Contents lists available at ScienceDirect

The Journal of Systems and Software

Face recognition based on curvelets and local binary pattern features via using local property preservation

Lijian Zhou[a,b,*], Wanquan Liu[b], Zhe-ming Lu[c], Tingyuan Nie[a]

ARTICLE INFO

Keywords:
Face recognition
Curvelet transform
Local binary pattern
Local property preservation

ABSTRACT

In this paper, we propose a new feature extraction approach for face recognition based on Curvelet transform and local binary pattern operator. The motivation of this approach is based on two observations. One is that Curvelet transform is a new anisotropic multi-resolution analysis tool, which can effectively represent image edge discontinuities; the other is that local binary pattern operator is one of the best current texture descriptors for face images. As the curvelet features in different frequency bands represent different information of the original image, we extract such features using different methods for different frequency bands. Technically, the lowest frequency band component is processed using the local binary pattern method, and only the medium frequency band components are normalized. And then, we combine them to create a feature set, and use the local preservation projection to reduce its dimension. Finally, we classify the test samples using the nearest neighbor classifier in the reduced space. Extensive experiments on the Yale database, the extended Yale B database, the PIE pose 09 database, and the FRGC database illustrate the effectiveness of the proposed method.

1. Introduction

Feature extraction is a key step in many classification tasks including face recognition. Over the past several decades, various multi-resolution analysis methods have been developed to represent signal features for different purposes, such as wavelets, contourlets, curvelets, and so on. Wavelet and its related classical multi-resolution methods can effectively represent the image information at all scales and locations, but they are isotropic, which cannot deal with the anisotropic property properly in face representation. Curvelet transform, developed by Candes and Donoho (2000), is a new anisotropic multi-resolution approach, and it can effectively represent edge discontinuities in face images. In a recent work, Majumdar (2007) showed curvelets perform well for some pattern recognition problems such as character recognition. Curvelet based face recognition has also been investigated by Mandal et al. (2007) and Zhang et al. (2011). In Mandal et al. (2007), curvelet transform is employed to extract features from bit quantized facial images, and proved to outperform the wavelet based approaches. Zhang et al. (2011) studied the approaches using wavelet, Gabor and Curvelet transforms for face recognition and demonstrated that Curvelet transform has better performance than other methods when the expression and illumination change randomly. In Mohammed et al. (2009), Song et al. (2010), El Aroussi et al. (2009), approaches based on Kernel Principal Component Analysis (KPCA), the boosting technique with linear discriminant analysis (B-LDA) and fuzzy discriminant analysis have been developed with curvelet features. In Zhang et al. (2007), a SVM based face recognition approach was developed, which uses Curvelet coefficients without any dimensionality reduction. The work Majumdar and Bhattacharya (2007) used the bit-quantized images to extract curvelet features in different resolutions, which is an extension of Mandal et al. (2007). In Aroussi et al. (2009) and Mandal et al. (2009), dimension reduction methods are proposed. For example, LDA is used in Aroussi et al. (2009), and Principal Component Analysis (PCA) and LDA are both used in Mandal et al. (2009). To the best of our knowledge, all these approaches either used the curvelet coefficients directly or simply combined with some linear dimension reduction techniques. In fact, the anisotropic characteristic of curvelet is not utilized appropriately in terms of the structure representation for face images.

As we know that the local binary pattern (LBP) operator Ojala et al. (1996) is one of the best current texture descriptors, which has been used extensively in various applications. LBP has been

* Corresponding author at: School of Communication and Electronic Engineering, Qingdao Technological University, Qingdao 266033, PR China.
Tel.: +86 13515328316.
E-mail addresses: zhoulijian@qtech.edu.cn, zhoulj_qd@163.com (L. Zhou).

http://dx.doi.org/10.1016/j.jss.2014.04.037
0164-1212/

proven to be highly discriminative and its key advantages lie in its invariance to monotonic gray level changes and computational efficiency, making it suitable for some demanding image analysis tasks. Heusch et al. (2006) proposed an image preprocessing approach using LBP for face authentication. Also Zhang et al. (2010) presented the local derivative pattern (LDP) technique for face recognition, which can capture more detailed information than LBP. Tan and Triggs (2010) proposed a local ternary patterns (LTP), which is a generalization of the LBP. Suruliandi et al. (2012) compared the LBP, LTP, LDP and the other improved LBP methods, which demonstrated the LTP and LDP are better than the other methods for face recognition. Among these techniques, the directly extracted features are used to recognize the face images. A real time face recognition using curvelet transform and complete loca1 binary pattern (CLBP) are proposed by Sirshendu Arosh et al. (2012), in which an image is decomposed into the curvelet sub-band components in three different resolutions and the descriptive feature sets are extracted using the CLBP method. In another related work, a facial expression recognition approach using curvelet based local binary patterns Saha et al. (2010) is proposed, which can recognize the facial expression better. These methods used all curvelet sub-band components processed by the CLBP or LBP to compute their histogram for recognition. By analyzing the characteristics of all curvelet frequency band components of an image and the texture descriptors, we can see that all the other sub-band components except from the lowest frequency band represent the detail information in different scales and the selected texture descriptors should reflect the detail information effectively. Therefore it is not necessary for us process all the high frequency sub-band images equally when using the LBP. So we adopt different methods to process different curvelet sub-band images in this paper.

Face images are usually in high dimension which is not suitable for robust and fast face recognition directly. Dimension reduction technique is an important method to solve this problem. Two of the most popular techniques in dimension reduction are PCA Turk and Pentland (1991) and LDA Belhumeur et al. (1997). However, both PCA and LDA only use the Euclidean metric and they usually fail to discover the underlying structure if face images lie on a nonlinear sub-manifold. He et al. (2005) proposed a face recognition approach using the Laplacianfaces, which explicitly investigates a specific manifold structure in terms of local property preservation (LPP). LPP is modeled by a nearest-neighbor graph which preserves the local structure of the image space. So people can obtain a face subspace for each individual and each face image is mapped into a low-dimensional face sub-space, which is characterized by a set of feature images, called the Laplacianfaces.

In this paper, we propose a new feature extraction approach for face recognition based on previous analysis. First we decompose the face image using Curvelet Transform. Then we extract the features in different ways for different frequency bands in curvelet frequency domain because the information contents are different in different frequency bands. The highest frequency band (Part 3 as explained in Section 2) mainly represents the noisy information as expected, so we remove it directly. The lowest frequency band (Part 1) represents the main structure of a face image and we will process it more precisely to reflect the face structures by using the LBP method. The information on the other frequency bands consists mainly of edge structure, we perform normalization on them in order to reflect the edge changes more clearly. Consequently, all these information are combined to build an elementary feature set. Finally, we reduce their dimension using LPP to obtain the Laplacian features, and classify the test sample using the nearest neighbor(NN) classifier. Extensive experiments on several benchmark datasets including the Yale database, the extended Yale B database, the PIE database and the FRGC dataset consistently show the effectiveness of our approach.

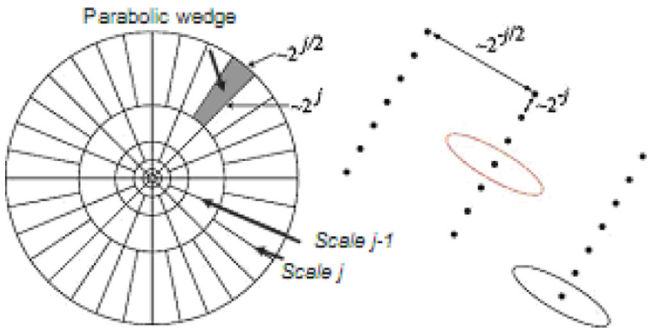

Fig. 1. Curvelets in Fourier frequency (left) and spatial domain (right) (Saha et al., 2010).

The organization of the paper is as follows. A review of related results is presented in Section 2, including the curvelet analysis, LBP and LPP. A concise description of the proposed approach is described in Section 3. The experimental results and analysis are presented in Section 4, and finally the conclusion is given in Section 5.

2. Related works

In this section, we introduce some techniques which are required in Section 3, including Curvelet Transform, LBP and LPP.

2.1. Curvelet transform

2.1.1. 2D continuous-time curvelet transform

Let us consider a 2D signal in R^2 with spatial variable $x(x_1, x_2)$, where x_1, x_2 are the rectangular coordinates. Let r and θ represent polar coordinates in frequency-domain. A continuous two dimensional (2D) curvelet transform can be represented by a pair of radial and angular windows $(W(r), V(t))$, which are all smooth. The radial windows $W(r)$ are positive real arguments and have a support on $r \in (1/2, 2)$, The angular windows $V(t)$ are the real arguments with a support on $t \in (-1, 1)$. These windows will always obey the following admissibility conditions:

$$\sum_{j=-\infty}^{\infty} W^2(2^j r) = 1,\ r \in (1/2, 2) \tag{1}$$

$$\sum_{l=-\infty}^{\infty} V^2(t-l) = 1, t \in (-1, 1) \tag{2}$$

where j is the curvelet decomposition scale level, and l stands for the translation. The left part of Fig. 1 shows the division of wedges of the Fourier frequency plane and the right one represents curvelets in the spatial Cartesian grid associated with a given scale and orientation Candels et al. (2005).

From Fig. 1, we can see that a polar wedge window U_j is supported by the radial and angular windows $W(r)$, $V(t)$. U_j can be defined in the Fourier domain by

$$U_j(r, \theta) = 2^{-3j/4} W(2^{-j} r) V\left(\frac{2^{\lfloor j/2 \rfloor}}{2\pi}\theta\right), \quad \forall j \geqslant 0 \tag{3}$$

where $\lfloor j/2 \rfloor$ is the integer part of $j/2$.

For simplicity, we take $U_j(\omega_1, \omega_2)$ as $U_j(\omega)$, which is the waveform $\varphi_j(x)$'s Fourier transform, i.e. $\hat{\varphi}_j(\omega) = U_j(\omega)$. Then the Curvelet transform of a space function $x = (x_1, x_2)$ can be defined as a function at scale 2^{-j}, orientation θ_t, and position $x_k^{(j,l)}$ by

$$\phi_{j,k,l}(x) = \phi_j(R_{\theta_t}(x - x_k^{j,l})) \tag{4}$$

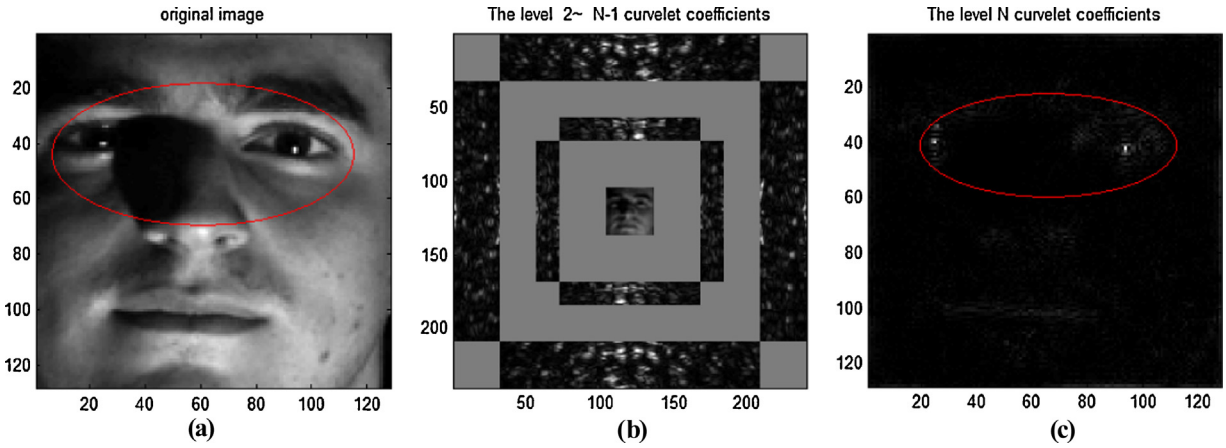

Fig. 2. The original image and its curvelet transform.

where R_θ is the rotation in θ radians, $k=(k_1,k_2)\in Z^2$ is the sequence of translation parameters. A curvelet coefficient is simply calculated by computing the inner product of the image f and $\phi_{j,k,l}(x)$. The reference Candels et al. (2005) has two different digital fast implementations: curvelets via Unequally Spaced Fast Fourier Transform and curvelets via Fast Wrapping Discrete Curvelet Transform. These new discrete curvelet transforms are simpler, faster and less redundant compared to their first generation versions. We take the second method in this paper, which is described as follows.

2.1.2. Fast wrapping discrete curvelet transform

In the continuous-time representation, the window U_j smoothly extracts frequencies near the dyadic corona $\{2^j\le r\le 2^{j+1}\}$ and near the angle $\{-\pi\cdot 2^{-j/2}\le\theta\le\pi\cdot 2^{-j/2}\}$. Coronae and rotations are not especially adapted to Cartesian arrays. Instead, it is convenient to replace these concepts by Cartesian equivalents; the Cartesian coronae $\tilde{U}_j(\omega)$ are based on concentric squares (instead of circles) and shears. The corresponding discrete area $\tilde{U}_j[n_1,n_2]$ is supported on some rectangle with length $L_{1,j}$ and width $L_{2,j}$ as below

$$P_j=\left\{(n_1,n_2): n_{1,0}\le n_1\le n_{1,0}+L_{1,j}, n_{2,0}\le n_2\le n_{2,0}+L_{2,j}\right\} \quad (5)$$

The wrapping computation of the array $\tilde{U}_{j,l}$ is defined as

$$W(\tilde{U}_{j,l}[n_1 \bmod L_{1,j}, n_2 \bmod L_{2,j}])=\tilde{U}_{j,l}[n_1,n_2] \quad (6)$$

where l stands for translation.

Let's take the original $n\times n$ image as $f(n_1,n_2)$, where n_1, n_2 are the rectangular coordinates. $\hat{f}[n_1,n_2]$ denote its 2D discrete Fourier transform. The detailed fast calculation of 2D curvelet transform Candels et al. (2005) can be described as follows.

(1) Apply 2D FFT and obtain Fourier samples $\hat{f}[n_1,n_2]$, where $-n/2\le n_1,n_2\le n/2$.

(2) For each scale j and angle l, calculate the product $\tilde{U}_{j,l}[n_1,n_2]\hat{f}[n_1,n_2]$.

(3) Wrap this product around the origin and obtain $\tilde{f}_{j,l}[n_1,n_2]=W(\tilde{U}_{j,l}\hat{f})[n_1,n_2]$, where n_1 and n_2 are in the range of $0\le n_1\le L_{1,j}$ and $0\le n_2\le L_{2,j}$ separately.

(4) Apply the inverse 2D FFT to each $\tilde{\mathbf{f}}_{j,l}$, and obtain the discrete coefficients.

We take a face image from the Yale dataset with resolution 64 × 64 as an example in Fig. 2(a). The curvelet decomposition coefficients are shown in Fig. 2, in which Fig. 2(b) is the lower frequency information and Fig. 2(c) is the redundant high frequency information. The curvelet coefficients in Fig. 2(b) are described as follows.

(1) The low frequency (coarse scale) coefficients are stored at the center of the display (Part 1).

(2) The Cartesian concentric coronae represents the coefficients at different scales, and the outer coronae corresponds to higher frequencies (Part 2).

We can see that the main information of the face is located in Fig. 2 (b). From the circle regions in Fig. 2(a) and Fig. 2 (c), regions in Fig. 2(c) contain some noisy information caused by the environment, which usually is white noise or other sudden changes. In this example, the frequency information in the reflection point of the light is induced by the sudden changes as shown in the circled regions of Fig. 2 (a) and Fig. 2 (c).

2.2. Logarithm computation and LBP

2.2.1. Logarithm computation

The aim of Logarithm (LOG) computation is to remove the dominance of some pixel values in an image and its computation equation is as follows (Tan and Triggs, 2010),

$$l(x,y)=\frac{\log(1+\alpha F(x,y))}{1+\alpha\cdot 255} \quad (7)$$

where $l(x,y)$ is the result of the logarithm computation for each pixel $F(x,y)$ and every pixel value is added by 1 in (7) in order to avoid 0 in logarithm computation. In addition, the regulation factor α is introduced for reducing the effect of illumination and enhancing the image. In general, α is a constant between 1 and 256 and is taken between 200 and 250 in this paper because it spreads out better for different gray values and can produce a similar value for the same gray value in this range. This preprocessing will be used in the proposed approach for the first part decomposition of curvelet transform, which is shown in Section 3.

2.2.2. LBP

The LBP operator is proposed by Ojala et al. (1996) in 1996, which is a powerful tool for texture description. The operator labels the image pixels by thresholding their $N\times N$-neighborhood of each pixel with the center value and giving the local binary image descriptor. For a 3 × 3-neighborhood, the LBP value on a point (x_c, y_c) can be calculated using the LBP operator around the center point (x_c, y_c) as follows.

$$LBP(x_c,y_c)=\sum_{n=0}^{7}2^n s(i_n-i_c), \quad (8)$$

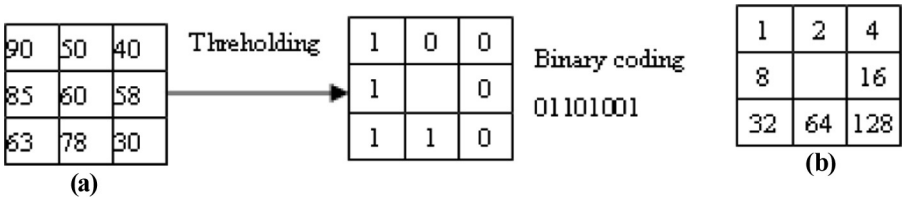

Fig. 3. LBP process.

where n presents the serial number of the neighbor points around the center from left to right line by line, i_c is the pixel value of the center point, i_n is the pixel value of the neighbor point and

$$s(u) = \begin{cases} 1 & u \geqslant 0; \\ 0 & u < 0. \end{cases} \tag{9}$$

The LBP process is shown in Fig. 3(a), and the weight value graph is presented in Fig. 3(b).

We take an image as example from the extended Yale B dataset, and process it using LOG and LBP operations. A 3×3 LBP operator is used in this paper. The results are shown in Fig. 4, where Fig. 4(a) is the original image, Fig. 4(b) is the processed image after LOG and LBP. From these images, we can see that Fig. 4(b) can describe the details of the face. So we will perform the same process for the lowest frequency information based on an image's curvelet representation.

2.3. LPP

PCA and LDA aim to preserve the global structure. However, in many real-world applications, the local structure is more important since its description can represent face characteristics more effectively than the global description in face recognition. So we decide to use the Locality Preserving Projection (LPP) (Sirshendu Arosh et al., 2012) to reduce the dimension of face image features. The complete derivation and theoretical justifications of LPP can be found in He and Niyogi (2003) and here we only give a brief introduction. LPP seeks to preserve the intrinsic geometry of the data in local structure and its objective function is as follows:

$$\min \sum_{i,j} \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 S_{ij} \tag{10}$$

where $\mathbf{y}_i$ is the reduced-dimensional representation of the original image $\mathbf{x}_i$ and $\mathbf{S}$ is a structure similarity matrix. A possible way of choosing $\mathbf{S}$ is as follows:

$$S_{ij} = \begin{cases} \exp(-\left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 / t) & \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 < \varepsilon; \\ 0 & \text{otherwise}. \end{cases} \tag{11}$$

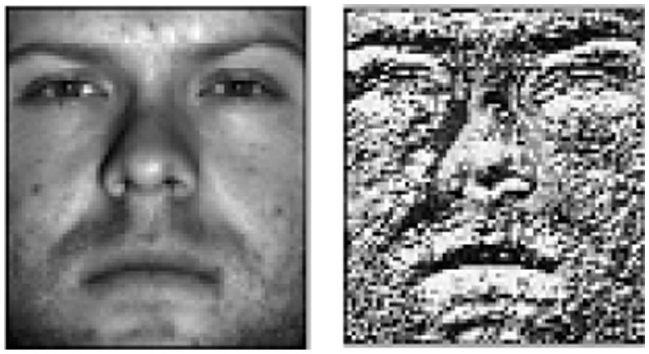

Fig. 4. Original image and the processed image by LOG + LBP.

where ε is a sufficiently small positive number. $\left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 < \varepsilon$ indicates $\mathbf{x}_i$ ($\mathbf{x}_j$) is among the nearest neighbors of $\mathbf{x}_j(\mathbf{x}_i)$, in which ε defines the radius of the local neighborhood. The objective function with our choice of symmetric weights S_{ij} with ($S_{ij} = S_{ji}$) incurs a heavy penalty if neighboring points $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart, i.e., if $(\mathbf{y}_i - \mathbf{y}_j)^2$ is large. Therefore, minimizing such objective function is an attempt to ensure that, if $\mathbf{x}_i$ and $\mathbf{x}_j$ are close, then $\mathbf{y}_i$ and $\mathbf{y}_j$ should be close as well. Following some simple algebraic manipulations, we can obtain

$$\begin{aligned} \frac{1}{2}\sum_{i,j} \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 S_{ij} &= \frac{1}{2}\sum_{i,j} |\mathbf{w}^{\mathrm{T}}\mathbf{x}_i - \mathbf{w}^{\mathrm{T}}\mathbf{x}_j|^2 S_{ij} \\ &= \sum_{i,j} \mathbf{w}^{\mathrm{T}}\mathbf{x}_i S_{ij} \mathbf{x}_i^{\mathrm{T}}\mathbf{w} - \sum_{i,j} \mathbf{w}^{\mathrm{T}}\mathbf{x}_i S_{ij} \mathbf{x}_j^{\mathrm{T}}\mathbf{w} \\ &= (\sum_{i,i} \mathbf{w}^{\mathrm{T}}\mathbf{x}_i D_{ii} \mathbf{x}_i^{\mathrm{T}}\mathbf{w}) - \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{S}\mathbf{X}^{\mathrm{T}}\mathbf{w} \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{D}\mathbf{X}^{\mathrm{T}}\mathbf{w} - \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{S}\mathbf{X}^{\mathrm{T}}\mathbf{w} \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{X}(\mathbf{D} - \mathbf{S})\mathbf{X}^{\mathrm{T}}\mathbf{w} \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{L}\mathbf{X}^{\mathrm{T}}\mathbf{w} \end{aligned}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$, and $\mathbf{w}$ is a projection. $\mathbf{D}$ is a diagonal matrix with its entries being column (or row since $\mathbf{S}$ is symmetric) sums of $\mathbf{S}$, and $\mathbf{L} = \mathbf{D} - \mathbf{S}$ is the Laplacian matrix. The matrix $\mathbf{D}$ provides a natural measure on the data points. The larger the values D_{ii} (corresponding to $\mathbf{y}_i$) are, the more important $\mathbf{y}_i$ will be. Also, we impose a constraint as follows:

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}\mathbf{y} = 1 \Rightarrow \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{D}\mathbf{X}^{\mathrm{T}}\mathbf{w} = 1 \tag{12}$$

Finally, the minimization problem reduces to

$$\underset{\mathbf{w}}{\operatorname{argmin}} \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{D}\mathbf{X}^{\mathrm{T}}\mathbf{w} \operatorname{st.} \mathbf{w}^{\mathrm{T}}\mathbf{X}\mathbf{D}\mathbf{X}^{\mathrm{T}}\mathbf{w} = 1 \tag{13}$$

The transformation vector $\mathbf{w}$ that minimizes the objective function is given by the minimum eigenvalue solution to the following generalized eigenvalue problem:

$$\mathbf{X}\mathbf{L}\mathbf{X}^{\mathrm{T}}\mathbf{w} = \lambda \mathbf{X}\mathbf{D}\mathbf{X}^{\mathrm{T}}\mathbf{w} \tag{14}$$

Note that the two matrices $\mathbf{X}\mathbf{L}\mathbf{X}^{\mathrm{T}}$ and $\mathbf{X}\mathbf{D}\mathbf{X}^{\mathrm{T}}$ are both symmetric and positive semi-definite since the Laplacian matrix $\mathbf{L}$ and the diagonal matrix $\mathbf{D}$ are both symmetric and positive semi-definite. The Laplacian matrix for a finite graph is analogous to the Laplace Beltrami operator on compact Riemannian manifolds. Belkin and Niyogi Belkin and Niyogi (2001) showed that the optimal map preserving locality can be found by solving an optimization problem on the manifold and the detailed LPP algorithm can be found in Sirshendu Arosh et al. (2012).

3. The proposed approach

With above mentioned techniques and their advantage analysis, we will address the proposed approach in this section. As we know that Curvelet is a powerful tool to describe the changes in all directions with any scales. So for an image, we first perform the Curvelet transform on it and obtain three parts of its coefficients as described in the middle part of Fig. 5. Instead of using these

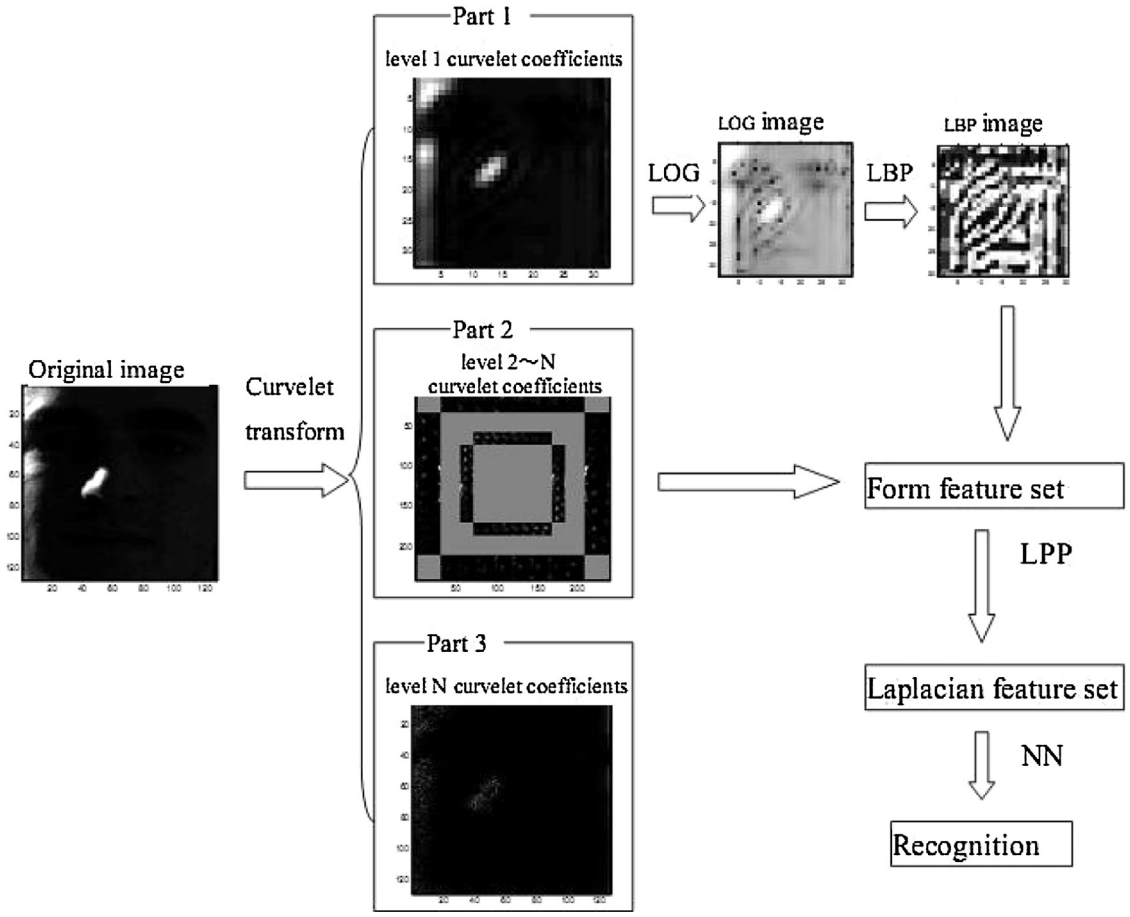

Fig. 5. The face recognition diagram of the proposed method.

coefficients directly as given in the existing literature, we analyze each part carefully and use them differently. In detail, we divide all coefficients into three parts, where Level 1 subimage is the first part (Part 1), Level 2, 3, $\cdots$, $N-1$ subimages are the second part (Part 2), and Level N is the third part (Part 3). The Part 1 is the lowest frequency component of the image, which represents the main structure of a face. The Part 2 is the middle frequency component of the image, which represents edge details of a face. The Part 3 is the highest frequency component of the image, which mainly represents noise. For extraction of image features, we process these coefficients using different methods. For Part 1, we extract the feature by enhancing its texture information and removing the background impact by using LOG + LBP. For Part 2, we keep the original data and only normalize it because it mainly contains the edge detail information of an image. The normalization is done in order to avoid specific image value dominance. For Part 3, we delete it directly because it only contains some noises with very few image feature information.

Though the features are obtained in different frequency bands with different approaches, their concatenation is still with high dimension. Then we use the LPP method to reduce its dimension. Finally, we can recognize a test face using the nearest neighbour classifier. The detail calculation steps are described as follows with a diagram shown in Fig. 5

Step 1: Perform the Curvelet transform for all M original training images according to Candels et al. (2005), and decompose each image into $N\,(N = \lfloor \log_2^K - 3 \rfloor)$ levels, where K is the minimum value of the row and column numbers of the original image. Take the ith original image as $f_i(n_1, n_2)$, where n_1, n_2 are the rectangular coordinates, $1 \leqslant i \leqslant M$.

(1) Apply 2D FFT to the original image f_i and obtain Fourier coefficients $\hat{f}_i[n_1, n_2]$, $-n/2 \leq n_1, n_2 \leq n/2$.

(2) For each scale j and angle l, compute the product $\tilde{U}\hat{f}_i[n_1, n_2] = \tilde{U}_{j,l}[n_1, n_2]\hat{f}[n_1, n_2]$, where $1 \leq j \leq N$.

(3) Wrap this product around the origin and obtain $\tilde{f}_{ij,l}[n_1, n_2] = W(\tilde{U}_{j,l}\hat{f}_i)[n_1, n_2]$, where n_1 and n_2 are in the range of $0 \leq n_1 \leq L_{1,j}$ and $0 \leq n_2 \leq L_{2,j}$ separately.

(4) Apply the inverse 2D FFT to each $\tilde{f}_{ij,l}$, and obtain the discrete curvelet coefficients.

Step 2: Divide every training image's curvelet coefficients into 3 parts. Level 1 subimage F_{i1} is the first part (Part 1), Level 2,3, $\cdots$, N -1 subimages F_{i2} are the second part (Part 2), and Level N subimage F_{i3} is the third part (Part 3) for the ith original image, $1 \leqslant i \leqslant M$.

Step 3: Compute the logarithm value for the subimage $\mathbf{F}_{i1}$ in Level 1(Part 1) according to Eq. (7), and implement the LBP according to Eq. (8) to obtain the subimage $\hat{\mathbf{F}}_{i1}$. Finally, reshape it to form a row vector $\tilde{\mathbf{F}}_{i1}$ for the ith image, $1 \leqslant i \leqslant M$.

Step 4: Reshape $\mathbf{F}_{i2}$ to form a row vector $\tilde{\mathbf{F}}_{i2}$ for the ith image, $1 \leqslant i \leqslant M$. And form a matrix $\mathbf{Q}$. Take normalization by using the Eq. (15) to get a matrix $\tilde{\mathbf{F}}_2$,

$$\tilde{F}_{i2}(x, y) = \frac{Q(x, y) - \min_{x,y}(\mathbf{Q})}{\max_{x,y}(\mathbf{Q}) - \min_{x,y}(\mathbf{Q})}. \tag{15}$$

At last, reshape it to obtain a nomalized row vector $\tilde{\mathbf{F}}_{i2}$ for the ith image, $1 \leqslant i \leqslant M$.

Step 5: Construct a row vector by combining the results $\tilde{\mathbf{F}}_{i1}$ and $\tilde{\mathbf{F}}_{i2}$ of Step 3 and Step 4 to obtain the feature vector $\tilde{\mathbf{F}}_i$ for the ith image, $1 \leqslant i \leqslant M$.

Step 6: Implement the LPP method for the labelled training set using the method in Section 2.3 and get the final features.

Step 7: Identify the testing face image using the nearest neighbour classifier.

We should keep in mind that we only process the first two parts of curvelet coefficients and also they are processed differently based on their different intuitions. Instead of using the conventional PCA and LDA for dimension reduction, LPP is used in the proposed approach. We will demonstrate in next section that the integration of these techniques are not trivial in terms of performance.

4. Experimental results

To illustrate the effectiveness of the proposed method, we use the following benchmark datasets for experiments. (1) the Yale Face Dataset (http://www.zjucadcg.cn/dengcai/Data/DimensionReduction.html, 2014), which contains 165 grayscale images of 15 individuals. There are 11 images per subject, which are different in facial expression and configuration. (2) the Extended Yale B Face Dataset (http://cswww.essex.ac.uk/mv/allfaces/faces94.html, 2014), which contains 38 people with 64 different illumination conditions. (3) the PIE Dataset, which contains 68 people with 24 different illumination and expression conditions (Sim et al., 2003). (4) the FRGC Face Database (http://www.nist.gov/itl/iad/ig/frgc.cfm, 2014) with different illumination, gait, expressions. For the purpose of computation efficiency, all face images are manually cropped and resized to 64 × 64 gray images for all databases. We randomly take n images from every person as the training set, the others as the testing set in every experiment. We take the Average Recognition Accuracy (ARA) and Standard Deviation (SD) of the face recognition rate as the assessment criterion to evaluate the proposed approach performance. In this paper, ten random experiments are implemented to compute their Average Recognition Accuracy (ARA) and Standard Deviation (SD) for every dataset. For comparisons with other typical methods, we also perform experiments using the other six approaches except for the proposed method (Curvelet + LBP + LPP) for every dataset, which include Curvelet + LTP + LPP, Curvelet + LDP + LPP, Curvelet + LPP (using coefficients directly with LPP), Curvelet + LBP + PCA + LDA, Curvelet + PCA + LDA method (Mandal et al., 2009), and Curvelet + level1 + LBP + LPP (only using the lowest frequency band coefficients). The dimension is reduced to 30 for the LPP method. And the dimension is reduced automatically by keeping all the non-zero eigenvalues for the PCA + LDA method.

4.1. Results on the Yale face dataset

In the Yale Face Dataset, every person has different facial expressions and configurations. For these two conditions, we design the following experiments. We choose 1–10 training samples randomly, and the other images are for testing. The results are presented in Table 1. For convenience, we denote Curvelet as CT in the tables. The recognition results are also given in Fig. 6 for visually observing the difference of these seven methods. From Table 1 and Fig. 6, we can see that the recognition rates are consistently improved in all cases, and the recognition performance is stable according to the standard deviation. The recognition results are becoming better and better with using more training samples per person. Especially with a smaller number of training samples, the recognition rates are improved significantly for the LPP method. Also we notice that the recognition accuracy by using the LBP method is significantly better than those without using the LBP method. These results demonstrate that the proposed method can achieve better performance when dealing with the facial expression and configuration issues.

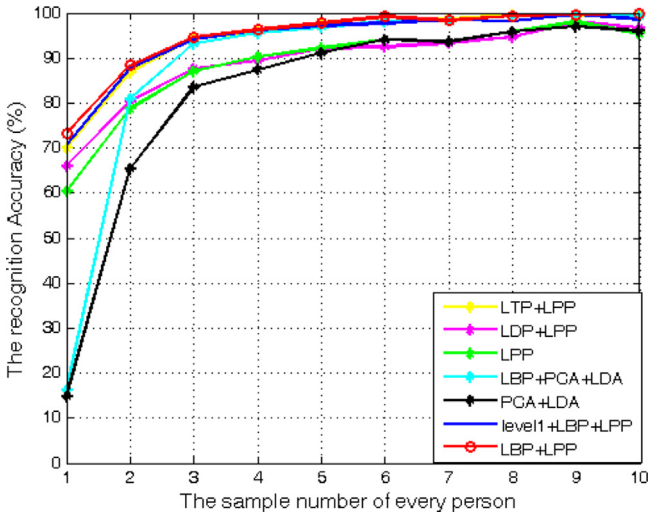

Fig. 6. Recognition results on the Yale face dataset.

4.2. Results on the extended Yale B face dataset

In the Extended Yale B face dataset, every person has different illumination conditions. We select 1–16 training samples randomly for training and the other images are for testing. The results are presented in Table 2. The recognition results are also illustrated in Fig. 7. From them, we can see that the recognition results are improved and also stable. Specifically, the approach based on LBP and LPP can achieve much better performance. This demonstrates that the proposed method can effectively solve the illumination issue for face recognition.

Table 1
Results on the Yale dataset.

n*	CT + LBP (LPP) ARA(SD)%	CT + LTP (LPP) ARA(SD)%	CT + LDP (LPP) ARA(SD)%	CT (LPP) ARA(SD)%	CT + LBP (PCA + LDA) ARA(SD)%	CT (PCA + LDA) ARA(SD)%	CT level 1 + LBP (LPP) ARA(SD)%
1	73.33(5.73)	69.93 (6.15)	66.07 (6.10)	60.47 (4.34)	16.33(6.01)	14.87 (3.53)	70.67 (4.84)
2	88.37(4.07)	86.74 (2.67)	80.30 (2.13)	78.74 (2.04)	80.89(3.71)	65.33 (3.42)	87.67 (3.95)
3	94.67(1.97)	94.33 (2.22)	87.58 (3.05)	87.00 (3.47)	93.17(2.63)	83.50 (5.27)	94.08 (1.98)
4	96.29(2.17)	95.71 (2.30)	89.33 (2.19)	90.29 (2.94)	95.52(2.16)	87.43 (2.83)	96.19 (2.11)
5	97.67(1.43)	97.89 (1.85)	92.11 (2.79)	92.22 (2.67)	96.89(1.72)	91.11 (2.96)	97.11 (2.23)
6	99.13(1.12)	99.07 (0.90)	92.53 (2.61)	94.13 (2.01)	98.00(1.13)	94.13 (1.69)	97.73 (1.67)
7	98.33(1.57)	98.83 (1.37)	93.33 (3.77)	93.67 (2.19)	98.33(1.92)	93.67 (2.70)	98.50 (2.00)
8	99.33(1.50)	99.56 (0.94)	94.67 (2.39)	95.78 (3.98)	99.33(1.50)	95.78 (3.39)	98.22 (1.41)
9	99.67(1.05)	99.33 (1.41)	98.00 (2.33)	98.00 (2.33)	99.67(1.05)	97.00 (2.46)	99.67 (1.05)
10	100.00(0.00)	100.00 (0.00)	96.67 (4.71)	95.33 (4.50)	99.33(2.11)	96.00 (4.66)	98.67 (2.81)

* The number of samples for every person.

Table 2
Results on the extended Yale B dataset.

n*	CT + LBP (LPP) ARA(SD)%	CT + LTP (LPP) ARA(SD)%	CT + LDP (LPP) ARA(SD)%	CT (LPP) ARA(SD)%	CT + LBP (PCA + LDA) ARA(SD)%	CT (PCA + LDA) ARA(SD)%	CT level 1 + LBP (LPP) ARA(SD)%
1	36.59 (2.68)	35.14 (2.27)	24.42 (2.17)	32.34 (2.75)	3.73(0.43)	3.10 (0.24)	34.10 (2.66)
2	58.88 (2.35)	53.25 (2.50)	41.17 (1.59)	50.21 (2.77)	45.29(1.72)	38.24 (2.44)	56.84 (2.31)
3	70.08 (1.23)	64.38 (1.86)	52.62 (1.04)	61.89 (1.63)	64.59(2.17)	53.97 (1.59)	68.05 (2.42)
4	78.08 (1.87)	72.17 (1.94)	60.55 (1.45)	69.31 (1.52)	75.00(2.11)	64.56 (2.00)	76.17 (1.74)
5	81.72 (1.39)	76.50 (1.74)	66.28 (1.49)	74.05 (1.46)	80.63(1.83)	70.95 (2.40)	79.62 (1.68)
6	85.61 (0.98)	80.70 (1.20)	72.25 (1.27)	78.31 (1.07)	85.12(1.30)	75.63 (1.33)	83.44 (1.93)
7	88.46 (1.05)	84.50 (1.33)	74.90 (1.73)	81.95 (1.17)	88.17(0.94)	80.79 (0.89)	85.31 (1.79)
8	88.89 (0.66)	85.25 (1.14)	76.99 (1.58)	82.71 (1.01)	89.00(0.94)	81.66(1.02)	86.65 (1.28)
9	90.82 (1.47)	88.40 (1.55)	78.47 (1.57)	85.81 (1.43)	90.67(0.74)	85.18 (1.49)	88.58 (1.17)
10	92.66 (0.89)	89.87 (1.24)	81.37 (1.41)	87.38 (1.31)	92.72(1.54)	87.07 (1.49)	88.87 (1.12)
11	92.96 (0.57)	91.16 (0.94)	82.62 (0.63)	88.70 (0.77)	92.80(1.10)	88.46 (0.71)	90.08 (1.38)
12	94.13 (0.73)	91.65 (1.17)	83.55 (1.19)	89.74 (1.00)	94.08(0.70)	89.57 (1.05)	90.59 (1.01)
13	94.43 (0.94)	93.25 (0.77)	84.33 (0.93)	91.14 (0.95)	94.69(0.84)	90.83 (0.94)	90.90 (0.91)
14	95.31 (0.59)	93.90 (1.05)	85.86 (1.28)	91.61 (0.80)	95.22(0.63)	91.58 (0.85)	91.33 (0.87)
15	95.75 (0.50)	94.83 (1.04)	86.17 (1.56)	92.38 (0.96)	95.70(0.50)	92.34 (1.13)	91.17 (0.56)
16	96.05 (1.25)	95.26 (1.15)	86.72 (1.05)	93.09 (1.48)	96.05(1.27)	93.19 (1.31)	91.06 (0.89)

* The number of samples for every person.

Table 3
Results on the PIE pose 09 dataset.

n*	CT + LBP (LPP) ARA(SD)%	CT + LTP (LPP) ARA(SD)%	CT + LDP (LPP) ARA(SD)%	CT (LPP) ARA(SD)%	CT + LBP (PCA + LDA) ARA(SD)%	CT (PCA + LDA) ARA(SD)%	CT level 1 + LBP (LPP) ARA(SD)%
1	60.67 (2.76)	62.09 (2.87)	29.76 (1.10)	50.27 (2.10)	2.43(0.61)	2.13 (0.24)	59.73 (2.13)
2	80.75 (1.50)	80.05 (1.67)	50.81 (3.09)	70.64 (2.08)	74.53(1.98)	67.87 (2.02)	77.61 (2.00)
3	87.22 (1.36)	86.16 (1.58)	62.51 (2.64)	78.12 (2.05)	84.60(1.21)	78.49 (2.01)	85.15 (1.09)
4	90.71 (0.83)	89.95 (0.80)	72.60 (3.23)	83.75 (1.55)	90.35(0.86)	85.72 (1.51)	88.57 (0.54)
5	92.02 (1.02)	91.06 (1.06)	77.56 (1.74)	86.01 (1.43)	91.72(1.11)	88.43 (1.42)	90.34 (1.03)
6	92.14 (0.95)	91.20 (0.67)	80.22 (1.37)	87.33 (0.96)	92.04(0.98)	89.40 (1.07)	91.32 (0.73)
7	93.19 (0.50)	92.54 (0.68)	82.69 (1.35)	88.83 (0.71)	93.10(0.48)	91.04 (0.81)	91.70 (0.68)
8	93.26 (0.73)	92.54 (0.55)	84.33 (0.95)	89.68 (0.84)	93.25(0.78)	91.58 (0.76)	92.18 (0.63)
9	93.33 (0.66)	92.30 (0.76)	85.31 (0.72)	89.78 (0.79)	93.27(0.77)	91.65 (0.68)	91.23 (0.65)
10	94.07 (0.57)	93.24 (0.50)	86.67 (0.81)	90.67 (0.80)	94.05(0.63)	92.57 (0.59)	90.82 (0.73)
11	93.36 (0.62)	92.44 (0.61)	86.52 (1.12)	89.93 (0.95)	93.30(0.78)	91.73 (0.81)	89.16 (0.83)
12	93.63 (0.74)	93.00 (0.85)	87.01 (1.03)	90.47 (1.12)	93.47(0.67)	92.57 (0.88)	84.82 (1.07)

* The number of samples for every person.

4.3. Results on the PIE pose 09 dataset

In the PIE pose 09 dataset, every person has different expression and illumination conditions. We take 1–12 training samples randomly, the other images from every person are for the testing. The results are presented in Table 3. The recognition results are also demonstrated in Fig. 8. From them, we can see that the recognition results are improved and also quite stable. One can see that LBP and LPP contribute significantly on the performance, especially in the case with a small number of training samples. This proves that the proposed approach can solve the expression and illumination problem in face recognition.

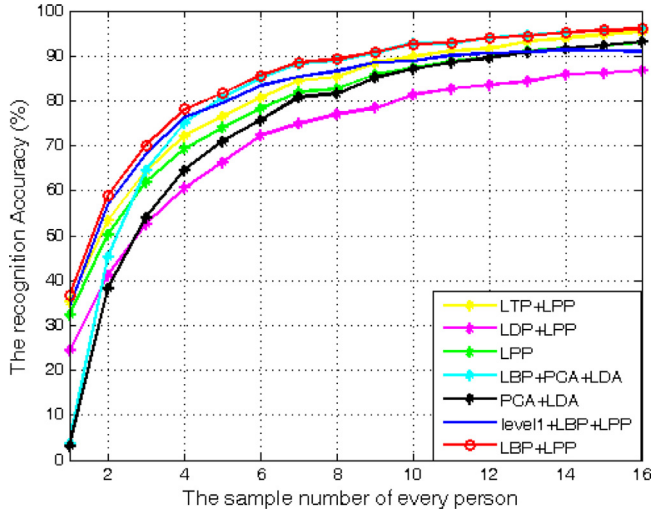

Fig. 7. Recognition results on the extended Yale B face dataset.

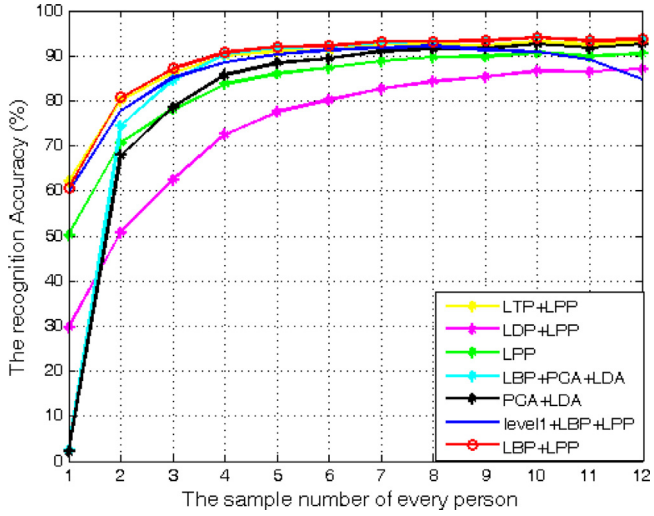

Fig. 8. Recognition results on the PIE Pose09 dataset.

Table 4
Results on the FRGC training and gallery dataset.

	CT + LBP (LPP)	CT + LTP (LPP)	CT + LDP (LPP)	CT (LPP)	CT + LBP (PCA + LDA)	CT (PCA + LDA)	CT level 1 + LBP (LPP)
RA(%)	99.65	80.90	79.34	94.79	99.65	96.70	99.33
T (s)	3.22e-004	3.22e-004	3.22e-004	3.22e-004	3.32e-004	3.32e-004	3.08e-004
D	30	30	30	30	39	39	30

4.4. Results on the FRGC data

For the FRGC dataset, we used 2560 images of 40 persons from its training set. And we take 576 images from the gallery set corresponding to the training set as the testing set. The recognition results (RA), the reduced dimension(D) and the time of recognizing a face image(T) are given in Table 4. From this table, we can see that the recognition results with LBP are improved significantly. The recognition time with LPP is slightly shorter than PCA + LDA. For convenient comparison, we compute the recognition time without using LPP for Curvelet + LBP features, the time is 0.0046s, which is longer than the method using the LPP method. In this table, we did not compute the SD value since we only did training and testing once due to the large size of this dataset. The reduced dimension size is selected based on the best performance.

It should be reminded that if we only use features from Part 2 on above datasets, we cannot achieve satisfactory recognition results since too much structure features are lost due to missing information from Part 1.

5. Conclusions

In this paper, a new approach on face recognition is proposed by using Curvelet transform, LBP and LPP. The main idea is to proces the curvelet information in different frequency bands with different techniques. Experiments are conducted on the Yale dataset, the Extended Yale B dataset, the PIE dataset and the FRGC database and the results validated the proposed approach consistently.

According Tables 1–4 and Figs. 6–8, it is proven that the proposed method in this paper can effectively recognize the face and has better recognition results than the method Mandal et al. (2009). Our contribution can be summarized as below. (i) Using the LPP approach with Curvelet features can significantly improve the recognition rate in comparison with using the PCA and LDA methods especially in case with a small number of training samples. (ii) Also using the LBP method can improve the recognition rate consistently. (iii) The proposed method can be used for face recognition problem with different facial expressions, configurations, and illuminations. (iv) The proposed approach can achieve very promising results in the case with a small number of training samples.

Acknowledgements

This work is supported by the National Natural Science Foundation of China with a grant No. 61171150 and the Zhejiang Province Natural Science Foundation of China with grant R1110006. It is also supported by an excellent teacher training plan in abroad from Shandong Province (2009).