (HOG) to light and posture, we extract uniform LBP (ULBP) and multiple HOG (MHOG) features from each block to form ULBP_MHOG features. Since different face areas have different information, the idea of weighting is proposed. We use information entropy to calculate weights and form adaptively weighted ULBP_MHOG (AWULBP_MHOG) features, which highlight more face details. In addition, because deep models can extract more discriminative features, we use the pre-trained VGG-Face model that is a deep model to extract features called VGG features from original face images. PCA is used to reduce the dimension of VGG features to obtain VGG_PCA features. Then, AWULBP_MHOG and VGG_PCA features are combined to form AMVP features. Finally, weighted sparse representation classification (WSRC) is utilized to improve performance of face recognition.

1.2 Contributions and the structure of the work

The contributions in this paper are shown as follows:

1. The fused features are put forward, which are AWULBP_MHOG_VGG_PCA features. And a discriminative classifier WSRC is used, which fuses the location and sparse information.
2. As for AWULBP_MHOG features, MHOG features have lower dimension than original HOG features and information entropy is used to calculate weights that are adaptive. As for VGG_PCA features, they have less dimension than VGG features.
3. The accuracy has been improved effectively. Experimental results have been compared with different blocks, features and classifiers.

The paper is organized as follows. In Sect. 2, the proposed method is presented. Experimental results are shown and analyzed in Sect. 3. Finally, Sect. 4 presents conclusions.

2 The proposed method

To improve the accuracy of face recognition with variable illumination and pose, the method based on AWULBP_MHOG_VGG_PCA features and WSRC is proposed. The fused and discriminative AMVP features are put forward, which are combined with the discriminative classifier WSRC to improve the accuracy. As LBP and HOG features are not sensitive to illumination and poses, we propose to extract the adaptively weighted ULBP_MHOG features (AWULBP_MHOG) from image blocks, where adaptive weights are obtained by information entropy and ULBP_MHOG features include ULBP and MHOG features. AWULBP_MHOG features of all patches are concatenated to feature vectors, which are robust to pose and illumination. In addition, deep networks have good results in face recognition in recent years due to their deep structures. But the training process of deep networks is complex. Thus, we use the pre-trained VGG-Face model to extract VGG features from original face images. But the extracted VGG features have higher dimension, so PCA is used to reduce the dimension of VGG features to form VGG_PCA features. Then, AWULBP_MHOG and VGG_PCA features are concatenated with AMVP features, which are robust to pose and illumination. Meanwhile, WSRC has better performance. The effective combination of AWULBP_MHOG_VGG_PCA features and WSRC is used to minimize pose and illumination variation. The specific process of our method is presented in Fig. 1.

2.1 Illumination normalization

Illumination is an issue for face recognition. Tan et al. [6] proposed three steps to preprocess illumination, which outperformed histogram equalization (HE) [5] and self-quotient image (SQI) [28], etc., under local ternary patterns (LTP) operator. Thus, this paper uses this illumination normalization method. The specific steps are as follows.

1. Gamma Correction. It is a nonlinear gray-level transformation that replaces image I with I^γ , where $\gamma \in [0, 1]$ is a user-defined parameter. Here, we choose $\gamma = 0.2$ as the default setting. The local dynamic range of the image is reduced in bright regions and enhanced in shadowed or dark regions.
2. Difference of Gaussian (DOG) Filtering. Gamma correction cannot eliminate the shading effects. DOG

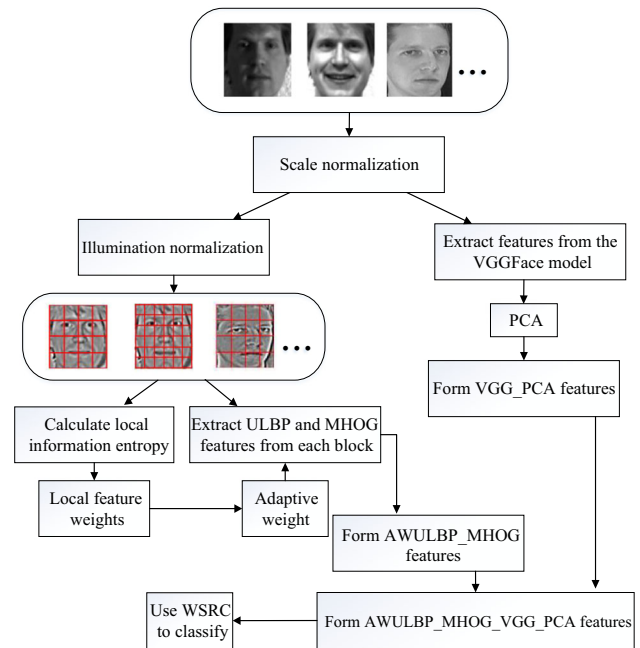


Fig. 1 Process of our method

filtering can suppress the illumination change in low frequency and the noise in high frequency. Its transfer function is the difference between two Gaussian functions with different width shown as Eq. (1), where $A_1 \geq A_2 > 0$, $\sigma_1 > \sigma_2$ are user-defined parameters. Here, we set $A_1 = A_2 = 1$, $\sigma_1 = 2$ and $\sigma_2 = 1$ as the default setting. Then, explicit convolution is used for filtering as Eq. (2).

$$G(x, y) = A_1 e^{-\frac{(x,y)^2}{2\sigma_1^2}} - A_2 e^{-\frac{(x,y)^2}{2\sigma_2^2}} \tag{1}$$

$$I(x, y) = I(x, y) * G(x, y). \tag{2}$$

3. Contrast Equalization. The final step globally rescales the image contrast. We have a simple approximation via two stage processes as Eqs. (3) and (4), where a is a compressive factor that reduces the influence of large values and τ is a threshold to remove large values. Here, we use $a = 0.1$ and $\tau = 10$ as default. Finally, to reduce a series of influences, we apply a nonlinear function to compress over-large values as Eq. (5), which limits I to $(-\tau, \tau)$.

$$I(x, y) \leftarrow \frac{I(x, y)}{(\text{mean}(|I(x, y)|)^a)^{1/a}} \tag{3}$$

$$I(x, y) \leftarrow \frac{I(x, y)}{(\text{mean}(\min(\tau, |I(x', y')|)^a))^{1/a}} \quad (4)$$

$$I(x, y) \leftarrow \tau \tanh\left(\frac{I(x, y)}{\tau}\right). \quad (5)$$

Figure 2 shows the results of illumination normalization. It can be seen that most influences of light are eliminated and more facial details are kept by illumination normalization.

2.2 Image blocks

Pentland et al. [29] firstly applied image blocks to face recognition. To extract more accurate features of face images, we divide the image into blocks. Then, local features of each block are extracted. At last, all local features are integrated into global features of the whole image for classification. Those extracted features are more robust to illumination, pose and expression changes.

This paper adopts non-overlapping manner to segment images and chooses 100×100 -sized face images as input. Each image is divided into $2N \times 2N$ blocks, where N is 1, 2, 3, etc. We assume the block template is $p \times p$ sizes, then $p = 100/2N$. Because of the indivisible phenomenon in image block process, we can segment images from the $(\text{mod}(100/2N)/2 + 1)$ th pixel, where the function $\text{mod}()$ is as remainder. Figure 3 presents face images with different blocks.

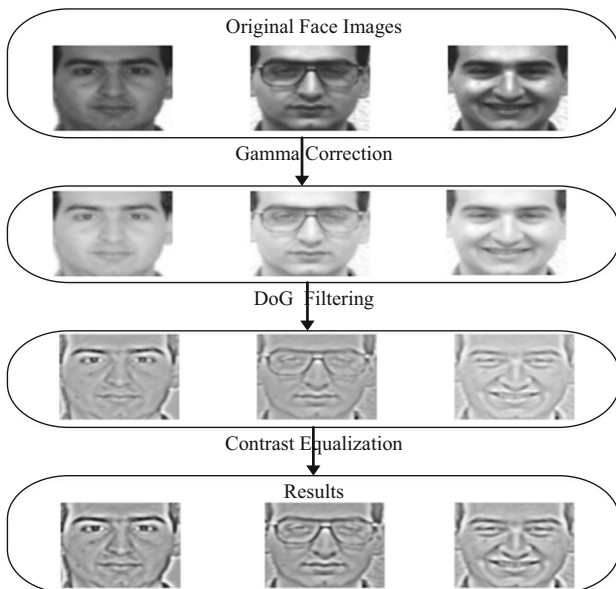


Fig. 2 Results of illumination normalization

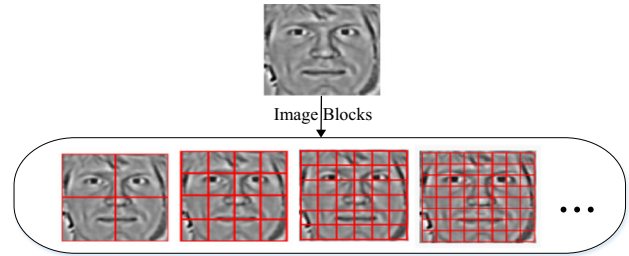


Fig. 3 Face images with different blocks

2.3 Uniform pattern LBP (ULBP)

As for the original LBP operator, a 3×3 neighborhood of each pixel is considered and the difference between every neighbor and the center pixel that is a threshold value is performed. The neighbor pixel which is less than the threshold value is set to 0, otherwise set to 1. Then, a binary number is formed and the value of LBP code for the center pixel is given as Eq. (6).

$$\text{LBP}_{p_c} = \sum_{i=1}^8 s(p_i - p_c)2^{i-1}, \quad s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (6)$$

where P_c is the center pixel which is considered as the threshold value, and neighbors are (P_1, \dots, P_8) .

Figure 4 shows the computation of the LBP code for a given pixel using a 3×3 neighborhood [30]. Ojala et al. [31] proposed some improved LBP operators, such as circular LBP operator, rotation-invariant operator, uniform pattern and rotation-invariant uniform pattern. As for uniform pattern LBP (ULBP), the binary code contains at most two bitwise transitions from 0 to 1 or from 1 to 0. For example, 11111111 (0 transitions) and 10001111 (2 transitions) are uniform patterns, while 11001001 (4 transitions) and 01010010 (6 transitions) are non-uniform patterns called mixed patterns. In terms of ULBP labels, each uniform pattern is as a separate label and mixed pattern is considered as one label. Then, the labels are reduced from 2^P to $P(P - 1) + 3$ for the ULBP operator. When we use the 8-neighborhood, there are a total of 256 patterns, 58 of which are uniform patterns. Thus, there are 59 different labels in total. Here, we use the ULBP operator that is computed on a circle of one radius and 8 neighbors.

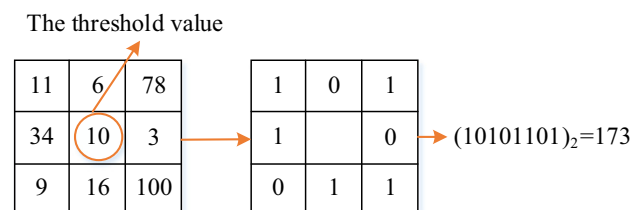


Fig. 4 Original LBP code illustration

2.4 Multiple histogram of oriented gradient (MHOG)

The HOG features were first introduced by Dalal and Triggs [32] for detecting humans in static images. HOG features are obtained by calculating and counting the local image gradient histogram. In this paper, we put forward a modified operator called multiple histogram of oriented gradient (MHOG). When original HOG features are extracted, one block of face images is divided into 4 cells and each cell is represented by 9 gradient orientations. When MHOG features are extracted, one block of face images is not divided into cells and each block is represented by three kinds of gradient orientations that are 3, 6 and 9 gradient orientations. Compared with HOG features, MHOG features are reduced by half in dimension. The generated MHOG operator is described as shown in Fig. 5.

2.5 Adaptively weighted approach

As for one face image, different areas of the image contain different texture information. In order to distinguish the information of different regions, this paper divides images into blocks and weights each block. Here, we use information entropy to calculate adaptive weights. The Shannon theorem [33] indicates that the information entropy represents how much information the image contains. When the information contained in one image is more, the information entropy of which is higher. On the contrary, if information contained in one image is less, the corresponding information entropy is lower. In this paper, ULBP_MHOG features are used, which include texture information. When the higher information entropy as the weight is combined with ULBP_MHOG features in the block of one image, the texture information is highlighted in the block of one image. Instead, texture details in the block of one image are reduced. Then, the weighted features are more

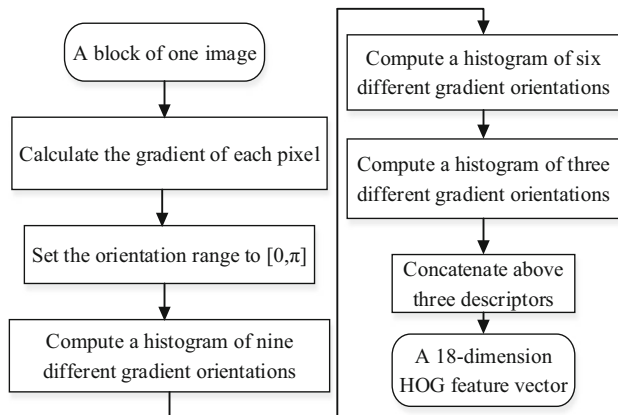


Fig. 5 MHOG operator illustration

discriminative. If one image is divided into m subblocks, the information entropy of the i th subblock can be expressed as Eq. (7).

$$E_i = - \sum_{k=1}^n p_i^k \lg p_i^k, \tag{7}$$

where n is pixel series which is 256 in this paper and p_i^k is probability of the k th series pixel. If one subblock has the higher information entropy, it can obtain the higher weight. The weight coefficient of the i th subblock can be determined by Eq. (8).

$$W_i = E_i / \left(\sum_{i=1}^m E_i \right). \tag{8}$$

2.6 AWULBP_MHOG and VGG_PCA features (AMVP)

After scale normalization, AWULBP_MHOG and VGG_PCA features are extracted from face images, respectively. As for AWULBP_MHOG features, we first normalize illumination and divide images into different blocks. Then, ULBP and MHOG features are extracted from each block, which are concatenated with new features called ULBP_MHOG features. Meanwhile, we normalize the information entropy of each block, which is as a weight of each block. The adaptively weighted ULBP_MHOG (AWULBP_MHOG) features are formed by multiplying ULBP_MHOG features by the weight in each block. Finally, the whole AWULBP_MHOG features are obtained by concatenating AWULBP_MHOG features of each block. As for VGG_PCA features, our implementation is based on the MATLAB toolbox MatConvNet (<http://www.vlfeat.org/matconvnet/>). In this paper, we use the VGG-Face model (http://www.robots.ox.ac.uk/~vgg/software/vgg_face/) that was provided by O. M. Parkhi et al. [27] to extract face features. The VGG-Face model contains 39 layers that can be seen from Fig. 6. We consider the output of 36th layer as the final VGG features that are 4096-dimension descriptor vectors. In order to reduce the feature dimension, 4096-dimension VGG features are reduced by PCA. Then, the VGG_PCA features can be obtained. Finally, AWULBP_MHOG and VGG_PCA features are concatenated to form AWULBP_MHOG_VGG_PCA features. The specific process of extracting AWULBP_MHOG and VGG_PCA features is shown in Fig. 6.

2.7 Weighted sparse representation classification (WSRC)

Sparse representation for classification (SRC) has attracted much attention and achieved better classification result

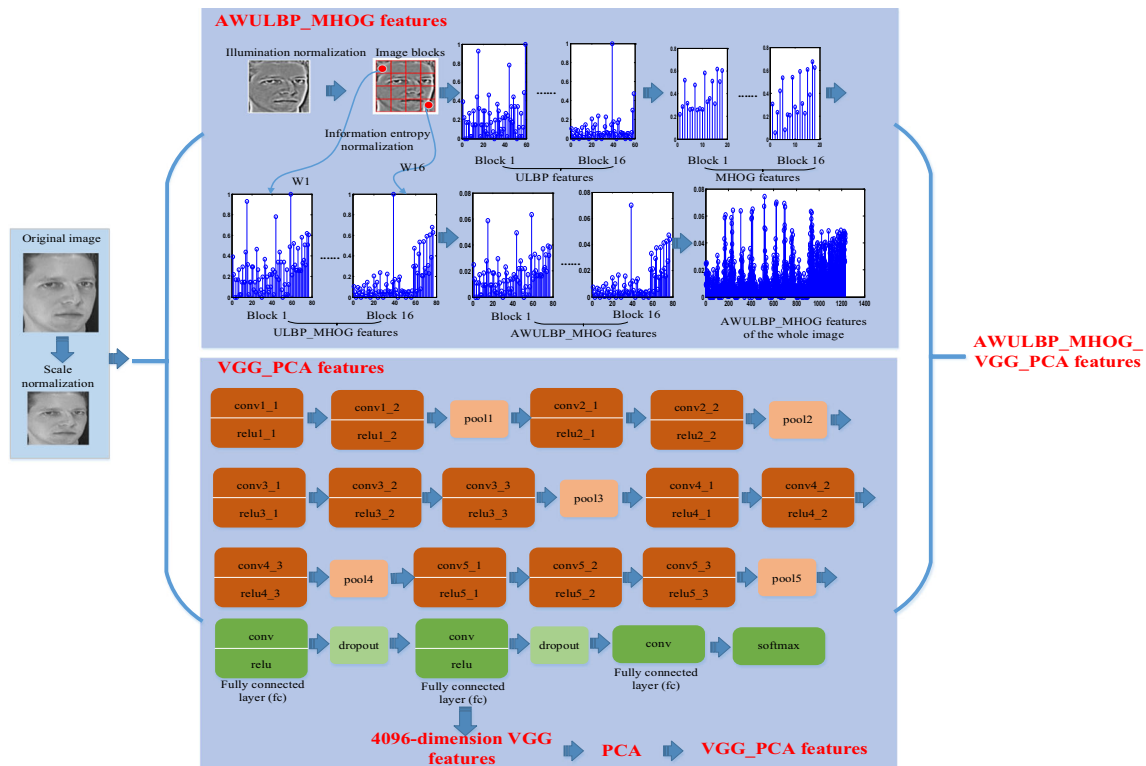


Fig. 6 Process of extracting AMVP features

than other typical classification methods. Fan et al. [34] pointed out that training samples that are more similar or closer to the test sample generally play more important role in representing the test sample in SRC. And Yu et al. [35] pointed out that SRC lacks locality information and locality information is more important than sparsity information under some conditions. To overcome above drawbacks, Lu et al. [36] and Fan et al. [34] put forward the weighted sparse representation for classification (WSRC). WSRC pays more attention to those training samples that are more similar to the test sample in representing the test sample. The goal of WSRC is to measure the significance of each training sample in representing test samples, and the significance can be evaluated by computing the distance between the training sample and the test sample. Because WSRC exploits the distance information in representing test samples, whereas SRC does not exploit it, WSRC enhances the classification effectiveness of SRC. The WSRC includes two steps. The first step is to calculate distances between training samples and a given test sample as weights of training samples. Here, for convenience, we just use the Euclidean distance to compute their distances. The second step is to perform SRC by using weighted training samples.

As for typical SRC, we set training sample vector as v_{ij} , which represents the feature vector of the j th sample in the

i th class. Then, all samples of the i th class make up a matrix denoted as $A_i = [v_{i1}, v_{i2}, \dots, v_{in_i}]$. The test sample y_i of the i th class is linearly represented by A_i as Eq. (9).

$$y_i = x_{i1}v_{i1} + x_{i2}v_{i2} + \dots + x_{in_i}v_{in_i}, \tag{9}$$

where x_{ij} is the reconstruction coefficient of the j th sample. Considering all training samples, y_i can be represented by all training samples as Eq. (10).

$$y_i = 0 \cdot v_{11} + \dots + 0 \cdot v_{1n_1} + \dots + x_{i1} \cdot v_{i1} + \dots + x_{in_i} \cdot v_{in_i} + 0 \cdot v_{m1} + \dots + 0 \cdot v_{mn_m}, \tag{10}$$

where $x_i = [0, \dots, 0, x_{i1}, \dots, x_{in_i}, 0, \dots, 0]$ is a coefficient vector, and corresponding elements of the i th class are the only nonzero terms. In order to calculate sparse coefficient vector x , the minimum ℓ^0 norm problems need to be solved according to Eq. (11).

$$\hat{x}_0 = \arg \min \|x\|_0 \quad s.t. \quad y = Ax, \tag{11}$$

where $\|\cdot\|_0$ denotes the ℓ^0 -norm, which counts the number of nonzero terms in a vector. But it is a NP-hard problem to find the sparsest solution of Eq. (11), which is difficult to appropriate [36]. If the solution x is sparse enough, the solution of ℓ^0 -minimization problem is equal to the ℓ^1 -minimization problem as Eq. (12).

$$\hat{x}_1 = \arg \min \|x\|_1 \quad s.t. \quad y = Ax, \tag{12}$$

where \hat{x}_1 is the sparse coefficient vector. After that, we use $\delta_i(\hat{x}_i)$ function to choose coefficients which are related to i th class and set other coefficients which are not related to i th class to 0. Then, the test sample can be represented as Eq. (13). And the residual is expressed as Eq. (14).

$$\hat{y} = A\delta_i(\hat{x}_i), \tag{13}$$

$$r_i(y) = \|y - \hat{y}\|_2. \tag{14}$$

The test sample y is classified according to the residual as Eq. (15).

$$y \in \arg \min_i r_i(y), \tag{15}$$

where $r_i(y)$ is the residual. When the residual is the minimum, its corresponding category is the class of test samples.

As for WSRC, we just treat all weighted training samples as dictionary on the ℓ^1 regularization. The weighted ℓ^1 –minimization is solved by Eq. (16).

$$\hat{x}_1 = \arg \min \|Wx\|_1 \quad s.t. \quad y = Ax, \tag{16}$$

where W is a weighted matrix, which is comprised of the distance between y and each training sample. Here, Euclidean distance can be used to compute their distances.

3 Experimental results and analyses

To verify the effectiveness of proposed method, many experiments are compared on the ORL database [37], the Yale database [38], the Yale B database [39] and the CMU-PIE database [40]. The ORL and the CMU-PIE databases include different poses about horizontal and vertical rotation and some variants in illumination. The Yale and the Yale B databases include various illumination changes. The illumination angle is the angle between the direction of light source and the camera axis on the Yale B database. As to pose various face recognition, the ORL and the CMU-PIE databases are used to verify. The databases mentioned above are employed to verify face recognition with different illumination conditions.

3.1 Experiments on the ORL database

AT & T Cambridge Laboratory captured a face image set called the ORL database. It includes 400 face images of 40 different persons, where each person has 10 face images. Face images of each person are shot at different time, which causes some differences about light changes, the facial expression, poses and whether to wear glasses, etc.

The original images are 112×92 pixels. Here, we crop all images to 100×100 pixels by scale normalization.

On the ORL database, AWULBP_MHOG features and WSRC method is firstly tested. Table 1 presents the accuracy of different blocks under AWULBP_MHOG features and WSRC. It indicates that the best result appears on 4×4 blocks. Particularly, when the number of training examples per class is 6, the highest accuracy (99.38%) is achieved. In Fig. 7, we present WSRC, random forests (RF), nearest neighbor (NN) and support vector machine (SVM) accuracy curves under different numbers of training samples for each class with 4×4 image blocks. As can be seen, WSRC outperforms all other methods in all cases. The RF method has the poorest performance when the number of training samples per class is 1, 2 and 3. As the number of training samples per class is 4, 5 and 6, the SVM has the lowest accuracy. Then, we test AWULBP_MHOG features, ULBP_MHOG features, AWULBP features and AWMHOG features under WSRC. The experimental results are shown in Fig. 8. We find that AWULBP_MHOG features are the best in all cases. In particular, when the number of training samples for each class is 6, the best accuracy (99.38%) is achieved based on AWULBP_MHOG features and WSRC.

In order to verify our proposed method, many contrast experiments of different methods are carried out. Table 2 shows the accuracy of different methods on the ORL database. We set cumulative contribution degree as 91, 93, 95, 97 and 99% in PCA. It indicates that our proposed method via AMVP features and WSRC has the highest accuracy (100%) when the number of training samples per class is 6. Our method is superior to the methods only using AWULBP_MHOG features. The fused AWULBP_MHOG_VGG_PCA features become more discriminative to variable illumination and pose.

3.2 Experiments on the Yale database

The Yale face database is conducted by the Center for Computational Vision and Control. There are 165

Table 1 Accuracy of different blocks on the ORL database (%)

Blocks	The number of training samples for each class					
	1	2	3	4	5	6
1×1	32.50	57.81	68.21	74.17	82.00	86.25
2×2	70.00	79.06	77.50	87.08	86.00	77.50
4×4	79.17	90.31	93.57	96.67	97.00	99.38
6×6	75.56	86.25	93.57	93.57	93.57	93.57
8×8	75.00	84.69	88.57	95.83	95.50	98.75

The bold number represents the highest accuracy under different training samples

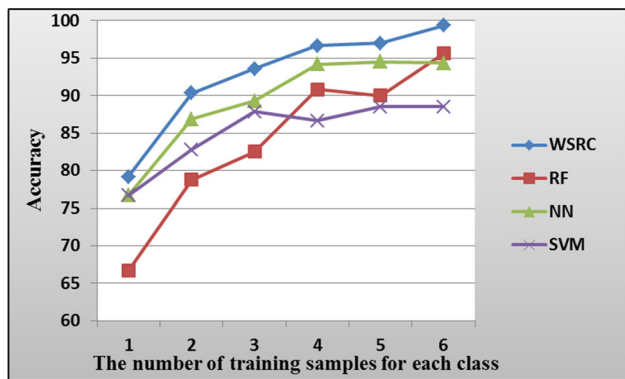


Fig. 7 Comparison of different classifiers on the ORL database

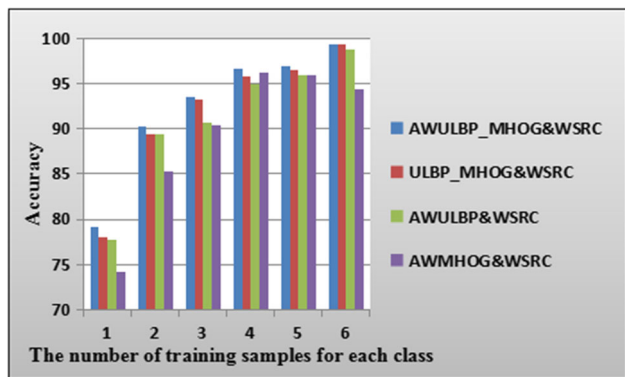


Fig. 8 Comparison of different features on the ORL database

grayscale images of 15 subjects, and each individual has 11 different images. These images include variations about illumination, facial expression, and whether to wear glasses. All images are cropped into 100×100 pixels.

The first experiment is to find the optimal image blocks on the Yale database under AWULBP_MHOG features. Face images are divided into $2N \times 2N$ blocks, where N is set to 0, 1, 2, 3, 4, 5 and 6, respectively. The relationship among the accuracy, image blocks and the number of training samples for each class is presented in Table 3. The results show that the highest accuracy has been achieved on 10×10 blocks except “2 Train.” Particularly, when the number of training samples per class is 3, 4 and 5, the accuracy can achieve the best (100%). Meanwhile, we can observe that block manner is better than non-block manner regardless of the number of training samples. Compared to 10×10 blocks, the accuracy reduces a little when the block is 12×12 . Figure 9 shows performance of different classifiers under AWULBP_MHOG features with 10×10 image blocks. We can observe that WSRC outperforms all other methods except “5 Train.” When the number of training samples for each class is 5, the accuracy of all classifiers is 100%. When there is only one training sample in each class, the accuracy of WSRC (87.33%) is much

Table 2 Accuracy of different methods on the ORL database (%)

Methods	Accuracy
AWULBP_MHOG + WSRC	99.38
AWULBP_MHOG + SVM	88.13
AWULBP_MHOG + KNN	94.38
AWULBP_MHOG + RF	95.63
VGG_PCA(91%) + WSRC	100
VGG_PCA(93%) + WSRC	100
VGG_PCA(95%) + WSRC	100
VGG_PCA(97%) + WSRC	100
VGG_PCA(99%) + WSRC	100
VGG_PCA(91%) + SVM	100
VGG_PCA(93%) + SVM	100
VGG_PCA(95%) + SVM	100
VGG_PCA(97%) + SVM	100
VGG_PCA(99%) + SVM	100
VGG_PCA(91%) + KNN	100
VGG_PCA(93%) + KNN	100
VGG_PCA(95%) + KNN	100
VGG_PCA(97%) + KNN	99.38
VGG_PCA(99%) + KNN	98.13
VGG_PCA (91%) + RF	100
VGG_PCA(93%) + RF	99.38
VGG_PCA(95%) + RF	99.38
VGG_PCA(97%) + RF	99.38
VGG_PCA(99%) + RF	95.00
AMVP(91%) + SVM	98.13
AMVP (93%) + SVM	98.75
AMVP (95%) + SVM	98.13
AMVP (97%) + SVM	98.75
AMVP (99%) + SVM	98.13
AMVP (91%) + KNN	98.13
AMVP (93%) + KNN	98.13
AMVP (95%) + KNN	98.13
AMVP (97%) + KNN	98.13
AMVP (99%) + KNN	97.50
AMVP (91%) + RF	99.38
AMVP (93%) + RF	98.75
AMVP (95%) + RF	98.75
AMVP (97%) + RF	98.75
AMVP (99%) + RF	98.13
Ours (AMVP (91%) + WSRC)	100
Ours (AMVP (93%) + WSRC)	100
Ours (AMVP (95%) + WSRC)	100
Ours (AMVP (97%) + WSRC)	100
Ours (AMVP (99%) + WSRC)	100

The bold number represents the highest accuracy under different methods

higher than that of RF (53.33%). As the training samples are too less, the accuracy of RF is lower. In addition, some

Table 3 Accuracy of different blocks on the Yale database (%)

Blocks	The number of training samples for each class				
	1	2	3	4	5
1 × 1	26.00	55.56	55.00	74.29	81.11
2 × 2	55.33	85.19	87.50	96.19	94.44
4 × 4	82.67	91.85	95.83	98.10	100
6 × 6	87.33	97.78	97.50	99.05	100
8 × 8	85.33	98.52	96.67	99.05	100
10 × 10	87.33	96.30	100	100	100
12 × 12	81.33	93.33	99.17	99.05	98.89

The bold number represents the highest accuracy under different training samples

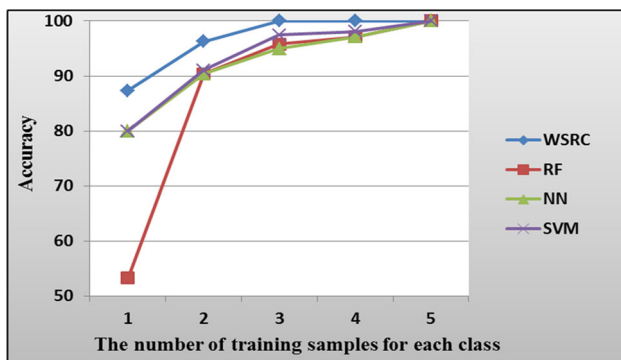


Fig. 9 Comparison of different classifiers on the Yale database

comparison experiments about different features using WSRC are shown in Fig. 10, from which we can see that AWULBP_MHOG features perform the best. Particularly, when the number of training samples per class is 1, 3 and 4, AWULBP_MHOG features are obviously superior to others. AWULBP_MHOG features and WSRC method can achieve 100% in “3 Train,” “4 Train” and “5 Train.” Experiments prove that if we use more galleries, we will get higher accuracies.

For our proposed method, we set cumulative contribution degree in PCA as 91, 93, 95, 97 and 99%. Table 4

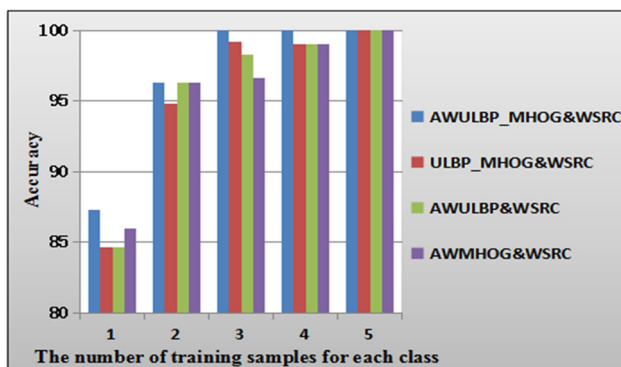


Fig. 10 Comparison of different features on the Yale database

Table 4 Accuracy of different methods on the Yale database (%)

Methods	Accuracy
AWULBP_MHOG + WSRC	100
AWULBP_MHOG + SVM	100
AWULBP_MHOG + KNN	100
AWULBP_MHOG + RF	100
VGG_PCA(91%) + WSRC	100
VGG_PCA(93%) + WSRC	98.89
VGG_PCA(95%) + WSRC	100
VGG_PCA(97%) + WSRC	100
VGG_PCA(99%) + WSRC	100
VGG_PCA(91%) + SVM	100
VGG_PCA(93%) + SVM	100
VGG_PCA(95%) + SVM	100
VGG_PCA(97%) + SVM	100
VGG_PCA(99%) + SVM	100
VGG_PCA(91%) + KNN	100
VGG_PCA(93%) + KNN	98.89
VGG_PCA(95%) + KNN	98.89
VGG_PCA(97%) + KNN	97.78
VGG_PCA(99%) + KNN	93.33
VGG_PCA(91%) + RF	97.78
VGG_PCA(93%) + RF	100
VGG_PCA(95%) + RF	100
VGG_PCA(97%) + RF	100
VGG_PCA(99%) + RF	100
AMVP (91%) + SVM	100
AMVP (93%) + SVM	100
AMVP (95%) + SVM	100
AMVP (97%) + SVM	100
AMVP (99%) + SVM	100
AMVP (91%) + KNN	100
AMVP (93%) + KNN	100
AMVP (95%) + KNN	100
AMVP (97%) + KNN	100
AMVP (99%) + KNN	100
AMVP (91%) + RF	100
AMVP (93%) + RF	100
AMVP (95%) + RF	100
AMVP (97%) + RF	98.89
AMVP (99%) + RF	100
Ours (AMVP (91%) + WSRC)	100
Ours (AMVP (93%) + WSRC)	100
Ours (AMVP (95%) + WSRC)	100
Ours (AMVP (97%) + WSRC)	100
Ours (AMVP (99%) + WSRC)	100

The bold number represents the highest accuracy under different methods

shows the accuracy of different methods on the Yale database. It can be shown that as the number of training

samples per class is 5, our proposed method via AWULBP_MHOG_VGG_PCA features and WSRC has the highest accuracy (100%).

3.3 Experiments on the Yale B database

There are 640 images of 10 people for the Yale B database, where each person has 64 different illumination conditions. All images are cropped and resized to 100 × 100 pixels.

Table 5 presents the results of searching for optimal image blocks on the Yale B database under AWULBP_MHOG features. Experiments are conducted on $2N \times 2N$ blocks. Here, the value of N is from 0 to 8. It can be seen that results on 14×14 and 16×16 blocks are almost similar, but the accuracy of 16×16 blocks is slightly better. The accuracy is higher with more image blocks, due to more useful details can be extracted from the Yale B images. When the image is divided into 16×16 blocks, accuracies are 99.52, 99.68, 99.67, 99.64, 99.58 and 100%, respectively, in AWULBP_MHOG and WSRC method. We compare WSRC with other classical algorithms which include RF, NN and SVM. In Fig. 11, the corresponding curves between accuracy and the number of training samples per class are shown, where the number of training samples per class is set to 2^N ($N = 0, 1, 2, 3, 4, 5$). Figure 11 indicates that the accuracies are almost similar among RF, SVM and WSRC except “1 Train” and “2 Train,” where the accuracies of WSRC are 99.67, 99.64, 99.58 and 100%, respectively, from “4 Train” to “32 Train.” And when the number of training samples is 1 and 2, WSRC method is superior to others. In order to prove the effectiveness of AWULBP_MHOG features, we conduct some comparison experiments which are shown in Fig. 12. When we use 16×16 blocks on the Yale B database, the accuracy of AWULBP_MHOG features is almost similar to the accuracy of ULBP_MHOG features under WSRC.

Table 5 Accuracy of different blocks on the Yale B database (%)

Blocks	The number of training samples for each class					
	1	2	4	8	16	32
1 × 1	45.08	48.06	38.67	53.57	52.29	55.63
2 × 2	62.86	68.06	69.00	74.82	71.88	77.81
4 × 4	85.71	88.87	94.50	95.00	96.46	99.06
6 × 6	85.71	95.97	98.83	98.93	99.38	99.69
8 × 8	90.63	99.03	99.50	99.29	99.38	99.69
10 × 10	90.63	98.23	99.50	99.64	99.58	100
12 × 12	99.05	98.55	99.67	99.46	99.58	100
14 × 14	99.52	99.19	99.67	99.64	99.58	100
16 × 16	99.52	99.68	99.67	99.64	99.58	100

The bold number represents the highest accuracy under different training samples

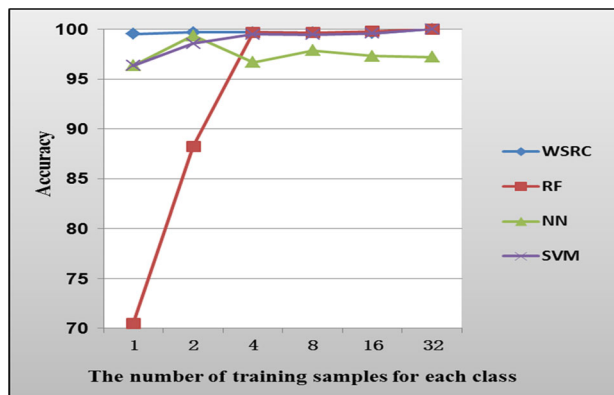


Fig. 11 Comparison of different classifiers on the Yale B database

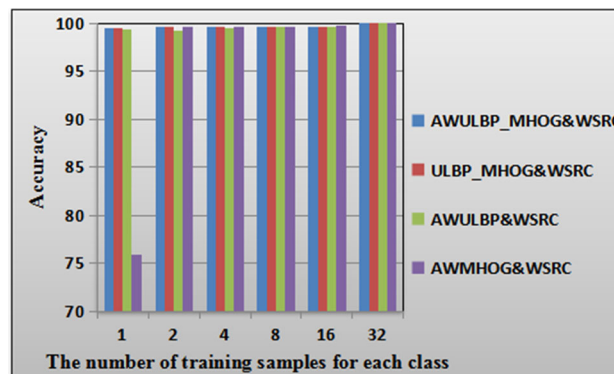


Fig. 12 Comparison of different features on the Yale B database

Excepting that the accuracy is the lowest (75.87%) based on AWMHOG and WSRC in “1 Train,” the accuracies are pretty high in other cases, which are nearly to 100%.

As for our proposed method, the cumulative contribution degree of 91, 93, 95, 97 and 99% is set in PCA. Table 6 shows the accuracy of different methods on the Yale B database. We can see that as the number of training samples per class is 32, our proposed method via AWULBP_MHOG_VGG_PCA features and WSRC has the highest accuracy (100%). The fused AMVP features have better performance than other features.

3.4 Experiments on the CMU-PIE database

A total of 41368 images of 68 different subjects are contained in the CMU-PIE database. In our experiments, we choose five pose subsets of the CMU-PIE database, which has pose yawing and the pitching variations in depth. They are pose set 05 and 29 (yawing about ± 22.5°), 07 and 09 (pitching about ± 20°) and 27 (near frontal), respectively. Each pose set includes 1632 images in total, where there are 68 subjects and each subset has 24 different lighting conditions. The pose class and face examples are given in Table 7. Here, we choose the pose 27 as the training set,

Table 6 Accuracy of different methods on the Yale B database (%)

Methods	Accuracy
AWULBP_MHOG + WSRC	100
AWULBP_MHOG + SVM	100
AWULBP_MHOG + KNN	97.20
AWULBP_MHOG + RF	100
VGG_PCA(91%) + WSRC	79.06
VGG_PCA(93%) + WSRC	78.44
VGG_PCA(95%) + WSRC	80.63
VGG_PCA(97%) + WSRC	82.81
VGG_PCA(99%) + WSRC	84.06
VGG_PCA(91%) + SVM	79.06
VGG_PCA(93%) + SVM	76.88
VGG_PCA(95%) + SVM	83.13
VGG_PCA(97%) + SVM	85.31
VGG_PCA(99%) + SVM	84.06
VGG_PCA(91%) + KNN	71.88
VGG_PCA(93%) + KNN	69.06
VGG_PCA(95%) + KNN	69.38
VGG_PCA(97%) + KNN	66.88
VGG_PCA(99%) + KNN	57.50
VGG_PCA(91%) + RF	80.00
VGG_PCA(93%) + RF	70.00
VGG_PCA(95%) + RF	84.06
VGG_PCA(97%) + RF	82.81
VGG_PCA(99%) + RF	77.50
AMVP (91%) + SVM	100
AMVP (93%) + SVM	100
AMVP (95%) + SVM	100
AMVP (97%) + SVM	100
AMVP (99%) + SVM	100
AMVP (91%) + KNN	97.81
AMVP (93%) + KNN	98.13
AMVP (95%) + KNN	98.44
AMVP (97%) + KNN	97.81
AMVP (99%) + KNN	98.44
AMVP (91%) + RF	100
AMVP (93%) + RF	100
AMVP (95%) + RF	100
AMVP (97%) + RF	100
AMVP (99%) + RF	100
Ours (AMVP (91%) + WSRC)	100
Ours (AMVP (93%) + WSRC)	100
Ours (AMVP (95%) + WSRC)	100
Ours (AMVP (97%) + WSRC)	100
Ours (AMVP (99%) + WSRC)	100

The bold number represents the highest accuracy under different methods

and other pose sets are used as test sets, respectively. As for the AWULBP_MHOG features, Table 7 shows the

Table 7 Accuracy of different blocks on the CMU-PIE database (%)

Blocks	Pose05	Pose07	Pose09	Pose29
1 × 1	12.13	44.14	43.32	16.97
2 × 2	38.42	81.22	80.21	35.72
4 × 4	60.23	85.45	85.21	51.53
8 × 8	79.53	93.49	93.93	76.04
10 × 10	77.70	87.72	93.08	73.96

The bold number represents the highest accuracy under different poses

accuracy of different blocks on the CMU-PIE database. As it is shown, when the block is 8 × 8, the higher accuracy (79.53, 93.49, 93.93 and 76.04%) is achieved in each pose set. However, when we increase blocks to 10 × 10, the accuracy declines instead. Besides, yawing variations may have a greater influence on accuracy in face recognition compared with pitching variations.

To evaluate the performance of our proposed method on the CMU-PIE database, we compare our method with other methods. Table 8 shows comparison results with different features and the combination with different classifiers. When the cumulative contribution degree in PCA is 99%, our method has the highest accuracy except Pose29 set. But the accuracy of our method is just 0.67% less than that of VGG_PCA (97%) +SVM in Pose29. And our method has the highest accuracy (90.67%) in average. Thus, our method has the better performance in the whole.

4 Conclusions

An illumination and pose variable face recognition method based on AMVP features and WSRC is presented. The main contribution of our work is that AMVP features are put forward, which include AWULBP_MHOG and VGG_PCA features. As for AWULBP_MHOG features, MHOG features are the variant of HOG features and have lower dimension, and adaptive weights are introduced to extract more effective detail information. As for VGG_PCA features, we use the pre-trained VGG-Face model to extract features, which are reduced by PCA according to different cumulative contribution degree. The fused features contain multi-source information and become more discriminative for classification. Another contribution is the combination with WSRC, which integrates locality and sparse information in classification. Besides, experiments have been conducted with different image blocks, classifiers, features on the different face databases. Extensive experiments show that the proposed method improves the recognition performance significantly.

Table 8 Accuracy of different methods on the CMU-PIE database (%)

Methods	Pose No.				Avg
	05	07	09	29	
AWULBP + WSRC	67.65	88.77	90.13	64.83	77.85
AWULBP + SVM	77.51	83.61	89.71	72.49	80.83
AWULBP + RF	45.28	64.40	63.66	48.71	55.51
AWULBP + NN	78.80	88.83	92.16	74.33	83.53
AWMHOG + WSRC	58.15	83.00	83.15	54.29	69.65
AWMHOG + SVM	36.95	54.51	68.87	41.12	50.36
AWMHOG + NN	49.75	77.35	86.15	54.72	66.99
AWMHOG + RF	42.28	52.30	54.17	38.66	46.85
ULBP_MHOG + SVM	67.46	89.87	91.97	63.73	78.26
ULBP_MHOG + RF	45.22	63.23	59.38	44.06	52.97
ULBP_MHOG + NN	79.35	90.55	93.81	76.29	85.00
ULBP_MHOG + WSRC	77.08	93.43	93.93	74.57	84.75
AWULBP_MHOG + SVM	77.02	83.30	90.44	68.38	79.79
AWULBP_MHOG + RF	49.26	63.90	59.31	43.14	53.90
AWULBP_MHOG + NN	82.11	90.42	93.75	77.63	85.98
AWULBP_MHOG + WSRC	79.53	93.49	93.93	76.04	85.75
VGG_PCA(91%) + SVM	81.43	86.07	87.56	79.66	83.68
VGG_PCA(93%) + SVM	83.15	86.80	89.83	82.60	85.60
VGG_PCA(95%) + SVM	78.25	79.68	91.91	83.03	83.22
VGG_PCA(97%) + SVM	79.29	84.78	90.87	83.88	84.71
VGG_PCA(99%) + SVM	82.11	82.57	88.97	82.11	83.94
VGG_PCA(91%) + KNN	78.37	81.65	82.72	73.84	79.15
VGG_PCA(93%) + KNN	73.59	79.80	82.05	70.89	76.58
VGG_PCA(95%) + KNN	63.76	60.16	75.61	62.68	65.55
VGG_PCA(97%) + KNN	51.04	65.13	66.91	56.25	59.83
VGG_PCA(99%) + KNN	46.69	55.99	53.98	48.28	51.24
VGG_PCA(91%) + RF	82.17	85.14	83.88	80.58	82.94
VGG_PCA(93%) + RF	82.54	84.47	88.66	80.58	84.06
VGG_PCA(95%) + RF	75.74	79.25	87.81	79.78	80.65
VGG_PCA(97%) + RF	73.90	84.65	86.46	80.45	81.37
VGG_PCA(99%) + RF	76.84	83.24	76.04	77.94	78.52
VGG_PCA(91%) + WSRC	80.51	84.78	87.38	78.31	82.75
VGG_PCA(93%) + WSRC	81.56	85.45	87.25	80.51	83.69
VGG_PCA(95%) + WSRC	81.25	83.79	87.68	79.35	83.02
VGG_PCA(97%) + WSRC	82.05	84.04	87.68	80.21	83.50
VGG_PCA(99%) + WSRC	82.84	84.71	88.24	80.88	84.17
AMVP (91%) + SVM	71.75	90.06	92.40	66.73	80.23
AMVP (93%) + SVM	71.69	90.06	92.34	66.48	80.14
AMVP (95%) + SVM	71.81	89.87	92.59	66.91	80.30
AMVP (97%) + SVM	72.49	90.18	92.46	67.10	80.56
AMVP (99%) + SVM	72.98	90.61	92.71	67.59	80.97
AMVP (91%) + KNN	82.90	90.79	94.24	78.43	86.59
AMVP (93%) + KNN	82.84	90.73	94.30	78.43	86.58
AMVP (95%) + KNN	83.15	90.85	94.36	78.62	86.75
AMVP (97%) + KNN	82.97	90.98	94.42	79.11	86.87
AMVP (99%) + KNN	83.76	91.28	94.61	79.47	87.28
AMVP (91%) + RF	53.68	68.45	65.32	50.37	59.46
AMVP (93%) + RF	54.11	67.53	65.63	50.06	59.33

Table 8 (continued)

Methods	Pose No.				Avg
	05	07	09	29	
AMVP (95%) + RF	53.98	69.18	64.95	49.51	59.41
AMVP (97%) + RF	54.90	67.04	65.07	48.47	58.87
AMVP (99%) + RF	53.55	67.16	63.60	49.20	58.38
Ours (AMVP (91%) + WSRC)	82.17	94.72	94.67	78.74	87.58
Ours (AMVP (93%) + WSRC)	82.90	94.78	95.28	78.92	87.97
Ours (AMVP (95%) + WSRC)	84.74	94.54	95.10	79.23	88.40
Ours (AMVP (97%) + WSRC)	85.36	95.27	95.89	80.64	89.29
Ours (AMVP (99%) + WSRC)	87.38	96.01	96.08	83.21	90.67

The bold number represents the highest accuracy under different methods

In the future, we will extend our work to solve face recognition with larger angles of illumination and pose changes: for instance, (1) to combine with other more effective illumination pretreatment techniques; (2) to combine with pose correction methods or frontal face synthesis; (3) to extract more robust illumination and pose invariant features; (4) to use more discriminated and faster classifiers; (5) to use other hybrid methods.

Acknowledgements This work has been supported by National Natural Science Foundation of China (61203261), China Postdoctoral Science Foundation funded project (2012M521335), Jiangsu Key Laboratory of Big Data Analysis Technology/B-DAT (Nanjing University of Information Science & Technology, Grant No.: KXK1404), Research Fund of Guangxi Key Laboratory of Multi-source Information Mining & Security (MIMS16-02) and the Fundamental Research Funds of Shandong University (2015JC014 and 2017JC043). We would also like to thank the ORL database of faces captured by AT & T Cambridge Laboratory, the Yale face database and the Yale face database B provided by the Center for Computational Vision and Control at Yale University and the CMU-PIE database from Carnegie Mellon University, respectively.

Compliance with ethical standards

Conflict of interest We declare that we have no conflict of interest.

References

- Garcia C, Delakis M (2004) Convolutional face finder: a neural architecture for fast and robust face detection. *IEEE Trans Pattern Anal Mach Intell* 26(11):1408–1423
- Chellappa R, Wilson CL, Sirohey S (1995) Human and machine recognition of faces: a survey. *Proc IEEE* 83(5):705–741
- Zhao W, Chellappa R, Phillips PJ et al (2003) Face recognition: a literature survey. *ACM Comput Surv (CSUR)* 35(4):399–458
- Cao X, Shen W, Yu LG et al (2012) Illumination invariant extraction for face recognition using neighboring wavelet coefficients. *Pattern Recogn* 45(4):1299–1305
- Yun SH, Kim JH, Kim S (2010) Image enhancement using a fusion framework of histogram equalization and laplacian pyramid. *IEEE Trans Consum Electron* 56(4):2763–2771
- Tan X, Triggs B (2010) Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Trans Image Process* 19(6):1635–1650
- Basri R, Jacobs DW (2003) Lambertian reflectance and linear subspaces. *IEEE Trans Pattern Anal Mach Intell* 25(2):218–233
- Gudur N, Asari V (2006) Gabor wavelet based modular PCA approach for expression and illumination invariant face recognition. In: 35th IEEE applied imagery and pattern recognition workshop (AIPR'06). IEEE, pp 13–18
- Roy H, Bhattacharjee D (2016) Local-gravity-face (LG-face) for illumination-invariant and heterogeneous face recognition. *IEEE Trans Inf Forensics Secur* 11(7):1414–1424
- Ramaiah NP, Ijjin EP, Mohan CK (2015) Illumination invariant face recognition using convolutional neural networks. In: 2015 IEEE international conference on signal processing, informatics, communication and energy systems (SPICES). IEEE, pp 1–4
- Beymer DJ (1994) Face recognition under varying pose. In: IEEE computer society conference on computer vision and pattern recognition (CVPR). IEEE, pp 756–761
- Kim TK, Kittler J (2005) Locally linear discriminant analysis for multimodally distributed classes for face recognition with a single model image. *IEEE Trans Pattern Anal Mach Intell* 27(3):318–327
- Gross R, Matthews I, Baker S (2004) Appearance-based face recognition and light-fields. *IEEE Trans Pattern Anal Mach Intell* 26(4):449–465
- Gonzalez-Jimenez D, Alba-Castro JL (2007) Toward pose-invariant 2-D face recognition through point distribution models and facial symmetry. *IEEE Trans Inf Forensics Secur* 2(3):413–429
- Blanz V, Romdhani S, Vetter T (2002) Face identification across different poses and illuminations with a 3d morphable model. In: Fifth IEEE international conference on automatic face and gesture recognition. IEEE, pp 192–197
- Chai X, Shan S, Chen X et al (2007) Locally linear regression for pose-invariant face recognition. *IEEE Trans Image Process* 16(7):1716–1725
- Castillo CD, Jacobs DW (2011) Wide-baseline stereo for face recognition with large pose variation. In: IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 537–544
- Li A, Shan S, Gao W (2012) Coupled bias-variance tradeoff for cross-pose face recognition. *IEEE Trans Image Process* 21(1):305–315
- Chen D, Cao X, Wen F, et al (2013) Blessing of dimensionality: high-dimensional feature and its efficient compression for face

- verification. In: IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 3025–3032
20. Li S, Liu X, Chai X, et al (2012) Morphable displacement field based image matching for face recognition across pose. In: European conference on computer vision. Springer, Berlin Heidelberg, pp 102–115
 21. Asthana A, Marks TK, Jones MJ et al. (2011) Fully automatic pose-invariant face recognition via 3D pose normalization. In: International conference on computer vision. IEEE, pp 937–944
 22. Jiang D, Hu Y, Yan S et al (2005) Efficient 3D reconstruction for face recognition. *Pattern Recogn* 38(6):787–798
 23. Ghorbel A, Tajouri I, Aydi W, et al. (2016) A comparative study of GOM, uLBP, VLC and fractional Eigenfaces for face recognition. In: 2016 International conference on image processing, applications and systems (IPAS). IEEE, pp 1–5
 24. Shu C, Ding X, Fang C (2011) Histogram of the oriented gradient for face recognition. *Tsinghua Sci Technol* 16(2):216–224
 25. Tan H, Yang B, Ma Z (2014) Face recognition based on the fusion of global and local HOG features of face images. *IET Comput Vis* 8(3):224–234
 26. Bhele SG, Mankar VH (2015) Recognition of faces using discriminative features of LBP and HOG descriptor in varying environment. In: 2015 International conference on computational intelligence and communication networks (CICN). IEEE, pp 426–432
 27. Parkhi OM, Vedaldi A, Zisserman A (2015) Deep face recognition. In: British machine vision conference, pp 1–12
 28. Shi S, Yang C, Wang T et al (2013) Face image enhancement based on self-quotient Image. *Comput Eng Appl* 49(13):142–144
 29. Moghaddam B, Pentland AP (1994) Face recognition using view-based and modular eigenspaces. In: SPIE's 1994 international symposium on optics, imaging, and instrumentation. international society for optics and photonics, pp 12–21
 30. Cosma C, Brehar R, Nedeveschi S (2013) Pedestrians detection using a cascade of LBP and HOG classifiers. In: IEEE international conference on intelligent computer communication and processing (ICCP). IEEE, pp 69–75
 31. Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans Pattern Anal Mach Intell* 24(7):971–987
 32. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: IEEE computer society conference on computer vision and pattern recognition (CVPR). IEEE, pp 886–893
 33. Shi W (2016) Facial expression recognition based on adaptive weighted LBP and collaborative representation. Nanjing: Nanjing University of Posts and Telecommunications, M. S. Thesis
 34. Fan Z, Ni M, Zhu Q et al (2015) Weighted sparse representation for face recognition. *Neurocomputing* 151(1):304–309
 35. Yu K, Zhang T (2010) Improved local coordinate coding using local tangents. In: Proceedings of the 27th international conference on machine learning (ICML-10), pp 1215–1222
 36. Lu CY, Min H, Gui J et al (2013) Face recognition via weighted sparse representation. *J Vis Commun Image Represent* 24(2):111–116
 37. AT&T Laboratories Cambridge, “The Database of Faces” [DB/OL]. Available: <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>
 38. UCSD Computer Vision, “Yale Face Database” [DB/OL]. Available: <http://vision.ucsd.edu/content/yale-face-database>
 39. Belhumeur PN, Hespanha JP, Kriegman DJ (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Trans Pattern Anal Mach Intell* 19(7):711–720
 40. Baker S, Sim T, Bsat M (2003) The CMU pose, illumination, and expression database. *IEEE Trans Pattern Anal Mach Intell* 25(12):1615–1618