Face Recognition Using Improved Fast PCA Algorithm

Neerja
Rayat & Bahra Inst. of Engg. & Bio-Tech, Kharar, Mohali (Punjab), INDIA. PIN: 140104, neerjamittal_2k1@yahoo.com Fax: +91 160 5009680

Ekta Walia
Computer Science Department, Punjabi University, Patiala (Punjab), INDIA. PIN: 147002, wekta@yahoo.com

Abstract

The Principal Component Analysis (PCA) is one of the most successful techniques that have been used to recognize faces in images. However, high computational cost and dimensionality is a major problem of this technique. There is evidence that PCA can outperform over many other techniques when the size of the database is small. In this paper, a fast PCA based Face Recognition Algorithm is proposed. In the proposed algorithm the database is sub grouped using some features of interest in faces. Only one of the obtained subgroups is provided to PCA for recognition. The performance of the proposed algorithm is tested on Indian face database, and the obtained results show an improvement in performance of the proposed algorithm as compared to the same with PCA method.

Keywords: Face Recognition, Fuzzy Feature Extraction, Eigenface, Principle Component Analysis.

1. Introduction

Face recognition is an unsolved problem under the conditions of pose, illumination, database size etc., still attracts significant research efforts. The main reasons of ongoing research are its many real world applications like human/computer interface, surveillance, authentication, perceptual user interfaces and lack of robust features and classification schemes for face recognition task. Principal Component Analysis (PCA) is a typical and successful face based technique. Turk and Pentland developed a face recognition system using PCA in 1991 [1] [2]. Belhumeur et. al proposed Fisherface technique based on Linear Discriminant Analysis (LDA) in 1997 [3]. Recently Facial feature extraction [4] [5] has become an important topic in automatic recognition of human faces. Extracting the basic features like eyes, nose and mouth exactly, is necessary for most of the feature based approaches [6]. Other Face Recognition approaches based on Discriminant Analysis and Feature Extraction also exists but the achievements in the field of automatic face recognition by computer is not as satisfactory as in other areas like fingerprints and pattern recognition etc.

Despite of the good results of PCA, this technique has the disadvantage of being computationally expensive and complex with the increase in database size, since all pixels in the image are necessary to obtain the representation used to match the input image with all others in the database. Here we present a new PCA based face recognition approach, using the geometry and symmetry of faces, which extract the features using fast Fuzzy Edge Detection [7] to locate the vital feature points on eyes, nose and mouth exactly and quickly. On the basis of these features, subgroups of database images are formed. During recognition only the images falling in same group as test image, will be loaded as image vectors in covariance matrix of PCA for comparison. This method can also improve the accuracy of face recognition.

The rest of the paper is organized as follows. In Section 2 we describe well known PCA method in detail. Section 3 explains the fast Fuzzy Edge Detection method for feature extraction. Section 4 explains the proposed method. Experimental results are presented and discussed in Section 5. Finally, in Section 6 conclusion and future recommendations are presented.

2. PCA Algorithm

This procedure consists of two phases: 1) training phase and 2) recognition phase. In training phase, the

978-0-7695-3119-9/08 $25.00 © 2008 IEEE
DOI 10.1109/CISP.2008.144

samples of data, on which the system needs to recognize, are used to create an Eigen Matrix which transforms the samples in the image space into the points in Eigenspace. The image samples are taken as grayscale images and are transformed from 2D matrix to 1D column vector of size $N^2 \times 1$ by placing the image matrix columns consecutively. These column vectors of *n* images are placed column wise to form the data matrix (image set) *X* of dimension $N^2 \times n$.

And *m* be the mean vector of data vectors in matrix *X* given by

$$m = \frac{1}{n}\sum_{i=1}^{n} x^i \tag{1}$$

The vector of data matrix *X* are centered by subtracting the mean vector *m* from all the column vectors of *X* to get the covariance matrix *Ω* of the column vectors and is given by

$$\Omega = \overline{XX^t} \tag{2}$$

The eigenvalues and corresponding eigenvectors are computed for the covariance matrix

$$\Omega V = \Lambda V \tag{3}$$

Where *V* is the set of eigenvectors associated with the eigenvalues Λ . Set the order of the eigenvectors $vi \in V$ according to their corresponding eigenvalues from high to low. This matrix of eigenvectors is eigenspace *V*. The data matrix *X* is projected onto the eigenspace to get P consisting n columns, where

$$P = V^t X \tag{4}$$

In the Recognition phase, the image *I* to be recognized is converted to 1D vector and form *J* as mentioned above which is projected on to same eigenspace to get *Z*.

$$Z = X^t J \tag{5}$$

And the Euclidean distance d between Z and all the projected samples in P is measured using L_2 norm (Euclidean distance) of an image A and image B is

$$L_2(A,B) = \sum_{i=1}^{N}(A_i - B_i)^2 \tag{6}$$

The projected test image is compared to every projected training image and the training image that is found to be closest to the test image is used to identify the training image.

3. Fuzzy Feature Extraction

The simplest way for defining a fuzzy edge detector is the determination of a proper membership function $\hat{\mu}_{mn}$ for each pixel g_{mn} at the position (m,n) with a surrounding *w*×*w* window. Let an $M \times N$ image *A* be the set of all pixels $g_{mn} \in [0, L]$, then A can be regarded as an array of fuzzy singletons $\mu_{mn} \in [0,1]$, indicating the degree of brightness of each gray level g_{mn}

$$A = \bigcup_{m=1}^{M}\bigcup_{n=1}^{N}\frac{\mu_{mn}}{g_{mn}} \tag{7}$$

The membership function μ_{mn} can be achieved, among other possibilities, by a simple normalization

$$\mu_{mn} = \frac{g_{mn}}{\max\limits_{i\in[1,M],\, j\in[1,N]} g_{ij}} \tag{8}$$

The image $A^{'}$ containing all edges is:

$$A' = \bigcup_{m=1}^{M}\bigcup_{n=1}^{N}\frac{\hat{\mu}_{mn}}{g_{mn}} \tag{9}$$

Where $\hat{\mu}_{mn}$ indicates the degree of edginess for each pixel. The task of edge detection is therefore, the determination of the membership function $\hat{\mu}_{mn}$ for each pixel.

$$\hat{\mu}_{mn} = \min(1, \frac{2}{w}\sum_{i}\sum_{j} T(\mu_{ij}, 1-\mu_{ij})) \tag{10}$$

65

Figure 1: Original image (2.tif)

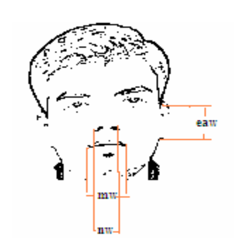

Figure 2: Fuzzy Feature Extraction

4. Proposed algorithm

4.1 Preprocessing

The proposed method incorporates a preprocessing step in training and recognition phase. For this well known Histogram Equalization is applied which results in contrast adjusted images by increasing the local contrast. This process re-assigns the gray values of pixels in input images such that the output images contain a uniform distribution of gray values. And images are normalized to reduce the variations in gray level values and light asymmetry effect using:

$$I_{new} = \frac{(I_{old} - m) \times ustd}{std} + um \qquad (11)$$

Where *std* and *m* are the standard deviation and mean of input image, *ustd* and *um* are the desired standard deviation and mean.

4.2 Procedure

The proposed algorithm involves the following operations:

Step1: Apply Histogram equalization and normalization to preprocess the images (training) in database.

Step2: Apply segmentation using fast Fuzzy Edge Detection (as described in section 3) to extract the features from images and store the desired values in feature vector.

Step3: Classify the training images into subgroups for the features in feature vector and store the group value in one of the columns of feature vector.

Step4: Get input test image.

Step5: Apply feature extraction and group classification on test image as in step number 2 and 3.

Step6: Training images in same group as of test image are loaded in the image set.

Step7: Apply PCA to recognize the test image on obtained image set.

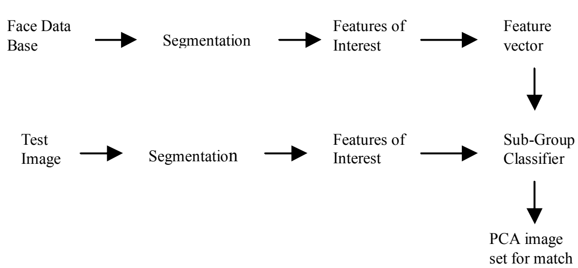

Figure3: Pictorial representation of the operations followed in the proposed method.

5. Experiments and Results

This section presents the results of the experiments conducted to compare the performance of PCA and proposed method. The methods are implemented in MATLAB on grayscale images of 32 different persons of size 128×128 in tiff format, as shown in Figure 2. Images are taken from Indian Face Database [9].

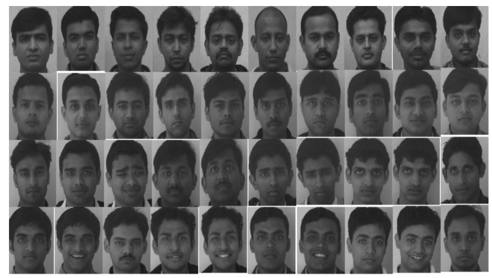

Figure 4: Training images

We performed a set of experiments to demonstrate the efficiency of the proposed method. Using the Indian Face Database, 40 different images of 32 persons are used in training set. Figure 2 shows a sample binary image detected using Fuzzy edge detection method described in section 3. Local Features extracted with Fuzzy method are sharp and clear as compared with other edge detection approaches and it is fast and computationally inexpensive than other edge detection methods. The performance of face recognition algorithms is compared on various parameters, among which time and dimensionality are most considerable. A face contains a number of features of interest like nose, mouth, eyes, ears, eyebrows etc; and each of these features has some values or weights. In our implementation we have used the nose width and mouth width. In the proposed algorithm, the faces are sub-categorized based on the weights of these features in the training database. Any face to be recognized, is compared with only a sub-group, in which it falls.

Comparison results of both the methods are presented in Figure 5 and Figure 6. Computational burden is an important factor in practical application, where the amount of required memory and speed of the processor have direct bearing on final cost. From Figure 5, we can see that PCA is computationally expensive to calculate than the Proposed method i.e. time is increasing as the number of images in training set is increasing. Here time is the total time spent on training database and recognition of test image. In Figure 6, we can again see that for PCA, dimensionality is same as of the number of images in the training set while in the proposed algorithm, only a few images in same subgroup are multiplied with image size (dimensionality) and others are ignored.

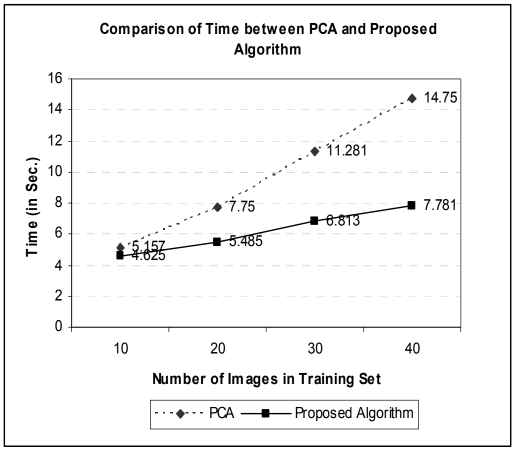

Figure 5: Comparison of the algorithms on time for a sample test image (2.tif)

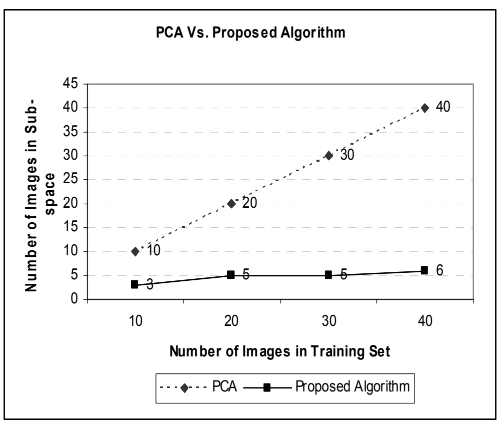

Figure 6: Comparison of Algorithms on dimensionality of Image set (Eigen space) for a sample test image (2.tif)

The results presented in Table1 show the better performance of proposed algorithm than PCA in terms of time and dimensionality for a given set of test images.

6. Conclusion

From the experimental results, it is apparent that our efforts to combine feature based group classification to PCA, significantly improve the performance of traditional method. The proposed method is simple in implementation and fast in computing. So far we have tested this approach on a relatively medium sized and clean database only. In future work, the performance of the face feature selection will be further examined and we intend to gather more data in order to obtain better generalization ability, to include expression and lighting changes.

7. References

[1] M. Turk and A. Pentland, "Eigenfaces for Recognition", *J. Cognitive Neuroscience*, vol. 3, pp. 71-86, 1991.

[2] M. Turk, W Zhao, R chellapa, " Eigenfaces and Beyond", *Academic Press*, 2005.

[3] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection", *IEEE trans. Pattern Analysis and Machine Intelligence*, vol. 19, pp. 711-720,1997.

[4] A. Saxena, A. Anand and Prof. A. Mukerjee, "Robust Facial Expression Recognition Using Spatially Localized Geometric Model", *International Conf. on Systemics, Cybernetics and Informatics*, Feb. 12-15, 2004, pp. 124-129.

[5] Ming-Hsuan Yang, David J. Kriegman, Narendra Ahuja, "Detecting Faces in Images: A survey", *IEEE transaction on pattern analysis and machine intelligence*, vol. 24, No. 1, pp. 24-58, 2002.

[6] P. Jonathan Phillips, Hyeonjoon Moon, Syed A. Rizvi, Patric J. Rauss, "The FERET evaluation methodology for face recognition algorithms," *IEEE trans. on pattern recognition and machine Intelligence*, pp. 1090-1104, 2000.

[7] Hamid R. Tizhoosh, "Fast Fuzzy Edge Detection", *IEEE transactions*, 2002, pp.239-242.

[8] Wang Guoqiang and Ou Zongying, "Face Recognition Based on Image Enhancement and Gabor Features", *Proceedings of 6th World Congress on Intelligent Control and Automation*, June 21-23, 2006 IEEE, pp.9761-9764.

[9] Indian Face Database from http://www.face-rec.org/databases/

Table 1: Comparison of results

Test image	No. of images	PCA		Proposed Method	
		Time (in sec.)	Dimensionality (Image set)	Time (in sec.)	Dimensionality (Image set)
1.tif	10	5.8910	16384×10	4.8910	16384×3
	20	7.7350	16384×20	5.4370	16384×4
	30	11.3750	16384×30	6.3440	16384×5
	40	15.0310	16384×40	7.3130	16384×6
2.tif	10	5.1570	16384×10	4.6250	16384×3
	20	7.7500	16384×20	5.4850	16384×5
	30	11.2810	16384×30	6.8130	16384×5
	40	14.7500	16384×40	7.7810	16384×6
3.tif	10	5.2040	16384×10	4.4840	16384×3
	20	7.6720	16384×20	5.2970	16384×4
	30	11.3910	16384×30	6.2650	16384×5
	40	14.9220	16384×40	7.2350	16384×6
5.tif	10	5.7030	16384×10	4.9840	16384×1
	20	8.1710	16384×20	5.4840	16384×1
	30	11.3120	16384×30	5.7810	16384×1
	40	15.0930	16384×40	6.8750	16384×1
8.tif	10	5.5320	16384×10	4.2840	16384×2
	20	8.1090	16384×20	5.7500	16384×5
	30	11.5780	16384×30	6.5940	16384×6
	40	15.0150	16384×40	8.3440	16384×9