Journal of Indian Research (ISSN: 2321-4155)
Vol.2, No.4, October-December, 2014, 63-73

Electronics

USER AUTHENTICATION WITH FUZZY FUSION OF FACE TECHNIQUES

Shiv Ratan Singh*
Jai Prakash**

ABSTRACT

The vast majority of the successful commercial biometric systems at present depend on fingerprint or face-recognition. Moreover, these biometric indicators complement one another in their strengths and advantages. While fingerprint gives exceptionally high verification precision, these still carry some verification errors. The Face recognition is the second most preferred method with reasonably good accuracy. In this paper, we attempt to integrate the face and fingerprint recognition techniques using fuzzy fusion method. The results presented in the paper are very promising and provide a direction for the future research in the user identification.

Keywords: Biometric, face recognition, Fuzzy Logic, fuzzy fusion, Linear Discriminant Analysis(LDA), Principal Component Analysis(PCA).

INTRODUCTION

Face recognition technology has become one of the most important biometric technologies, for its non-intrusive nature and its potential applications like personal identification, security access control, surveillance systems, tele-communications, digital libraries, human-computer interaction, military and so on. In past, much research is done for the enhancement of the accuracy of the Face recognition system. In this exercise, the algorithms have become very computationally complex and such algorithms can't be used in real time systems. In this paper, we have shown that using the basic algorithms for the face recognition and then Fuzzy fusion of these methods significantly improves the results. In this work, Face identification techniques: Principal Component Analysis (PCA and linear discriminant analysis (LDA) [4] are discussed and finally fuzzy fusion of these two techniques is done to further improve the results.

FACE RECOGNITION

It is quite easy to obtain facial images with a couple of inexpensive fixed cameras. Good face recognition algorithms and appropriate preprocessing of the images can compensate for noise and slight variations in orientation, scale and illumination [3].

Face recognition is used for two primary purposes:

1. *Verification* (one-to-one matching): When presented with a face image of an unknown individual along with a claim of identity, making sure whether the individual is who he/she claims to be.

2. *Identification* (one-to-many matching): Given an image of an unknown individual, determining the identity

***Shiv Ratan Singh** is a Lecturer at the Department of ECE Guru Nanak Dev Co-ed Polytechnic, Government of NCT of Delhi. He is also a Research Scholar at Mewar University, Gangrar, Chittorgarh, Rajasthan. Email: sratanphd@gmail.com.
****Jai Prakash** is a Visiting faculty at Mewar University Gangrar, Chittorgarh, Rajasthan.

of that person by comparing (possibly after encoding) that image with a database of (possibly encoded) images of known individuals [3].

The flow diagram for the face recognition techniques is shown in the Figure 1 given below:

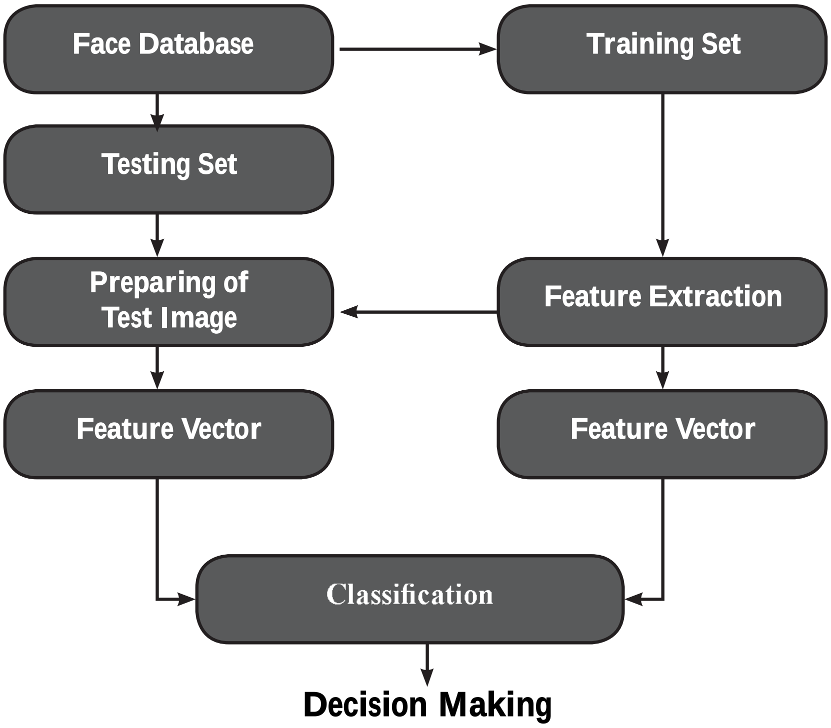

Figure 1: Flow diagram for the face recognition

In the face recognition system the flow must be followed. It defines all the required steps in the face recognition system. Figure 1 defines all the required steps, but the most important step is the Feature Extraction which is ultimately used for dimensional reduction as well as for extracting features from input of the system. Extracted features are passed to the last phase that is classification where the identification or verification rate is calculated.

PRINCIPAL COMPONENT ANALYSIS

A 2-dimension facial image under investigation can be represented as 1- dimension vector by concatenating each row (or column) into a long thin vector [4, 5]. Considering *M vectors* of size *N* (= rows of image × columns of image) a representation, a set of sampled images an be created.

$$x_i = [p_1, p_2 \ldots p_N]^T \quad i = 1 \ldots\ldots M \qquad \textbf{(1)}$$

Where, p_j represents the pixel values.

Let $m = \frac{1}{M}\sum_{i=1}^{M} x_i$ represent the mean image, and let w_i be defined as mean centered image (subtraction of the mean image from each image vector) $w_i = x_i - m$ **(2)**

Our goal is to find a set of e_i 's which have the largest possible projection onto each of the w_i 's. We wish to find a set of M orthonormal vectors e_i for which the quantity $\lambda_i = \frac{1}{M}\sum_{n=1}^{M}\left(e_i^T w_n\right)^2$ is maximized with the orthonormality constraint

$$e_l^T e_k = \delta_{lk} \quad \textbf{(3)}$$

It has been shown that the e_i 's and λ_i 's are given by the eigenvectors and eigenvalues of the covariance matrix $C = WW^T$, where W is a matrix composed of the column vectors w_i placed side by side [4]. The size of C is $N \times N$ which could be enormous. For example, images of size 64×64 create the covariance matrix of size 4096×4096. It is not practical to solve for the eigenvectors of C directly. According to a common theorem in linear algebra, the vectors e_i and scalars¸ λ_i can be obtained by solving for the eigenvectors and eigenvalues of the $M \times M$ matrix $W^T W$. Let d_i and μ_i be the eigenvectors and eigenvalues of $W^T W$ respectively.

$$W^T W d_i = \mu_i d_i \quad \textbf{(4)}$$

By multiplying left to both sides by W

$$WW^T W d_i = W \mu_i d_i \quad \textbf{(5)}$$

which means that the first $M-1$ eigenvectors e_i and eigenvalues λ_i of WW^T are given by Wd_i and μ_i respectively. Wd_i needs to be normalized in order to be equal to e_i . Since we only sum up a finite number of image vectors, M , the rank of the covariance matrix cannot exceed $M-1$ (The -1 come from the subtraction of the mean vector m).

The eigenvectors corresponding to nonzero eigenvalues of the covariance matrix produce an orthonormal basis for the subspace within which most image data can be represented with a small amount of error. The sorting of eigenvectors is done according to their corresponding eigenvalues from high to low. The eigenvector corresponding to the largest eigenvalue has the greatest variance in the image [5]. Similarly, the smallest eigenvalue corresponds to the least variance. It is noticeable that the reduced dimensions are first five to ten percent of the total dimensions. A facial image can be projected onto $M' \ll M$ dimensions by computing

$$\Omega = [v_1, v_2 \ldots v_M]^T \quad \textbf{(6)}$$

where $v_i = e_i^T w_i$. v_i is the i^{th} coordinate in the new space, which is the principal component. The vectors e_i are also for images, hence, known as *eigenimages*, or *eigenfaces* and was first named by [3-5]. The simplest method for determining which face class provides the best description of an input facial image is to find the face class k that minimizes the Euclidean distance

$$\varepsilon_k = \left\|\Omega - \Omega_k\right\| \quad \textbf{(7)}$$

where, Ω_k is a vector describing the k^{th} face class. If ε_k is less than some predefined threshold, a face belongs to the class k.

Limitations Of PCA

The main limitations of the PCA are as follows:

1. The face image should be normalized and frontal-view

2. The system is an auto-associative memory system. It is harmful to be over-fitted.

3. Training is very computationally intensive.

4. It is hard to decide suitable thresholds - It is a kind of Art!

5. The suggested methods to deal with unknown faces and non-faces are not good enough to differentiate them from known faces.

Simulation Results

Face images for the test are taken from AT&T data base. The database has 400 images. We have selected 12 images, for the demonstration of the algorithm. All the files are in PGM format. Each image is displayed by 92×112 pixels, with 256 grey levels per pixel. The images are arranged in 12 directories (one for each 'subject'), which have names of the form sX, where X indicates the subject number (between 1 and 25).

Case

In the first case, 12 images are taken as training set, each with mean 100 and standard deviation of 80. In the second step, the mean and standard deviation of all images are changed for normalization. This is done to reduce the error due to lighting conditions and background

Figure 2: Training Set (AT&T)

The normalized images are shown in Figure 3, and these images are very much similar to the images in Figure 2. However when background changes abruptly, the normalization is very effective.

Figure 3: Normalized Training Set (AT&T)

In the next step, the mean image is generated as shown in Figure 4. The pixel values of the images ranges form 0 to 255.

Figure 4: Mean image

In the next step, co-variance matrix is created, thereafter the Eigen-values are obtained, and the Eigen values close to zero are dropped and for the left over Eigen values, Eigen vector are obtained. Finally, after the normalization of Eigen vectors, Eigen faces are calculated (Figure 5).

Figure 5: Eigenfaces

In case of user authentication, template matching is done. In Figure 6, the input facial image and the re-constructed facial image is shown. The re-constructed image is very much similar to the input image.

(a) **(b)**

Fig. 6 (a) Input and (b) Re-constrcuted images

The major significant difference, between the input and reconstructed image can be seen at the forehead portion of the two images.

LINEAR DISCRIMINANT ANALYSIS

LDA is a powerful face recognition technique that overcomes the limitation of Principle Component Analysis technique. The LDA maximizes the ratio of the determinant of the between-class scatter matrix to the determinant of the within- class scatter matrix of the projected samples. Linear discriminant group images of the same class and separates images of different classes of the images [3].

Considering a *C*-class problem with each class *i* consisting of a set of N_i, *d*-dimensional samples $\{x_1^i, x_2^i \ldots x_{N_i}^i\}$, where the superscript $(.)^i$ represents the class label. Defining the total number of samples as $N = \sum_{i=1}^{C} N_i$ and the probability of occurrence of class '*i*' as $p_i = \frac{N_i}{N}$, the sample mean for class '*i*' as

$\mu^i = \frac{1}{N_i} \sum_{j=1}^{N_i} x_j^i$ and the grand sample mean as μ

$$\mu = \frac{1}{N} \sum_{i=j}^{C} \sum_{j=1}^{N_i} x_j^i = \sum_{i=j}^{C} P^i \mu^i \qquad \textbf{(8)}$$

The within and between class scatter matrices represented as $\sum_W$ and $\sum_B$, respectively, and computed as:

$$\sum\nolimits_W = \sum_{i=1}^{C} P^i \sum\nolimits_W^i = \frac{1}{N} \sum_{i=1}^{C} \sum_{j=1}^{N_i} \left(x_j^i - \mu^i\right)\left(x_j^i - \mu^i\right)^T$$
$$\sum\nolimits_B = \sum_{i=1}^{C} P^i \sum\nolimits_B^i = \frac{1}{N} \sum_{i=1}^{C} N_i \sum_{j=1}^{N_i} \left(\mu^i - \mu\right)\left(\mu^i - \mu\right)^T \qquad \textbf{(9)}$$

In above expression $\sum_w^i$ is the covariance matrix estimate for class *i* and computed as

$$\sum\nolimits_w^i = \frac{1}{N_i} = \sum_{j=1}^{N_i} \left(x_j^i - \mu^i\right)\left(x_j^i - \mu^i\right)^T \qquad \textbf{(10)}$$

and $\sum_B^i$ is the scatter matrix between the class *i* and the 'grand class' and computed as

$$\sum\nolimits_B^i = \left(\mu^i - \mu\right)\left(\mu^i - \mu\right)^T \qquad \textbf{(11)}$$

In other words, $\sum w$ is estimated by 'pooling' together $\{\sum_w^i, i = 1 \ldots C\}$. Similarly, this also holds for $\sum B$. Finally, LDA evaluates a projection matrix *W*, say of size *r*×*d*, that maximizes the criterion function [4]

$$J_W = \frac{\det\{W^T \sum_B W\}}{\det\{W^T \sum_W W\}} \tag{12}$$

Above det {.} is matrix determinant. The maximum value of r is $d-1$. For a test pattern y, its class label C_y can be computed as

$$C_Y = \arg\ \min_{i=1,2,..C}\{W^T(y-\mu^i)^2 + D_i\} \tag{13}$$

where D_i is used to incorporate prior information.

Simulation

In case of LDA, sample images and normalized are same as in Figure 2 and 3, respectively. The obtained *fisherfaces* using the LDA algorithm is shown in Figure 7. The mean image is shown in Figure 8, which consists of feature of all the training images. In Figure 9, input and reconstructed image is shown. These images are very much similar, with slight difference in intensity.

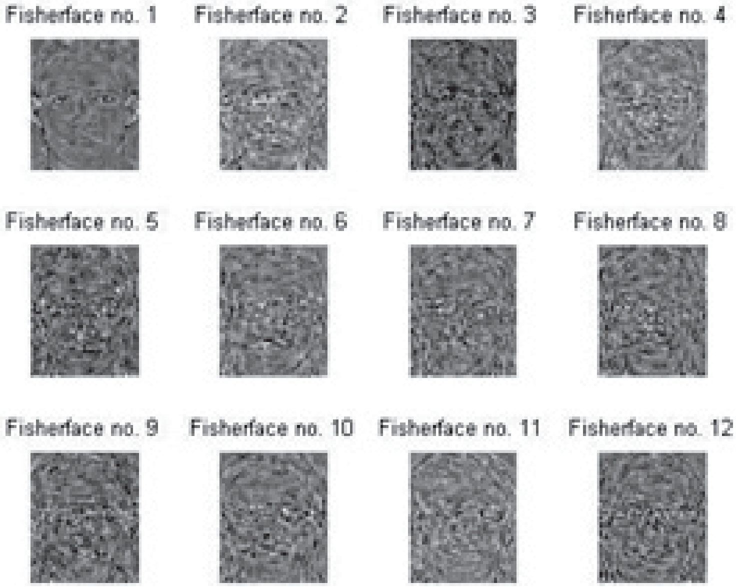

Figure 7: *Fisherface* Images

Figure 8: Mean Images

(a) (b)

Figure 9: (a)Input and (b)Re-constructed images

PERFORMANCE EVALUATION

In the face recognition algorithm, the reconstructed image may or may not be same as input image. The degree of accuracy needs to be measured theoretically. Therefore assessment of the different techniques carried out in the present work was performed on the following criterion.

The FAR and FRR can be described as [6]:

$$FAR = \frac{n_{a_i}}{n_i} \times 100\% \quad (14)$$

$$FRR = \frac{n_{r_c}}{n_c} \times 100\%$$

where: n_{a_i} number of accepted impostor,

n_i number of all impostor identity claims made,

n_{r_c} number of rejected genuine, and

n_c number of all genuine identity claims made.

Results for above test are as follows:

Table 1: Performance of different algorithms for different metric for 400 images

Metric	PCA	ILDA
The equal error rate equals	5.03%	4.28%
The verification rate at 1% FAR equals	86.79%	90.00%
The verification rate at 0.1% FAR equals	66.79%	76.43%
The verification rate at 0.01% FAR equals	45.00%	64.29%

In Table 1, performance of different algorithms for different metric for 400 images is tabulated.

The equal error rate for PCA, and ILDA is 5.03%, and 4.28% respectively. The verification rate at 1% FAR is 90.00% of ILDA and decreases down to 64.29% for verification rate at 0.01% FAR. Thus, as false acceptance rate decreases, the percentage of verification reduces significantly. Now, even in case of ILDA algorithm, at 0.01% FAR the verification rate is poor. In comparison, the performance of ILDA algorithm is better than the PCA algorithm but not acceptable. Therefore, in security prone areas these methods alone can't provide robust solution.

FUZZIFICATION OF FACE RECOGNITION METHOD

The idea of Fuzzification of the Face and fingerprint recognition techniques is shown in Figure 10. For the input of the Fuzzifier the selected membership function is $\prod$ and at the output of the fuzzy system the membership function is chosen to be Δ. As the samples follow I.I.D. process, therefore, a truncated Gaussian membership function known as Π function is selected. In the function α, β and γ defined as minimum, maximum and mean value of the training data set. The c_1, c_2 define the values at which the membership function takes a value of 0.5.

$$\Pi(z;\alpha,\gamma,\beta)=\begin{cases}0 & z\le\alpha\\ 2^{m-1}\left(\dfrac{z-\alpha}{\gamma-\alpha}\right)^m & \alpha<z\le c_1\\ 1-2^{m-1}\left(\dfrac{\gamma-z}{\gamma-\alpha}\right)^m & c_1<z\le\gamma\\ 2^{m-1}\left(\dfrac{z-\gamma}{\beta-\gamma}\right)^m & \gamma<z\le c_2\\ 1-2^{m-1}\left(\dfrac{\beta-z}{\beta-\gamma}\right)^m & c_2<z\le\beta\\ 0 & z\ge\beta\end{cases} \quad \textbf{(15)}$$

The value of the *m* can be selected to alter the shape of the Π function. In this work the value of *m* is considered to be 2.

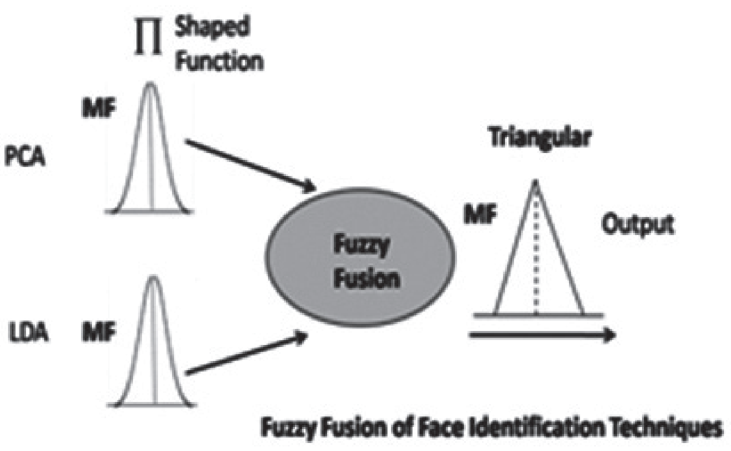

Figure 10: Fuzzy fusion of Face Techniques

It is clear from above expression that the shape and structure of the Π function can be altered by varying the mentioned parameters.

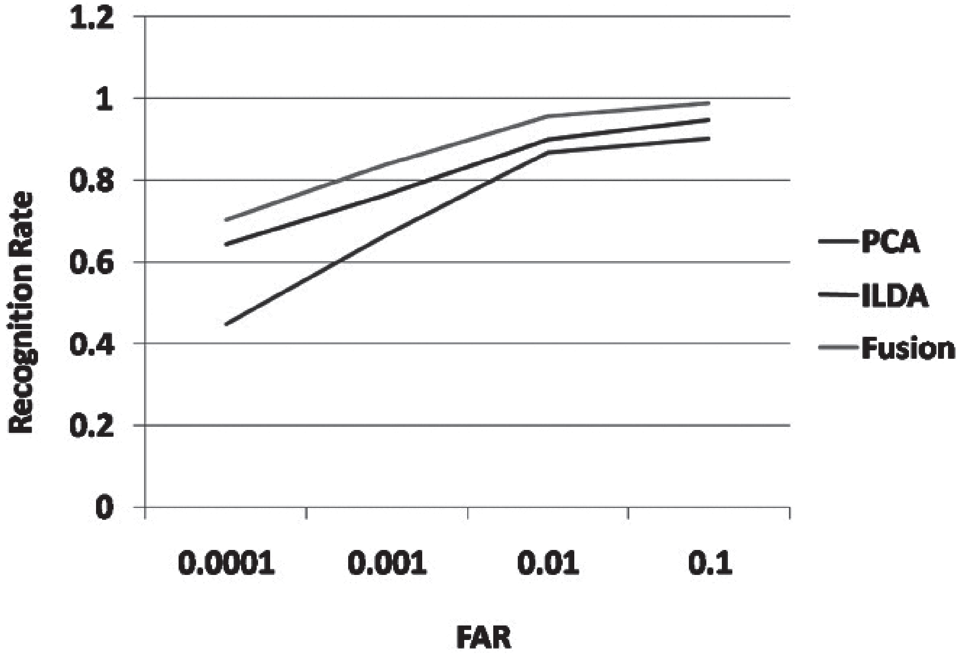

Figure 11: FAR vs. Recognition rate

Figure 11 shows the face recognition rate for PCA, LDA, and for the fusion of these two processes. The performance of PCA method is poorest. As for the FAR < 0.001, the recognition rate is only 67 %, and for the FAR < 0.01, the recognition rate is 86 %. The performance of the LDA method is better in comparison to PCA method. For both the methods, as the FAR increases, the recognition rate increases. However as the large FAR is not acceptable in most of the applications, therefore above two methods are combined using fuzzy methods and the obtained results are superior in comparison to others as for FAR < 0.001, the recognition rate is 83.92 %, which can be reached at the level of 98.73% for FAR of 0.1. The obtained preliminary results are promising and provide a basic foundation for the future research.

CONCLUSION

In this paper, two face recognition algorithms PCA and ILDA are discussed. These algorithms are not the most accurate of face recognition techniques but definitely are one of the easiest to implement. Due to time restrictions in real time systems, these methods were adopted over more accurate methods. The face recognition program returned good results with a few discrepancies. The eigenvalues and eigenvectors of a group of images were calculated correctly. Faces were able to be compared correctly using their decomposition coefficients. Faces were successfully verified as user faces and then the corresponding user was successfully logged on. The only problems were with the time it took to recognise a face. Continually loading images from a database took its toll on the overall recognition time. The comparison of two algorithms is done in terms for, verification rate at 1, 0.1, 0.01% FAR etc. Overall the performance of ILDA is much better than PCA algorithms and fuzzy fusion of two algorithms further improves the results.

REFERENCES

1. Jain, A.K.; Bolle, R. & Pankanti, S. (Eds.).1999. *Biometrics: Personal Identification in Networked Security*. Dordrecht, Netherlands: Kluwer Academic Publishers.
2. Jafri, Rabia & Arabnia, Hamid R.2009. ‘A Survey of Face Recognition Techniques’, *Journal of Information Processing Systems*, Vol.5, No.2, June 2009.
3. Turk, M. and Pentland, A. 1991. “Eigenfaces for recognition”, *J. Cognitive Neuroscience*, Vol. **3**,pp. 71-86.
4. Delac, K.; Grgic, M. & Grgic, S. 2006. Independent Comparative Study of PCA, ICA, and LDA on the FERET Data Set, *International Journal of Imaging Systems and Technology*, Vol. **15**, 5:252-260
5. Beveridge, J.R.; She, K.; Draper, B.A. & Givens, G.H. 2001. *A Nonparametric Statistical Comparison of Principal Component and Linear Discriminant Subspaces for Face Recognition,* CVPR, 2001: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Published by *IEEE*, Vol.**1**, 1:535:542.
6. Bellhumer, P. N.; Hespanha, J & Kriegman, D.1997. “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. **17**, 7:711–720.