American Journal of Applied Sciences 2 (6): 1872-1875, 2006
ISSN 1546-9239

Face Recognition using Eigenfaces and Neural Networks

[1]Mohamed Rizon, [2]Muhammad Firdaus Hashim, [2]Puteh Saad, [1]Sazali Yaacob,
[3]Mohd Rozailan Mamat, [2]Ali Yeon Md Shakaff, [2]Abdul Rahman Saad, [1]Hazri Desa and [1]M. Karthigayan
[1]School of Mechatronics Engineering
[2]School of Computer and Communication Engineering
Kolej Universiti Kejuruteraan Utara Malaysia, Jalan Kangar-Arau 02600 Jejawi, Perlis, Malaysia
[3]Terengganu Advanced Technical Institute (TATI), Jalan Panchor, Teluk Kalong
24000 Kemaman, Terengganu, Malaysia

Abstract: In this study, we develop a computational model to identify the face of an unknown person's by applying eigenfaces. The eigenfaces has been applied to extract the basic face of the human face images. The eigenfaces is then projecting onto human faces to identify unique features vectors. This significant features vector can be used to identify an unknown face by using the backpropagation neural network that utilized euclidean distance for classification and recognition. The ORL database for this investigation consists of 40 people with various 400 face images had been used for the learning. The eigenfaces including implemented Jacobi's method for eigenvalues and eigenvectors has been performed. The classification and recognition using backpropagation neural network showed impressive positive result to classify face images.

Key words: Feature vector, eigenfaces, eigenvalues, eigenvector

INTRODUCTION

The developing of face recognition system is quite difficult because human faces is quite complex, multidimensional and corresponding on environment changes. For that reason the human machine recognition of human faces is a challenching problem due the changes in the face identity and variation between images of the same due to illumination and viewing derection. The issues are how the features adopted to represent a face under environmental changes and how we classify a new face image based on the chosen representation. Computers that recognize human faces systems have been applied in many applications such as security system, mug shot matching and model-based video coding.

The eigenfaces is well known method for face recognition. Sirovich and Kirby[1] had efficiently representing human faces using principle component analysis. M.A Turk and Alex P. Pentland[2] developed the near real-time eigenfaces systems for face recognition using eigenfaces and Euclidean distance.

We develop a technique to extract features from an intensity image of human frontal face to represent the features using eigenfaces. Figure 1 shows the block diagram of our system. These advantages no face features being required, the ability to learn and later recognize new faces in an unsupervised manner and that it is easy to implement using neural network architecture.

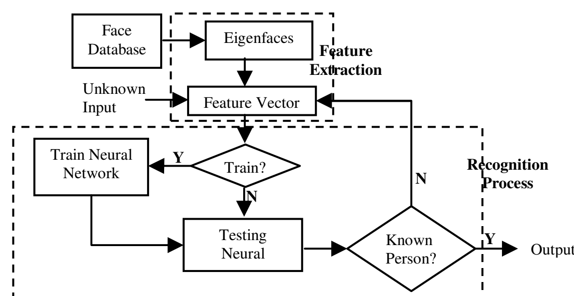


Fig. 1: Face recognition system

The research is focused to develop the computational model of face recognition that is fast, simple and accurate in different environments. Therefore, in this paper the eigenfaces method is described and then it is demonstrated that the features vectors obtained from the eigenfaces can easily be used for classification and recognition.

Eigenfaces method: The basic idea of eigenfaces is that all face images are similar in all configurations and they can be described in its basic face images. Based on this idea, the eigenfaces procedures[3] are as follows:

a. We assume the training sets of images are $\Gamma_1, \Gamma_2, \ldots, \Gamma_m$ with each image is $I(x, y)$. Convert each image into set of vectors and new full-size matrix $(m \times p)$, where m is the number of training images and p is $x \times y$.

Corresponding Author: Mohamed Rizon, School of Mechatronic Engineering, Kolej Universiti Kejuruteraan Utara Malaysia, Jalan Kangar-Arau, 02600 Jejawi, Perlis, Malaysia, Tel: +6049798335, Fax: +6049798334

b. Find the mean face by:

$$\Psi = \frac{1}{m}\sum_{i=1}^{m}\Gamma_i \tag{1}$$

c. Calculated the mean-subtracted face:

$$\Phi_i = \Gamma_i - \Psi, i = 1, 2, \ldots, m \tag{2}$$

and a set of matrix is obtained with $A = [\Phi_1, \Phi_2, \ldots, \Phi_m]$ is the mean-subtracted matrix vector with its size A_{mp}.

d. By implementing the matrix transformations, the vectors matrix is reduced by:

$$C_{mm} = A_{mp} \times A_{pm}^{T} \tag{3}$$

where C is the covariance matrix and T is transpose matrix.

e. Find the eigenvectors, V_{mm} and eigenvalues, λ_m from the C matrix using Jacobi method[4-7] and ordered the eigenvectors by highest eigenvalues. Jacobi's method is chosen because its accuracy and reliability than other method[8,9].

f. Apply the eigenvectors matrix, V_{mm} and adjusted matrix, Φ_m. These vectors determine linear combinations of the training set images to form the eigenfaces, U_k by:

$$U_k = \sum_{n=1}^{m}\Phi_n V_{kn}, k = 1, 2 \ldots, m \tag{4}$$

Instead of using m eigenfaces, $m' < m$ which we consider the image provided for training are more than 1 for each individuals or class. m' is the total class used.

g. Based on the eigenfaces, each image have its face vector by:

$$W_k = U_k^T(\Gamma - \Psi), k = 1, 2, \ldots, m' \tag{5}$$

and mean subtracted vector of size $(p \times 1)$ and eigenfaces is $U_{pm'}$. The weights form a feature vector:

$$\Omega^T = [w_1, w_2, \ldots, w_{m'}]$$

h. A face can reconstructed by using its feature, Ω^T vector and previous eigenfaces, $U_{m'}$ as :

$$\Gamma' = \Psi + \Phi_f \tag{6}$$

where $\Phi_f = \sum_{i=1}^{m'} w_i U_i$.

RESULTS AND DISCUSSION

The Code for eigenfaces is developed using Visual C++. The eigenvectors and eigenvalues play a major role in producing eigenfaces. The results obtained are compared with MATLAB and eigenvalues and eigenvectors java applet[10].

The experiments have been conducted using the Olivetti Research Laboratory (ORL) database (Fig. 2). Figure 3 shows the mean image after the transformation of training images. The eigenfaces result has been obtained (Fig. 4-6).

From the Fig. 4-6 each training session shows the variations of eigenfaces. In Fig. 4, 16 images are used (8 classes with 2 images per-class), Fig. 5 used 32 images (8 classes with 4 images per-class) and Fig. 6 used 48 images (8 classes with 6 images per-class). The eigenfaces above shows exactly, if the experiments is conducted using more images, the eigenfaces becomes more whitening. Means, lesser images make the eigenfaces become darker and indistinct. Sirovich and Kirby evaluated a limited version of this framework on an ensemble of 115 images of Caucasian males digitized in a controlled manner, and found that 40 images were sufficient for a very good description of face images.

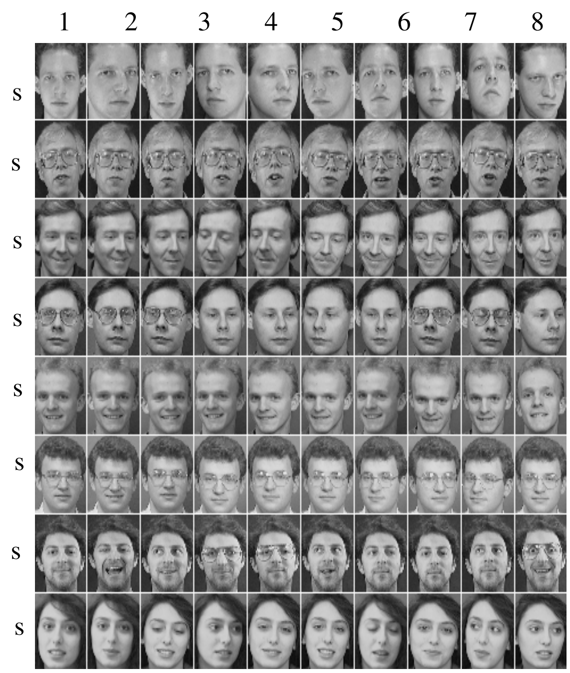


Fig. 2: Some example of the ORL face database that has scale, illumination, expression and pose

Fig. 3: Mean image, Ψ

The eigenfaces used for each training images and unknown images to determine its weight vectors to describe class identity (equation 5). These features are used for classification and recognize the unknown human face.

Table 1: Example the original weight feature vectors, Ω^T

Ω^T / Γ_i	w_1	w_2	....	w_8
Γ_1	7774753.959189	8802601.177177	....	6917700.938766
Γ_2	27859705.231132	32696166.830578	....	29307673.817428
Γ_3	-4875383.270625	-6547910.700724	....	-6594768.822582
Γ_4	-4820597.908669	-6032421.641661	....	-6834500.429694
Γ_5	-6717293.696702	-7592655.784587	....	-7268246.998013
⋮	⋮	⋮	⋮	⋮
Γ_{16}	-7518927.783058	-9070221.818964	....	-7926633.853074

Table 2: Example normalization feature vectors

Ω^T / Γ_i	w_1	w_2	....	w_8
Γ_1	0.539188	1.000000	....	0.154948
Γ_2	0.010280	1.000000	....	0.306589
Γ_3	1.000000	0.162313	....	0.138844
Γ_4	1.000000	0.417547	....	0.032034
Γ_5	0.647926	0.000000	....	0.240121
⋮	⋮	⋮	⋮	⋮
Γ_{16}	0.643505	0.000000	....	0.474381

Table 3: Training Result using Backpropagation Neural Network

Pattern	Actual Target			Network Output			MSE
s1	0	0	0	0.023054	0.002209	0.017150	0.000277
s1	0	0	0	0.066030	0.002194	0.008177	0.001477
s2	0	0	1	0.006697	0.017725	0.971296	0.000394
s2	0	0	1	0.006466	0.006999	0.995777	0.000036
s3	0	1	0	0.005084	0.993605	0.015986	0.000107
s3	0	1	0	0.006870	0.999276	0.013192	0.000074
s4	0	1	1	0.001566	0.987113	0.993907	0.000069
s4	0	1	1	0.006877	0.998734	0.990349	0.000047
s5	1	0	0	0.994456	0.004343	0.012045	0.000065
s5	1	0	0	0.893595	0.005204	0.012156	0.003832
s6	1	0	1	0.988798	0.017077	0.997288	0.000141
s6	1	0	1	0.988189	0.015871	0.997172	0.000133
s7	1	1	0	0.991254	0.978264	0.050029	0.001017
s7	1	1	0	0.983754	0.999764	0.048246	0.000864
s8	1	1	1	0.956998	0.998838	0.995266	0.000624
s8	1	1	1	0.965853	0.999469	0.955677	0.001044

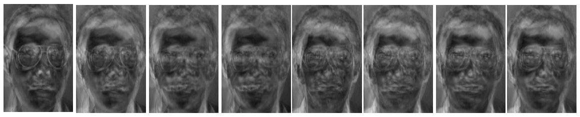
Epoch MSE = 0.000638
Epoch = 136

Some previous work[2,11] used these features to recognize unknown human face using euclidean distance. Table 1, shows the example result of weight vectors for 16 images (8 classes with 2 images per-class).

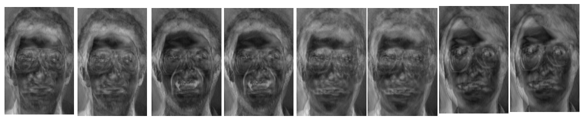
The features vectors used into backpropagation neural network for classification and recognition human faces[3,12,13]. Before the learning phase, the previous feature vectors Ω^T is normalize to a range (0, 1) to meet the backpropagation neural network requirement, avoid computational problems and to facilitate learning[14]. Table 2 shows the normalize features (0, 1) from the original features in Table 1.

The backpropagation neural network is used for the classification and recognition purposes. Table 3 shows the training results using neural network. In this experiment, 16 patterns are used, 8 inputs per-pattern, 5 hidden neurons, 3 output neurons, 0.9 for momentum, 0.7 for learning rate and the error were set to 0.001 for stopping condition.

In the recognition step, the identity of human face is determined if any network output error value is less

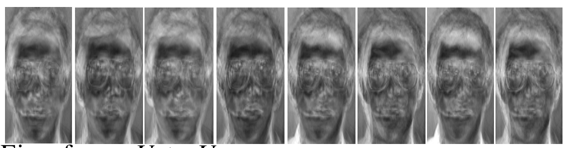


Eigenfaces; U_1 to U_8

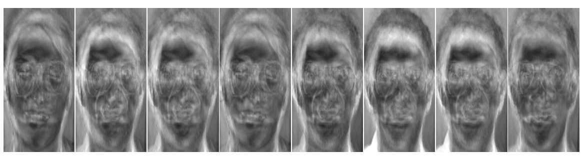


Eigenfaces; U_9 to U_{16}

Fig. 4: The eigenfaces from class s1 to s8 with 2 image per-class

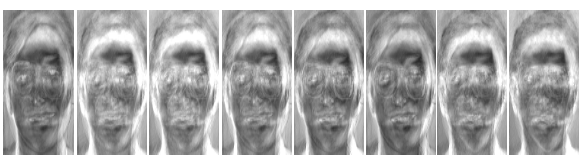


Eigenfaces; U_1 to U_8



Eigenfaces; U_{16} to U_{24}

Fig. 5: The eigenfaces from class s1 to s8 with 4 images per-class



Eigenfaces: U_{25} to U_{32}

Fig. 6: The eigenfaces from class s1 to s8, 6 images per-class

than error (0.001). The recognition rate worked perfectly if the entire training pattern used for recognition. The recognition performance is decrease dramatically if only one image per class used in learning phase. However, when face images with different pose are added in learning step, the recognition rate increase.

CONCLUSION

In this study, we used the eigenfaces to represent the features vectors for human faces. The features are extracted from the original image to represents unique identity used as inputs to the neural network to measure similarity in classification and recognition. The eigenfaces has proven the capability to provide the significant features and reduces the input size for neural network. Thus, the network speed for recognition is raise.

REFERENCES

1. Kirby *et al*., 1990. Application of the Karhunen-Loeve procedure for the characterization of human faces. EEE Trans. Pattern Analysis and Machine Intelligence, 12: 103-108.
2. Turk, M.A. and A.L. Pentland, 1991. Face recognition using eigenfaces. Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, pp: 586-591.
3. Zhujie, Y.L.Y., 1994. Face recognition with eigenfaces. Proc. IEEE Intl. Conf. Industrial Technol., pp: 434-438.
4. Firdaus, M. *et al*., 2005. An approach of feature extraction for face recognition. Proc. Conf. Intelligent Systems and Robotics (iCISAR2005). CD-ROM.
5. Firdaus, *et al*., 005. Face recognition using neural network. Proc. Intl. Conf. Intelligent Systems (ICIS), CD-ROM.
6. Firdaus, M. *et al*., 2006. Dimensions reductions for face recognition using principal component analysis. Proc. 11th Intl. Symp. Artificial life and Robotics (AROB 11th 06). CD-ROM.
7. Mathew, J.H., 2003. Eigenvalues and Eigenvectors, pp: 612-621.
8. Demmel, J. and K. Veselic, 1989. Jacobi's method is more accurate than QR. Technical Report: UT-CS-89-88.1-60.
9. Health, M.T., 2002. Scientific Computing: An Introductory Survey. McGraw-Hill, pp: 191-200.
10. Java applet for Eigenvalues and Eigenvectors, Citing internet sources URL:http://users.ugent.be/~mvdaele/cgi-bin/ILONA/theorie/Applets/Jacobi/ Jacobi.html.
11. Santiago Serrano, citing internet sources URL:www.pages.drexel.edu/~sis26 /Eigencode.html
12. Debipersand, S.C. and A.D Broadhurst, 1997. Face recognition using neural network. Proc. IEEE Commun. Signal Processing (COMSIG'97), pp: 33-36.
13. Nazish, *et al*., 2001. Face recognition using neural network. Proc. IEEE INMIC 2001, pp: 277-281.
14. Saad, P., *et al*., 2001. A comparison of feature normalization techniques on complex image recognition. Proc. 2nd Conf. Information Technology in Asia, pp: 397-409.