Pattern Recognition 41 (2008) 3237 – 3243

Two-dimensional Laplacianfaces method for face recognition

Ben Niu [a,*], Qiang Yang [b], Simon Chi Keung Shiu [a], Sankar Kumar Pal [c]

[a] *Department of Computing, Hong Kong Polytechnic University, Hong Kong, China*
[b] *Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, China*
[c] *Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India*

Received 27 April 2006; received in revised form 21 May 2007; accepted 3 December 2007

Abstract

In this paper we propose a two-dimensional (2D) Laplacianfaces method for face recognition. The new algorithm is developed based on two techniques, i.e., locality preserved embedding and image based projection. The 2D Laplacianfaces method is not only computationally more efficient but also more accurate than the one-dimensional (1D) Laplacianfaces method in extracting the facial features for human face authentication. Extensive experiments are performed to test and evaluate the new algorithm using the FERET and the AR face databases. The experimental results indicate that the 2D Laplacianfaces method significantly outperforms the existing 2D Eigenfaces, the 2D Fisherfaces and the 1D Laplacianfaces methods under various experimental conditions.

Keywords: Feature extraction; Image based projection; Two-dimensional Laplacianfaces; Eigenfaces; Fisherfaces

1. Introduction

The Laplacianfaces method is a recently developed face recognition method [1]. It is a natural generalization of the locally linear embedding (LLE) [2] algorithm, which has been shown to be able to effectively handle the nonlinearity of the image space for dimensionality reduction. The main idea of the Laplacianfaces is to find out a low-dimensional representation of the data that can maximally preserve their locality, i.e., the pattern of distribution of the data in the local neighborhoods of the sample space. Differing from the Eigenfaces and the Fisherfaces, which search for the optimal projections by analyzing the global patterns of the data density, the Laplacianfaces method seeks its optimal solutions by examining closely the local geometry of the training samples. The features learned are thus quite effective in maintaining the locality of the training data, making it robust to the outlier samples for training and suitable for classification with neighborhood based k nearest neighbor method. The Laplacianfaces has been observed to significantly outperform the popular Eigenfaces and Fisherfaces methods on the Yale, the MSRA and the PIE face databases [1]. However, all of these methods are inefficient in that when handling the associated eigen equations, the computation and the memory complexities of the problem go up exponentially with the dimensionality of the training image vectors.

A number of researchers have attempted to improve the efficiency of Eigenfaces and Fisherfaces by using image based projection technique. Liu et al. [3] and Yang et al. [4] developed two-dimensional (2D) Eigenfaces, whereas Yang et al. [5,6], Xiong et al. [7] and Jing et al. [8] developed the 2D Fisherfaces methods. Both of these methods dramatically reduced the complexity of the algorithms from $O(m^2 \times n^2)$ to $O(m^2)$, or $O(n^2)$. These methods also reduced the size of the matrices in the eigen equations, allowing them to be more accurately evaluated. The objective function in the algorithm hence can be fully optimized to improve the classification accuracy [4]. While the 2D Eigenfaces and the Fisherfaces methods are effective, it is unclear whether the image based projection technique can also be applied effectively to improve the performance of the Laplacianfaces method. In this paper, we use this technique to develop the 2D Laplacianfaces method. In extensive experiments on a variety of databases, this new face recognition

* Corresponding author. Tel.: +86 852 27667312.
E-mail addresses: csniuben@comp.polyu.edu.hk (B. Niu), qyang@cse.ust.hk (Q. Yang), csckshiu@comp.polyu.edu.hk (S.C.K. Shiu), sankar@isical.ac.in (S.K. Pal).

algorithm is shown to outperform the one-dimensional (1D) Laplacianfaces, the 2D Eigenfaces and Fisherfaces methods.

2. The 2D Laplacianfaces

In this section we first provide a detailed description on the proposed 2D Laplacianfaces algorithm, and show how it differs from the standard Laplacianfaces method We then describe its utilization for feature extraction and classification.

2.1. Idea and algorithm

Let X denote an n-dimensional unitary column vector. A represents an image of m rows and n columns. In the 1D Laplacianfaces method, the sample image, A, has to be transformed to form a vector of $m \times n$ dimensions prior to training. Instead, in the new algorithm, the 2D Laplacianfaces method, we project the image matrix directly onto the vector X:

$$Y = AX. \tag{1}$$

The obtained m-dimensional vector Y is called the projection feature vector, which is the horizontal projection of the image A. Given a set of training images $T = \{A_1, \ldots, A_i, \ldots, A_j, \ldots, A_N\}$ the objective function of the 2D Laplacianfaces method is defined as

$$\operatorname*{Min}_X \sum_{ij} \|Y_i - Y_j\|^2 S_{ij}, \tag{2}$$

where Y_i is the projection feature vector corresponding to the image A_i, $\|\cdot\|$ is the L_2 norm and S_{ij} is the similarity between the image A_i and A_j in the observation space and is defined as

$$S_{ij} = \begin{cases} \exp(-\|A_i - A_j\|^2/t) & \text{if } x_i \text{ is among the } k \\ & \text{nearest neighbors of } x_j\text{, or} \\ & x_j \text{ is among the } k \text{ nearest} \\ & \text{neighbors of } x_i, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where k is the size of the local neighborhood, and t is the window width determining the rate of decay of the similarity function. As shown in Eq. (2) the objective function imposes a heavy penalty if two arbitrary neighboring samples A_i and A_j in the original space are mapped far apart. Minimizing this function ensures that if A_i and A_j are near each other, their projection feature vectors Y_i and Y_j are close to each other as well. Therefore, the locality of the sample space can be maximally preserved when the original data are transformed to the feature space through projections. By taking several algebraic steps, the 2D Laplacianfaces method is formulated to minimize the following objective function

$$\begin{aligned} \operatorname*{Min}_X \sum_{ij} \|Y_i - Y_j\|^2 S_{ij} &= \sum_{ij} \|A_i X - A_j X\|^2 S_{ij} \\ &= \sum_{ij} [X^{\mathrm{T}}(A_i - A_j)^{\mathrm{T}}(A_i - A_j)X] S_{ij} \\ &= X^{\mathrm{T}} \left[\sum_i A_i^{\mathrm{T}} A_i \sum_j S_{ij} - \sum_{ij} A_i^{\mathrm{T}} S_{ij} A_j \right] X \\ &= X^{\mathrm{T}} A^{\mathrm{T}} (D - S) A X \\ &= X^{\mathrm{T}} A^{\mathrm{T}} L A X, \end{aligned} \tag{4}$$

where $A^{\mathrm{T}} = [A_1^{\mathrm{T}}, \ldots, A_N^{\mathrm{T}}]$ and $A = [A_1, \ldots, A_N]^{\mathrm{T}}$ take the mathematical operations as the $1 \times N$ and the $N \times 1$ block matrix, whose row and column consists of the image matrix A_i^{T} and A_i, $i = 1, \ldots, N$, respectively. D is the $N \times N$ block diagonal matrix, whose diagonal element is d_{ii}, $d_{ii} = \sum_j S_{ij}$, which is the sum of the similarity values of all the sample images to the ith image in the original space. S is the similarity matrix, and L is called the Laplacian matrix. Both of these two matrices are of $N \times N$ dimensions. The entry of the matrix D indicates how important each point is. A constraint is imposed as follows:

$$X^{\mathrm{T}} A^{\mathrm{T}} D A X = 1. \tag{5}$$

Hence, the 2D Laplacianfaces method is formulated as

$$\begin{aligned} &\operatorname*{Min}_X \quad X^{\mathrm{T}} A^{\mathrm{T}} L A X, \\ &\text{s.t.} \quad X^{\mathrm{T}} A^{\mathrm{T}} D A X = 1. \end{aligned} \tag{6}$$

In Eq. (6), the matrix D provides a natural measure on the importance of the training samples. In the original data space, the outlier samples have fewer close neighbors than those in the regions of high density of distribution. Some distortion of the local geometry near around these outliers after transformation is unlikely to have the significant impact on the result of classification. Hence, they are less important than those samples that have more close neighbors in determining the optimal directions of projection. In Eq. (6), by using the constraint, we are able to not only remove the arbitrary scaling factor of the projection vectors, but also take into consideration the importance of each sample for optimization [1].

By applying the Lagrange multiplier method, we are able to reduce Eq. (6) to a generalized eigen problem, as shown in Eq. (7)

$$A^{\mathrm{T}} L A X = \lambda A^{\mathrm{T}} D A X, \tag{7}$$

where the matrices $A^{\mathrm{T}} L A$ and $A^{\mathrm{T}} D A$ are both of $N \times N$ dimensions, and L and D are symmetric and positive semidefinite. We can work out the optimal projection vector X by solving this equation. The eigenvectors associated with the first d smallest eigenvalues will be utilized for feature extraction.

2.2. Feature extraction

Let us denote the optimal projection vectors as $X_1, \ldots, X_d$. For a given input image A, let $Y_i = AX_i, i = 1, \ldots, d$. A set of the projection feature vectors, $Y_1, \ldots, Y_d$, can then be obtained. Note that the features extracted in the 2D Laplacianfaces method are vectors, while in the original algorithm they are scalars. The projection vectors are used to form an $m \times d$ matrix $B = [Y_1, \ldots, Y_d]$ called the feature matrix of the sample image A.

2.3. Classification

After obtaining the feature matrix of all the training images, the one nearest neighbor classifier is then used for classification. The distance between any two feature matrices $B_i = [Y_{i1}, \ldots, Y_{id}]$ and $B_j = [Y_{j1}, \ldots, Y_{jd}]$ is defined by

$$d(B_i, B_j) = \sum_{p=1}^{d} \|Y_{ip} - Y_{jp}\|. \tag{8}$$

Suppose that the feature matrices are $B_1, \ldots, B_N$ and each of these samples is assigned a class label C. Given an input testing image B, if $d(B, B_1) = \min \mathrm{d}(B, B_j)$ and B_1 belongs to class C, then B is classified as belonging to C.

3. Experimental results

In this section, we experimentally evaluate the proposed 2D Laplacianfaces method on two well-known face databases, FERET and AR. The FERET database is employed to test the performance of the face recognition algorithms when various numbers of samples are selected for training, while the AR database is used to assess its performance when the face images are taken with the variations of illuminations, facial expressions and time sessions. The experiments are performed on a Pentium 4 2.6 GHz PC with 512 MB RAM memory under Matlab 7.1 platform.

3.1. Results on FERET database

The FERET face image database is a result of the FERET program that is sponsored by the US Department of Defense, through the Defense Advanced Research Products Agency (DARPA) [9]. In our evaluation, we choose a subset of the database that contains 1400 images collected from 200 individuals for examination. Specifically, seven facial images are captured for each subject with varying facial expressions, poses and illumination conditions. In the preprocessing stage, the images are histogram equalized, manually cropped and resized from the size of 80×80 to 40×40, to further reduce the computation and the memory costs of experiments. We perform six tests with various numbers of samples for training. Hence, in the kth test, we select the first k images of each individual for training, and use the others for testing purpose. The top recognition rates achieved in the six tests and the numbers of the projection vectors used for classification are presented in Table 1.

It can be observed that when we choose only one sample from each class for training; the recognition rates of all the six methods are about 70% on average. Of all the methods the proposed 2D Laplacianfaces is consistently better than the rest Also, we note that the Fisherfaces (1D and 2D) fail to construct the within-class scatter matrix for feature extraction, as there is only one sample in each class available for training. When we increase the number of training samples from 1 to 6, the recognition rate gets improved. When we choose six samples for training and leave one sample for testing the recognition rate reaches to its maximum of over 90% averagely. In all the six tests, the proposed 2D Laplacianfaces outperforms the 2D Eigenfaces and the 2D Fisherfaces, significantly and consistently. On the other hand, we also note that all the 2D methods show better performance than the 1D methods in terms of accuracy, which is consistent with the results obtained in [4–8]. In Fig. 1, we show the average recognition rates of the first 40 projection vectors used for classification. For each dimension, the curve depicts the mean average of the recognition rates achieved using the various numbers of samples for training.

In Fig. 1, the 2D Laplacianfaces is consistently more accurate than the 2D Fisherfaces and the 2D Eigenfaces methods. The 1D Laplacianfaces also outperforms the 1D Fisherfaces and the 1D Laplacianfaces. Here, we may note that for the 2D methods, an optimal number of projection vectors have to be carefully chosen in order to achieve the best result of classification. After testing on the first several eigenvectors, we can hardly improve the recognition rate by simply recruiting more projection vectors for classification The reason why there is such performance loss is that for 2D methods, the selected leading eigenvectors

Table 1
Top recognition rate (%) and number of components used

Method	Number of training samples of each class					
	1	2	3	4	5	6
Eigenfaces	69.5 (64)	73.6 (77)	81.8 (78)	87.7 (55)	90.8 (48)	90.8 (72)
Fisherfaces	–	75.3 (40)	83.6 (44)	89.3 (39)	92.2 (55)	92.7 (70)
Laplacianfaces	72.3 (66)	76.1 (45)	84.9 (50)	89.5 (42)	92.9 (60)	93.2 (60)
2D Eigenfaces	71.8 (2)	75.7 (2)	83.7 (3)	88.2 (4)	90.8 (3)	91.2 (3)
2D Fisherfaces	–	77.5 (3)	84.5 (2)	90.6 (4)	92.4 (3)	92.6 (2)
2D Laplacianfaces	73.2 (2)	78.1 (3)	85.2 (3)	91.1 (2)	93.1 (2)	93.4 (2)

(with the largest eigen values for 2D Eigenfaces and 2D Fisherfaces, and the smallest eigen values for 2D Laplacianfaces) are quite effective in explaining most of the discriminative information of the training data; yet the remaining suboptimal eigenvectors are far less informative and incapable of providing further useful information for classification. The employment of these vectors can only bring up more noises that reduce the signal-to-noise ratio, which leads to the slight decreases of the recognition rates.

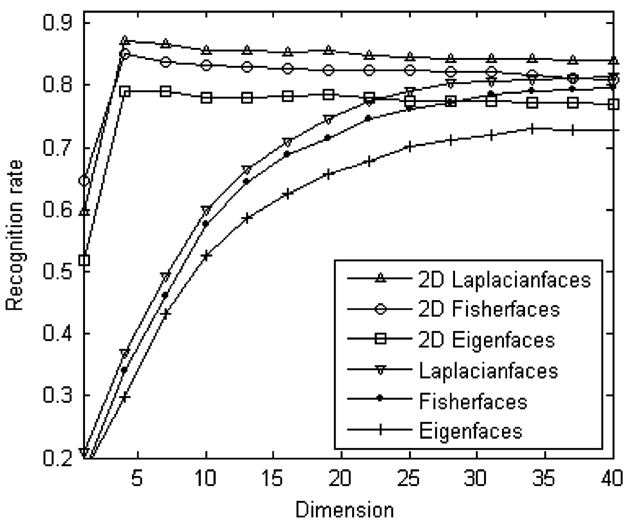

Fig. 1. Average recognition rate with varying dimension of projection vectors.

In Table 2, we compare the computational and the memory space complexities of the six methods. Here m and n is the number of the rows and the columns of the image matrix. L, M and N is the number of the projection vectors, the testing and the training samples, respectively.

Table 2
Time and memory complexities

Method	Complexity		
	Time (training)	Time (testing)	Memory
Eigenfaces/Fisherfaces	$O(m^2n^2L)$	$O(MNL)$	$O(m^2n^2)$
Laplacianfaces	$O(m^2n^2L + mnN^2)$	$O(MNL)$	$O(m^2n^2)$
2D Eigenfaces/2D Fisherfaces	$O(n^2L)$	$O(mMNL)$	$O(n^2)$
2D Laplacianfaces	$O(n^2L + mnN^2)$	$O(mMNL)$	$O(n^2)$

In Table 2, for the Eigenfaces and the Fisherfaces (1D and 2D), since we need to perform $O(MN)$ tests when using the nearest neighbor rule for classification and for each test it has the time complexity of $O(L)$ and $O(mL)$, the testing time is $O(MNL)$ and $O(mMNL)$ for the 1D and the 2D method, respectively. The memory cost is determined by the size of the matrices of the associated eigen equations, which is $O(m^2n^2)$ and $O(n^2)$ for the two types of methods The training time complexity depends on both the size of the matrices in the eigen equations and the number of the projection vectors that are required to be computed. For Eigenfaces and Fisherfaces (1D and 2D), this is $O(m^2n^2L)$ and $O(n^2L)$, respectively. For the Laplacianfaces method, an extra time cost to construct the similarity matrix, i.e., $O(mnN^2)$, will be taken into account. Specifically, for the 1D and the 2D Laplacianfaces, we present and compare in Table 3 the CPU time for training and testing, and the size of the matrices of the eigen equations.

In Table 3, while the 1D Laplacianfaces method takes averagely 977.22 s for training, our proposed 2D Laplacianfaces uses only 1.59 s Moreover, the size of the matrix is reduced from 1600×1600 to 40×40, which significantly improves the memory efficiency of the algorithm. We may also note that the testing time of the 2D Laplacianfaces is 7.72, slightly

Table 3
Time and memory space used for training and testing

Method	Average time (s) and memory cost				
	Time (training)	Time (testing)	Time (testing KDT)	Time (total)	Size of matrix
Laplacianfaces	977.22	4.86	0.14	977.36	1600×1600
2D Laplacianfaces	1.59	7.72	0.18	1.77	40×40

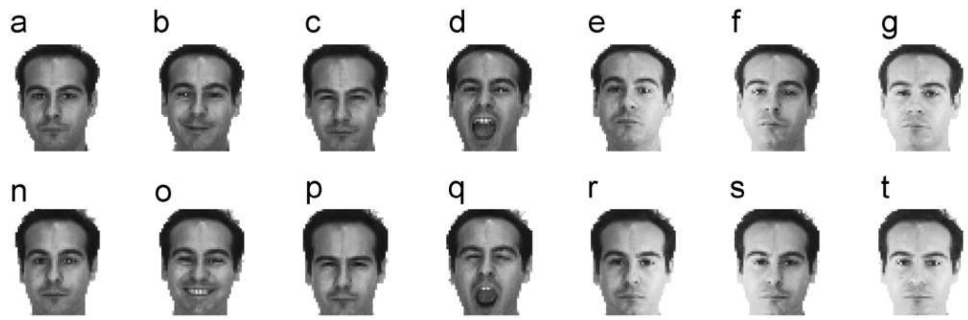

Fig. 2. Sample images for one subject of the AR database.

higher than that of the 1D Laplacianfaces, 4.86 s. To improve the testing efficiency of the algorithm, we can exploit the *k*-dimensional tree (KDT) method [12] to accelerate the searching process of the nearest neighbor classification. The KDT method is a popular decision tree algorithm. It can recursively partition the sample space into a number of the smaller subsets for efficient pattern indexing and query. Given some input pattern for matching, it transverses the tree structure while making simple test at each branching node to discard a large portion of the data, so as to speed up the searching process. The query time complexity of the KDT algorithm is at worst $O(MN)$ and at best $O(M \log N)$, which is much lower than that of the simple *kNN* retrieval. In our experiment, by taking advantage of the KDT algorithm [10], the testing time of the two methods is reduced significantly from 4.86 to 0.14, and 7.72 to 0.18 s, respectively, which makes 2D Laplacianfaces a practical choice for real world applications

3.2. Results on AR database

The AR face database [11] consists of over 4000 face images of 126 individuals taken in two time sessions under the variations of illuminations, facial expressions and occlusion conditions. Each person has 26 images. In our experiment we consider using a subset of 14 images of each person for training and testing. Fig. 2 shows the selected sample images of one subject.

In Fig. 2, the images (a)–(g) and (n)–(t) are drawn from the first and the second time sessions, respectively. For each session the first four images (a)–(d) and (n)–(q) involve the variation of facial expressions (neutral, smile, anger, scream) while the images (e)–(g) and (r)–(t) are taken under different lighting conditions (left light on, right light on, all sides light on). The images are manually cropped and scaled down to 50×40 pixel to reduce the computation and the memory costs of the experiment in the preprocessing stage. We design and perform three experiments to examine the performance of 2D

Table 4
Indices of training and testing images

Data set	Experiment conditions		
	Illumination	Expression	Time
Training set	$\{e, s\}$	$\{a, n\}$	$\{a, b, c, d, e, f, g\}$
Testing set	$\{f, g, r, t\}$	$\{b, c, d, o, p, q\}$	$\{n, o, p, q, r, s, t\}$

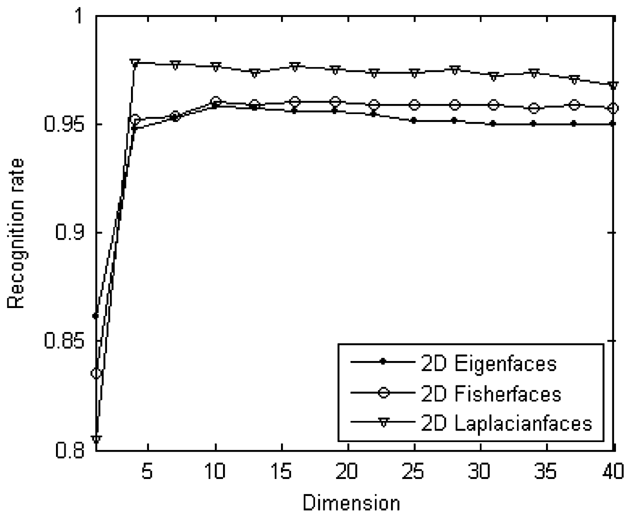

Fig. 3. Recognition rate over dimensions of feature vectors (expressions).

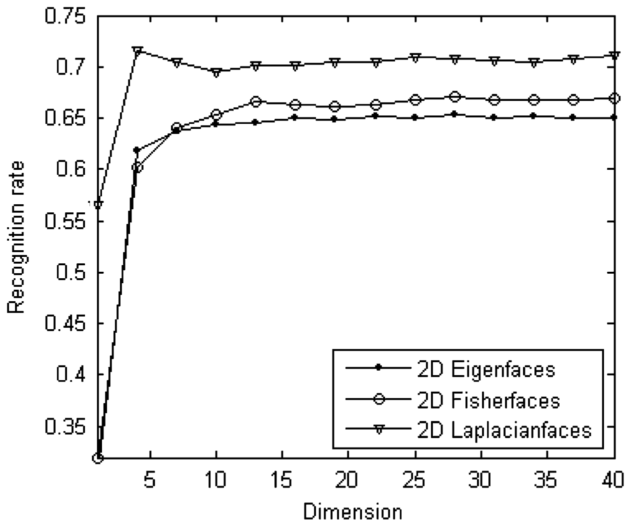

Fig. 4. Recognition rate over dimensions of feature vectors (time).

Table 5
Performance of three algorithms using image based projection technique

Experiment		Top recognition rate (%)	Dimension	Classification time (s)
Expression	2D Eigenfaces	95.4	10	5.547
	2D Fisherfaces	95.6	10	5.281
	2D Laplacianfaces	97.8	4	4.765
Time	2D Eigenfaces	65.2	22	42.42
	2D Fisherfaces	68.6	14	28.75
	2D Laplacianfaces	71.5	4	17.66
Illumination	2D Eigenfaces	80.2	27	12.375
	2D Fisherfaces	91.4	9	3.765
	2D Laplacianfaces	93.7	3	1.975

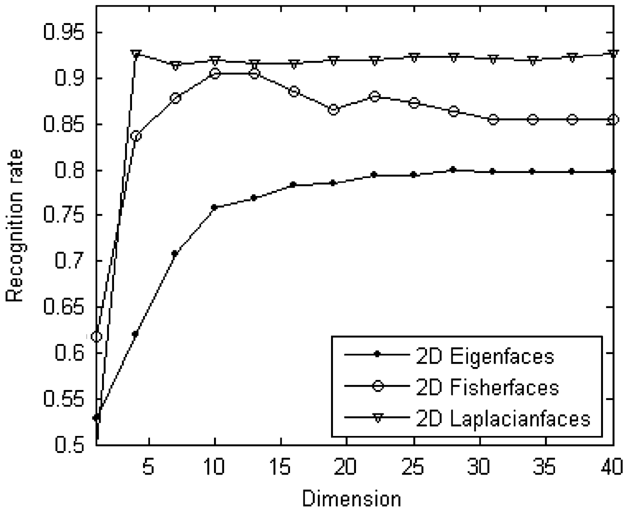

Fig. 5. Recognition rate over dimensions of feature vectors (illumination).

Eigenfaces, 2D Fisherfaces and 2D Laplacianfaces under the variations of facial expressions, time sessions and illumination conditions. The indices of the images of each person used in the three tests are listed in Table 4.

Table 5 shows the top recognition rate, the number of the dimensions of feature vectors used for classification and the testing time of the three algorithms.

It can be seen that the proposed 2D Laplacianfaces method outperforms the 2D Fisherfaces and the 2D Eigenfaces methods in all the three tests. It improves the recognition rate by 2.4, 6.3, 3.5% over the 2D Eigenfaces, and 2.2, 2.9, 2.3% over the 2D Fisherfaces, respectively. It requires fewer dimensions of projection vectors and time to achieve the top recognition rate as shown in column 5 of Table 5. Further, in Figs. 3–5 we also show the relationship between the accuracy rate of the three algorithms and the dimension of the feature vectors used for recognition.

In these figures we can observe that the 2D Laplacianfaces method can explain most of the effective discriminative information with only a small number of projection vectors, as opposed to the other two methods where more features have to be provided to achieve the top recognition rate. The 2D Laplacianfaces is also quite stable and consistent in outperforming the 2D Eigenfaces and the 2D Fisherfaces methods with various number of feature vectors, as indicated in the figures.

4. Conclusion

In this paper, we developed the two-dimensional (2D) Laplacianfaces method and applied it to the face recognition problem. The proposed method has the following three properties: First, it can maximally preserve the locality of the geometric structure of the sample space to extract the most salient features for classification. The learned local patterns of the training data are suitable for the neighborhood based *kNN* queries in the projected low-dimensional feature space. Experimental results on the two well-known face image databases, FERET and AR, indicate that the proposed 2D Laplacianfaces is more accurate than the 2D Eigenfaces and the 2D Fisherfaces that rely on the global information of the data space for analysis. Second, by taking advantage of the image based projection technique, 2D Laplacianfaces is computationally more efficient than the one-dimensional (1D) Laplacianfaces for training. Both the training time and the memory efficiency of the algorithm are improved significantly. The recognition accuracy of the 2D Laplacianfaces is also better than that of the 1D Laplacianfaces as the size of the matrix is small, enabling the full optimization of the objective function. Third, the utilization of the KDT algorithm is quite effective in speeding up the *kNN* query process. By adopting the KDT method, the 2D Laplacianfaces is improved to be not only more efficient for training, but also as competitively fast as other methods for query and classification.

Finally, it should be pointed out that the application of our proposed 2D Laplacianfaces method is not limited to the face recognition problem. It can also be potentially utilized to address many other types of problems in pattern recognition, such as palm and finger print recognition, gesture recognition, audio and video clustering, gene microarray analysis, financial time-series predictions, web document classification, etc., where the analysis of the high dimensional data is required.

Acknowledgments

The work described in this paper was fully supported by the grant from Hong Kong Polytechnic University (Grant No. A-PD55) and NECLC 05/06 EG01.

References

[1] X. He, S. Yan, Y. Hu, P. Niyogi, H.-J. Zhang, Face Recognition Using Laplacianfaces, IEEE Trans. Pattern Anal. Mach. Intell. 27 (3) (2005) 328–340.
[2] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[3] K. Liu, Y.Q. Cheng, J.Y. Yang, Algebraic feature extraction for image recognition based on an optimal discrimination criterion, Pattern Recognition 26 (6) (1993) 903–911.
[4] J. Yang, D. Zhang, A.F. Frangi, J.Y. Yang, Two dimensional PCA: a new approach to appearance based face representation and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 26 (1) (2004).
[5] J. Yang, J.Y. Yang, From image vector to matrix: a straightforward image projection technique—IMPCA vs. PCA, Pattern Recognition 35 (9) (2002) 1997–1999.
[6] J. Yang, D. Zhang, Y. Xu, J.Y. Yang, Two dimensional discriminant transform for face recognition, Pattern Recognition 38 (7) (2005) 1125–1129.
[7] H. Xiong, M.N.S. Swamy, M.O. Ahmad, Two dimensional FLD for face recognition, Pattern Recognition 38 (7) (2005) 1121–1124.
[8] X. Jing, H.S. Wong, D. Zhang, Face recognition based on 2D Fisherface approach, Pattern Recognition 39 (4) (2006) 707–710.
[9] The Facial Recognition Technology (FERET) Database, ⟨http://www.itl.nist.gov/iad/humanid/feret/feret_master.html⟩, 2002.
[10] Matlab Central File Exchange, ⟨http://www.mathworks.com/matlabcentral/⟩.
[11] A.M. Martinez, R. Benavente, The AR Face Database, CVC Technical Report 24, June 1998.
[12] S. Arya, D.M. Mount, N.S. Netanyahu, R. Silverman, A.Y. Wu, An optimal algorithm for approximate nearest neighbor searching, J. ACM 45 (1998) 891–923.

About the Author—BEN NIU received his Bachelor's degree in 1999 and the Master's degree in 2002, both in mathematics, from Hebei University, PR China. He began to pursue his PhD degree in the Department of Computing, Hong Kong Polytechnic University, in 2003. His current research interests include statistical pattern recognition, computer vision, machine learning and case-based reasoning.

About the Author—QIANG YANG received the PhD degree from the University of Maryland, College Park. He is a faculty member in the Hong Kong University of Science and Technology's Department of Computer Science and Engineering. His research interests are AI planning, machine learning, case-based reasoning and data mining. He is a senior member of the IEEE and an associate editor for the IEEE Transactions on Knowledge and Data Engineering and IEEE Intelligent Systems.

About the Author—SIMON C.K. SHIU received the MSc degree in computing science from the University of Newcastle Upon Tyne, U.K., in 1985, the MSc degree in business systems analysis and design from City University, London, in 1986 and the PhD degree in computing in 1997 from Hong Kong Polytechnic University. He is an assistant professor in the Department of Computing at Hong Kong Polytechnic University, Hong Kong. He worked as a system analyst and project manager between 1985 and 1990 in several business organizations in Hong Kong. His current research interests include pattern recognition, machine learning and case-based reasoning. He has coguest edited a special issue on soft case-based reasoning in the journal Applied Intelligence. He is a member of the British Computer Society and the IEEE.

About the Author—SANKAR K. PAL is the Director and a Distinguished Scientist of the Indian Statistical Institute. He founded the Machine Intelligence Unit and the Center for Soft Computing Research: A National Facility in the Institute in Kalkata. He received a PhD in Radio Physics and Electronics from the University of Calcutta in 1979, and another PhD in Electrical Engineering along with DIC from Imperial College, University of London, in 1982. He worked at the University of California, Berkeley, and the University of Maryland, College Park, in 1986–1987; the NASA Johnson Space Center, Houston, Texas, in 1990–1992 and 1994; and in US Naval Research Laboratory, Washington, DC, in 2004. Since 1997 he has been serving as a distinguished visitor of IEEE Computer Society (USA) for the Asia-Pacific Region, and held several visiting positions in Hong Kong and Australian universities. Prof. Pal is a Fellow of the IEEE, USA, The Academy of Sciences for the Developing World, Italy, International Association for Pattern recognition, USA, and all the four National Academies for Science/Engineering in India. He is a co-author of 13 books and about 300 research publications in the areas of pattern recognition and machine learning, image processing, data mining and web intelligence, soft computing, neural nets, genetic algorithms, fuzzy sets, rough sets and bioinformatics. He has received the 1990 S.S. Bhatnagar Prize (which is the most coveted award for a scientist in India), and many prestigious awards in India and abroad including the 1999 G.D. Birla Award, 1998 Om Bhasin Award, 1993 Jawaharlal Nehru Fellowship, 2000 Khwarizmi International Award from the Islamic Republic of Iran, 2000–2001 FICCI Award, 1993 Vikram Sarabhai Research Award, 1993 NASA Tech Brief Award (USA), 1994 IEEE Trans. Neural Networks Outstanding Paper Award (USA), 1995 NASA Patent Application Award (USA), 1997 IETE-R.L. Wadhwa Gold Medal, the 2001 INSA-S.H. Zaheer Medal, and 2005-06 P.C. Mahalanobis Birth Centenary Award (Gold Medal) for Lifetime Achievement. Prof. Pal is an associate editor of IEEE Transactions of Pattern Analysis and Machine Intelligence, IEEE Transactions on Neural Networks, Pattern Recognition Letters, Neurocomputing (1995–2005), Applied Intelligence, Information Sciences, Fuzzy Sets and Systems, Fundamenta Informaticae, International Journal of Computational Intelligence and Applications, and Proceedings of INSA-A; a member, Executive Advisory Editorial Board, IEEE Transactions on Fuzzy Systems, International Journal on Image and Graphics, and International Journal of Approximate Reasoning; and a guest editor of IEEE Computer.