Face Recognition Using Eigenfaces

Prof. V.P. Kshirsagar, M.R.Baviskar, M.E.Gaikwad,
Dept. of CSE,
Govt. Engineering College, Aurangabad (MS), India.
vkshirsagar@gmail.com, madhumita_baviskar@yahoo.co.in, gaikwad.manoraj@gmail.com

***Abstract*— Face is a complex multidimensional visual model and developing a computational model for face recognition is difficult. The paper presents a methodology for face recognition based on information theory approach of coding and decoding the face image. Proposed methodology is connection of two stages – Feature extraction using Principle Component Analysis and recognition using the feed forward back propagation Neural Network. The goal is to implement the system (model) for a particular face and distinguish it from a large number of stored faces with some real-time variations as well. The Eigenface approach uses Principal Component Analysis (PCA) algorithm for the recognition of the images. It gives us efficient way to find the lower dimensional space.**

Keywords- Face Recognition; Principal Component Analysis; Eigenfaces; Eigen values; Eigenvector;

I. INTRODUCTION

This paper is a step towards developing a face recognition system which can recognize static images and can be modified to work with dynamic images. In that case the dynamic images received from the camera can first be converted in to the static ones and then the same procedure can be applied on them. The scheme is based on an information theory approach that decomposes face images into a small set of characteristic feature images called 'Eigenfaces', which are actually the principal components of the initial training set of face images. Recognition is performed by projecting a new image into the subspace spanned by the Eigenfaces ('face space') and then classifying the face by comparing its position in the face space with the positions of the known individuals. The Eigenface approach gives us efficient way to find this lower dimensional space. Eigenfaces are the Eigenvectors which are representative of each of the dimensions of this face space and they can be considered as various face features. Any face can be expressed as linear combinations of the singular vectors of the set of faces, and these singular vectors are eigenvectors of the covariance matrices.

In general, face recognition techniques can be divided into two groups based on the face representation they use:

1) Appearance-based, which uses holistic texture features and is applied to either whole-face or specific regions in a face image.

2) Feature-based, which uses geometric facial features (mouth, eyes, brows, cheeks etc), and geometric relationships between them.

II. FACE RECOGNITION SYSTEM

Face recognition is a pattern recognition task performed specifically on faces. It can be described as classifying a face either "known" or "unknown", after comparing it with stored known individuals. It is also desirable to have a system that has the ability of learning to recognize unknown faces.

The outline of typical face recognition system is given in Figure 1.

There are six main functional blocks, whose responsibilities are as below:

A. The acquisition module

This is the entry point of the face recognition process. The user gives the face image as the input to face recognition system in this module.

B. The pre-processing module

In this module the images are normalized and enhanced to improve the recognition of the system. The pre-processing steps implemented are as follows:

- Image size normalization
- Histogram equalization
- Median filtering
- High-pass filtering
- Background removal
- Translation and rotational normalizations
- Illumination normalization

C. The feature extraction module

After the pre-processing the normalized face image is given as input to the feature extraction module to find the key features that will be used for classification. The module composes a feature vector that is well enough to represent the face image.

D. The classification module

With the help of a pattern classifier, the extracted features of face image is compared with the ones stored in the face database. The face image is then classified as either known or unknown

978-1-61284-840-2/11/$26.00 ©2011 IEEE

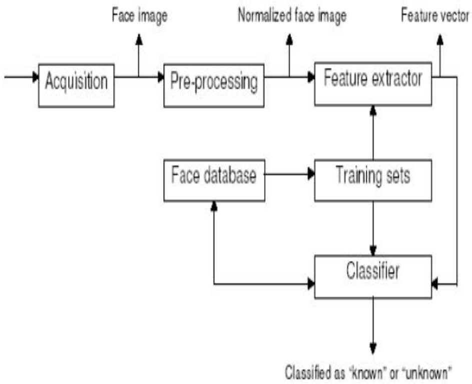

Figure 1. Outline of typical face recognition system

E. The classification module

With the help of a pattern classifier, the extracted features of face image is compared with the ones stored in the face database. The face image is then classified as either known or unknown.

F. Training set

Training sets are used during the learning phase of the face recognition process.

G. Face database

If the face is recognized as "unknown", face images can then be added to the database for further comparisons.

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.

III. Eigenface Approach

Focus is made towards developing a sort of unsupervised pattern recognition scheme that does not depend on excessive geometry and computations like deformable templates. Eigenfaces approach seemed to be an adequate method to be used in face recognition due to its simplicity, speed and learning capability. Eigenfaces are a set of eigenvectors used in the computer vision problem of human face recognition. They refer to an appearance-based approach to face recognition that seeks to capture the variation in a collection of face images and use this information to encode and compare images of individual faces in a holistic manner. Specifically, the eigenfaces are the principal components of a distribution of faces, or equivalently, the eigenvectors of the covariance matrix of the set of face images, where an image with N x N pixels is considered a point (or vector) in N2-dimensional space [6].

Eigenfaces are mostly used to:

- Extract the relevant facial information, which may or may not be directly related to human intuition of face features such as the eyes, nose, and lips. One way to do so is to capture the statistical variation between face images.
- Represent face images efficiently. To reduce the computation and space complexity, each face image can be represented using a small number of dimensions.

IV. Eigen Values and Eigen Vectors [2]

In linear algebra, the eigenvectors of a linear operator are non-zero vectors which, when operated on by the operator, result in a scalar multiple of them. The scalar is then called the eigenvalue (**λ**) associated with the eigenvector(**X**). Eigen vector is a vector that is scaled by a linear transformation. It is a property of a matrix. When a matrix acts on it, only the vector magnitude is changed not the direction.

$$AX=\lambda X \tag{1}$$

Where A is a Vector function.

V. Calculations of Eigen Values and Eigen Vectors

By using (1), we have the equation,

$$(A-\lambda I)X=0 \tag{2}$$

Where **I** is the n x n Identity matrix. This is a homogeneous system of equations, and from fundamental linear algebra, we know that a nontrivial solution exists if and only if

$$\det(A-\lambda I)=0 \tag{3}$$

Where **det()** denotes determinant. When evaluated, becomes a polynomial of degree n. This is known as the characteristic equation of A, and the corresponding polynomial is the characteristic polynomial. The characteristic polynomial is of
degree n. If A is n x n, then there are n solutions or n roots of the characteristic polynomial. Thus there are n eigenvalues of A satisfying the equation,

$$AX_i=\lambda X_i \tag{4}$$

Where i=1, 2, 3….n

If the eigenvalues are all distinct, there are n associated linearly independent eigenvectors, whose directions are unique, which span an n dimensional Euclidean space.

VI. Approach Followed For Face Recognition Using Eigenfaces

The whole recognition process involves two steps [1],

- Initialization process
- Recognition process

The Initialization process involves the following operations:

1) Acquire the initial set of face images called as training set as shown in Figure 2.

Figure 2. Sample Faces

2) Calculate the eigenfaces from the training set, keeping only the highest eigenvalues. These M images define the *face* space. As new faces are experienced, the eigenfaces can be updated or recalculated. (Figure 3.)

3) Calculate the corresponding distribution in M-dimensional weight space for each known individual, by projecting their face images on to the "face space". (Figure 4)

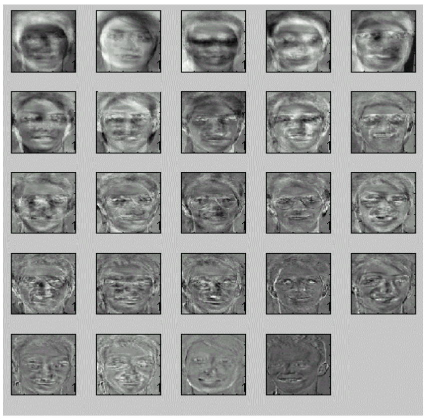

Figure 3. Eigenfaces for Sample faces

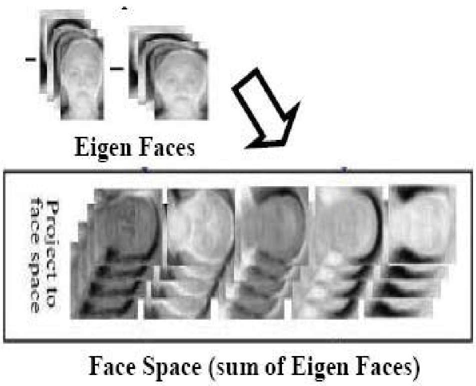

Figure 4. Describes about Eigen Faces [7]

These operations can be performed from time to time whenever there is a free excess operational capacity. This data can be cached which can be used in the further steps eliminating the overhead of re-initializing, decreasing execution time thereby increasing the performance of the entire system. Having initialized the system, the next process involves the steps [6],

a) Calculate a set of weights based on the input image and the M eigenfaces by projecting the input image onto each of the eigenfaces

b) Determine if the image is a face at all (known or unknown) by checking to see if the image is sufficiently close to a "free space".

c) If it is a face, then classify the weight pattern as either a known person or as unknown.

d) Update the eigenfaces or weights as either a known or unknown. If the same unknown person face is seen several times then calculate the characteristic weight pattern and incorporate into known faces.

The last step is not usually a requirement of every system and hence the steps are left optional and can be implemented as when the there is a requirement.

VII. 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 $\Gamma_1, \Gamma_2, .., \Gamma_m$ with each image is I (x, y). Convert each image into set of vectors and new full-size matrix(m×p), where m is the number of training images and p is x × y.

b) Find the mean face by:

$$\psi = \frac{1}{m}\sum_{i=1}^{m}\Gamma_i \tag{7}$$

c) Calculated the mean-subtracted face:

$$\Phi i = \Gamma i - \Psi, \quad i = 1, 2, \ldots, m \tag{8}$$

$A = [\Phi 1, \Phi 2, \ldots, \Phi m]$ is the mean-subtracted matrix vector with its size A_{mp} .

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

$$C_{mn} = A_{mp} \times A^T_{pm} \tag{9}$$

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

e) Find the eigenvectors, Vmm and eigenvalues, λm from the C matrix using Jacobi method and ordered the eigenvectors by highest eigenvalues. Jacobi's method is chosen because its accuracy and reliability than other method.

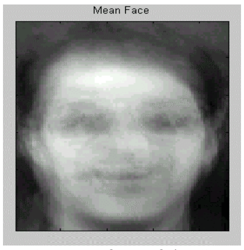

Figure 5. Average face of the Sample faces

f) Apply the eigenvectors matrix, Vmm and adjusted matrix, Φm . These vectors determine linear combinations of the training set images to form the eigenfaces, Uk by:

$$U_k = \sum_{n=1}^{m} \Phi_n \; V_{kn} \, , \; k=1,2\ldots,m \tag{10}$$

Instead of using m eigenfaces, $m' \leq m$ 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:

$$W_k = U_k^T \,(\Gamma - \Psi), \quad k = 1,2,\ldots,m' \tag{11}$$

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, ΩT vector and previous eigenfaces, $U_{m'}$ *as :*

$$\Gamma' = \Psi + \Phi_f \tag{12}$$

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

The above procedure takes place in a Face Recognition System such as shown in the Figure 6:

To summarize whole process of Face Recognition using Eigenface approach from the above diagram, the training set of images are given as input to find eigenspace. Using these images, the average face image is computed. The difference of these images is represented by covariance matrix. This is used to calculate Eigenvectors and Eigenvalues. These are the Eigenfaces which represent various face features. Sort the eigenvalues, and consider higher of them since they represent maximum variations. This becomes eigenspace spanned by the eigenfaces, which has lower dimension than original images. Now given two test images are projected onto this eigenspace to give the weight vector also known as Face key for that image. The Euclidean distance between these two face key vectors is calculated. If this is below some threshold value, then two images are said to be matching that means they belong to same person. Depending on this result, False Acceptation Rate (FAR) and False Rejection Rate (FRR) are found. These are used to change value of Threshold. In this way Face Recognition is carried out using Eigenface Approach.

VIII. Drawbacks of Eigenfaces Approach

The tests conducted on various subjects in different environments shows that this approach has limitations over the variations in light, size and in the head orientation**,** nevertheless this method showed very good classifications of faces(>85% success rate).

A noisy image or partially occluded face would cause recognition performance to degrade.

IX. Applications

Various potential applications are:

- Person Identification
- Human-Computer interaction
- Security and Surveillance Capability Systems

X. Summary

In this paper, the dimensionality problems are solved for face recognition. The approach using Eigenfaces and PCA is quite robust in the treatment of face images with varied facial expressions as well as the directions. However, this approach is sensitive to images with uncontrolled illumination conditions.

XI. 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] *"Facial Recognition using Eigenfaces by PCA"* by Prof. Y. Vijaya Lata, Chandra Kiran Bharadwaj Tungathurthi, H. Ram Mohan Rao, Dr.A. Govardhan, Dr.L.P. Reddy, International Journal of Recent Trends in Engineering, Vol. 1, No. 1, May 2009.

[2] *"Face Recognition using Eigenface Approach"* by Vinaya Hiremath, Ashwini Mayakar, .

[3] *"Face Recognition using Eigenfaces and Neural Networks"* by Mohd Rozailan Mamat, Mohamed Rizon, Muhammad Firdaus Hasim, American Journal of Applied Sciences 2 (6): 1872-1875, 2006 ISSN 1546-9239, 2006 Science Publications.

[4] *"Face Recognition using Eigen Faces and Artificial Neural Network"* by Mayank Agarwal, Nikunj Jain, Mr. Manish Kumar and Himanshu Agrawal. International Journal of Computer Theory and Engineering, Vol. 2, No. 4, August, 2010,1793-8201

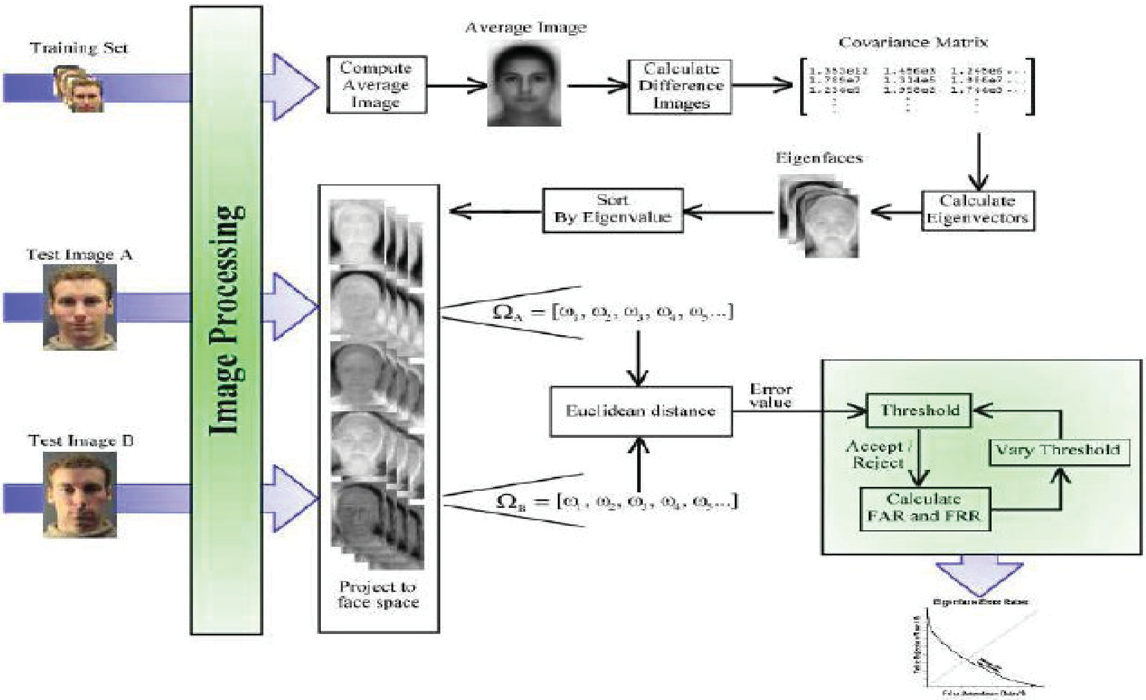

Figure 6. Face Recognition System [7]

[5] *"Biometrics- A Look At Facial Recognition"* by John D. Woodward, Jr., Christopher Horn, Julius Gatune, and Aryn Thomas

[6] *"Face Recognition using Eigenfaces and Distance Classifiers: A Tutorial"* by Shubhendu Trivedi, February 11, 2009.

[7] " *Evaluation of image preprocessingtechniques for eigenface based facerecognition"* Nick Pears Thomas Heseltine and Jim Austin.. ACA Group, Dept.of Computer Science, University of York, 2002.

[8] "*Face Recognition Using Eigenfaces"*, by Matthew A. Turk and Alex P. Pentland, MIT Vision and Modeling Lab, CVPR '91.

[9] "*Eigenfaces for Recognition"* by Matthew A. Turk and Alex P. Pentland, Journal of Cognitive Neuroscience '91.