REAL TIME FACE RECOGNITION SYSTEM USING EIGEN FACES

Prof. B.S PATIL[1] Prof. A.R YARDI[2], ,Dr Mrs Patil S B[3]

1 HOD, IT DEPARTMENT PVPIT, BUDHGAON ,2.DY DIRECTOR WALCHAND COLLEGE OF ENGG.SANGLI

3.HOD ELECTRONICS DEPARTMENT,JJMCOE,JAYSINGPUR

ABSTRACT

Face recognition have been fast growing, challenging and interesting area in real-time applications. A large number of face recognition algorithms have been developed from decades. The present paper primarily focuses on principal component analysis, for the analysis, the software is implemented using Matlab and C#.net This face recognition system detects the faces in a picture taken by web-cam, and these face images are then checked with training image dataset based on Eigen features. Eigen features are used to characterize images.

Keywords—Eigen faces, eigenvalues PCA, face recognition, person identification, face classification,

---

I. INTRODUCTION

Face recognition systems have been grabbing high attention from commercial market point of view as well as pattern recognition field. Face recognition has received substantial attention from researches in biometrics, pattern recognition field and computer vision communities. The face recognition systems can extract the features of face and compare this with the existing database. The faces considered here for comparison are still faces. Machine recognition of faces from still and video images is emerging as an active research area. The present paper is formulated based on still or video images captured by a web cam.

The face recognition system extracts the Eigen features from trainee set. It later compares with the database of faces, which is collection of faces in different poses. The present system is trained with the database shown in Figure (1), where the images are taken in different poses like head variation , light variation, scale variation , feature variation means with glasses, with and without beard.

II. EIGEN FACES

Eigen faces are a set of eigenvectors used in the computer vision problem of human face recognition. Eigen faces assume ghastly appearance. 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. Specifically, the Eigen faces 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 NxN pixels is considered a point (or vector) in N^2-dimensional space. Eigen faces is still considered as the baseline comparison method to demonstrate the minimum expected performance of such a system.

Eigen faces are mostly used to: a. 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. b. Represent face images efficiently. To reduce the computation and space complexity, each face image can be represented using a small number of dimensions The Eigen faces may be considered as a set of features which characterize the global variation among face images. Then each face image is approximated using a subset of the Eigen faces, those associated with the largest eigenvalues. These features account for the most variance in the training set.

In the language of information theory, we want to extract the relevant information in face image, encode it as efficiently as possible, and compare one face with a database of models encoded similarly. A simple approach to extracting the information contained in an image is to somehow capture the variations in a collection of face images, independently encode and compare individual face images.

Mathematically, it is simply finding the principal components of the distribution of faces, or the eigenvectors of the covariance matrix of the set of face images, treating an image as a point or a vector in a very high dimensional space. The eigenvectors are ordered, each one accounting for a different amount of the variations among the face images. These eigenvectors can be imagined as a set of features that together characterize the variation between face images. Each image locations contribute more or less to each eigenvector, so that we can display the eigenvector as a sort if “ghostly” face which we call an Eigen face.

Each of the individual faces can be represented exactly in terms of linear combinations of the Eigen faces. Each face can also be approximated using only the "best" Eigen face, which has the largest eigenvalues, and the set of the face images. The best M Eigen faces span an M dimensional space called as the "Face Space" of all the images.

The basic idea using the Eigen faces was proposed by Sirovich and Kirby, using the principal component analysis, starting with an ensemble of original face image they calculated a best coordinate system for image compression where each coordinate is actually an image that they termed an Eigen picture. They argued that at least in principle, any collection of face images can be approximately reconstructed by storing a small collection of weights for each face and small set if standard picture (the Eigen picture). The weights that describe a face can be calculated by projecting each image onto the Eigen picture. Also according to the Turk and Pentland[1], the magnitude of face images can be reconstructed by the weighted sums of the small collection of characteristic feature or Eigen pictures and an efficient way to learn and recognize faces could be to build up the characteristic features by experience over feature weights needed to (approximately) reconstruct them with the weights associated with Matched individuals. Each individual, therefore would be characterized by the small set of features or Eigen picture weights needed to describe and reconstruct them, which is an extremely compact representation of the images when compared to themselves.

A. Approach followed for facial recognition using Eigen faces The whole recognition process involves three steps,

1. Acquire the initial set of face images called as training set.

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

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

B. The face recognition process involves following steps,

1. Calculate a set of weights based on the input image and the M Eigen faces by projecting the input image onto each of the Eigen faces

2. Determine if the image is a face at all (Matched or unmatched) by checking to see if the image is sufficiently close to a training image set

3. Calculate Euclidian distance between Test image and trainee set images , if distance is below threshold value then Test image is matched else unmatched.

III. FACIAL RECOGNITION BASED ON PRINCIPAL COMPONANT ANALYSIS

A. Generating Eigen faces

Assume a face image I(x,y) be a two-dimensional M by N array of intensity values, or a vector of dimension MxN. The Training set used for the analysis is of size 92x112, resulting in 10,304 dimensional space. A typical image of size 256 by 256 describes a vector of dimension 65,536, or, equivalently, a point in 65,536-dimensional space. For simplicity the face images are assumed to be of size NxN resulting in a point in N^2 dimensional space. An ensemble of images, then, maps to a collection of points in this huge space.

The main idea of the principal component analysis is to find the vectors which best account for the distribution of face images within the entire image space. These vectors define the subspace of face images, which we call "face space". Each vector is of length N^2, describes an N by N image, and is a linear combination of the original face images. Because these vectors are the eigenvectors of the covariance matrix corresponding to the original face images, and because they are face like in appearance, we refer to them as “Eigen faces”.

The Training set images used for the analysis purpose are shown in the Figure (1) and the Eigen faces for the training sets are shown in the Figure (2).

Fig-1 Trainee set images of one user

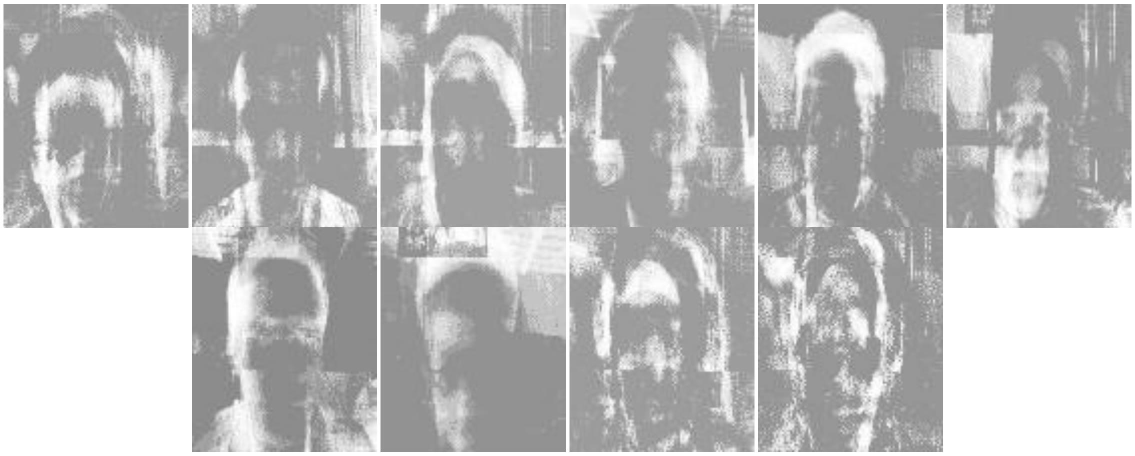

Figure 2 Eigen Faces of above Training images

Let the training set of face images be $\Gamma_1\ \Gamma_2 \ldots \Gamma_M$. The average face of the set is defined by

$\Psi = (1/M)\ \Sigma\, \Gamma_k$

Each face differs from the average by the vector $\Phi_i = \Gamma i - \Psi$.

An example training set is shown in Figure (1), with the average face Ψ shown in Figure (3).

Fig.3 Average Face for the training set shown in Figure (1)

This set of very large vectors is then subject to principal component analysis, which seeks a set of M vectors, uk , which best describes the distribution of the data. The k^{th} vector is u_k chosen such that,

$$\lambda_k = \frac{1}{M}\left(u_k^T \Phi_n\right)^2$$

The vectors u_k and λ_k scalars are eigenvectors and eigenvalues, respectively, of the covariance matrix

$$C = \frac{1}{M} \Sigma_{n=1}^{M} \Phi \,.\, \Phi^{T}$$
$$= A \,.\, A^{T}$$

Where the matrix $_M$ A= $[\Phi_1, \Phi_2, \Phi_3 \ldots\ldots \Phi_L]$

The matrix C, however, is N^2 x $N^{\,2}$ by N , and determining the N eigenvectors and eigenvalues is an intractable task for typical image sizes.

A Computationally feasible method is to be funded to calculate these eigenvectors. If the number of data points in the image space is $M(M<N^2)$, there will be only M-1 meaningful eigenvectors, rather than N^2. The eigenvectors can be determined by solving much smaller matrix of the order M^2xM^2 which reduces the computations from the order of N^2 to M, pixels. Therefore we construct the matrix L

$$L = A \,.\, A^{T}$$
$$A^{T} \,.\, A$$

Fig. 1 The Training images that have been used for the analysis and find the M eigenvector u^l of L . These vectors determine linear combination of the M training set face images to form the Eigen faces v^l

$$v_l = \sum u_{lk} . \Phi_k$$
$$\text{where } l = 1 \cdots M$$

IV. CLASSIFATION AND IDENTIFICATION OF FACE

Once the Eigen faces are created, identification becomes a pattern recognition task. The Eigen faces span an N^2-dimensional subspace of the original A image space. The M' significant eigenvectors of the L matrix are chosen as those with the largest associated eigenvalues.

The Euclidean distance between two weight vectors d(i,j) provides a measure of similarity between the corresponding images i and j. If the Euclidean distance between Test and Trainee faces exceeds some threshold value, then Test face is not present.

$$d(\boldsymbol{A}, \boldsymbol{B}) = \sqrt{\sum_{i=1}^{D} (a_i - b_i)^2} = \|\boldsymbol{A} - \boldsymbol{B}\|$$

V. IMPLEMENTATION IN MATLAB & RESULTS

The above discussed methodologies have been implemented in Matlab ,

The image database generated using application developed in c#.net through which we capture the 10 images of each class as a trainee images in different poses. The test images by varying head, scale, features and light are captured using same application.

The Algorithm has been tested on above generated own Image databases. We also have created an Image Database having 12 users each with 10 facial postures and the so a total of 120 images.

Following figure shows the Test images with variations for recognition.

			
Feature Variation	**Head Variation**	**Scale Variation**	**Light Variation**

Fig.4 Test Images in different poses

And the results from the above implementation are as shown in fig-5

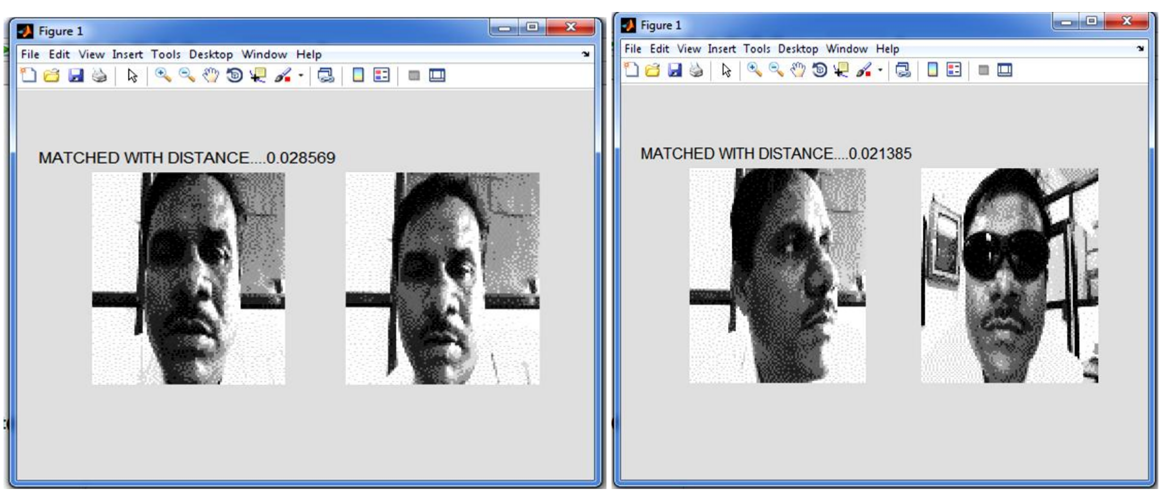

Fig-5 output of software implemented in MATLAB

Table 1 showing the success and error rates of face recognition on own Image Database having 120 images in different conditions

Table-1

Variation	**SUCCESSS %**	**ERROR %**
Head	89.75%	10.25%
Light	91.38%	8.62%
Scale	93.44%	6.56%
Feature	92.20%	7.80%
Total Efficiency	**91.60%**	**8.4%**

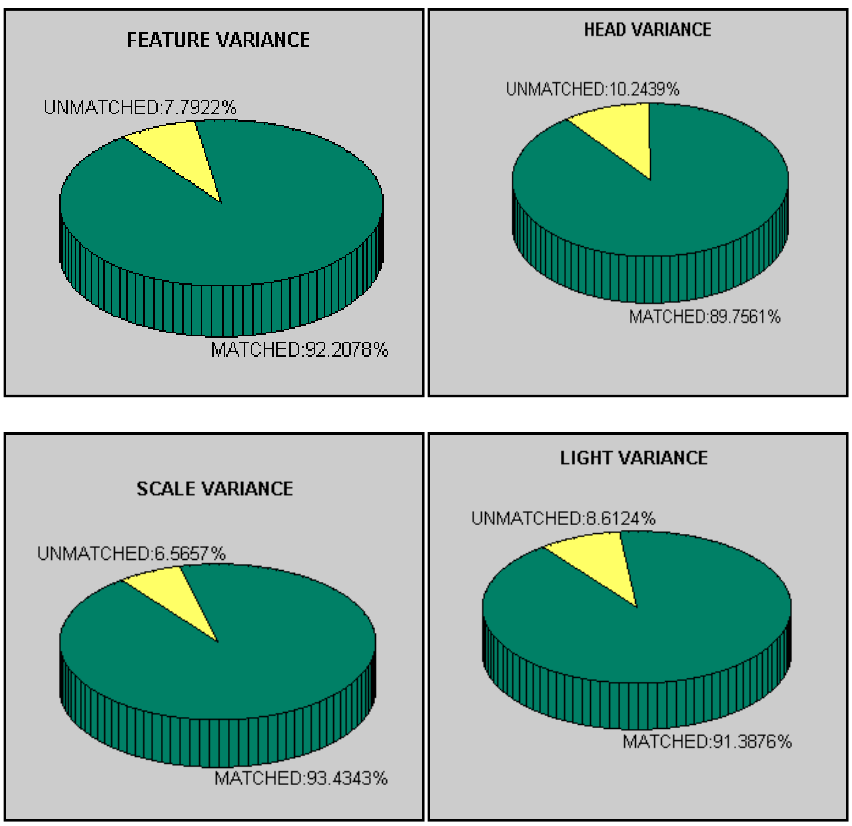

Fig.6 Graph of matched/unmatched percentage

V. CONCLUSION

The tests conducted on various users in different environments shows that this approach has limitations over the variations in light and head orientation, however this method showed very good recognition in feature and Scale variations. The overall success rate is above 91%.

When an image is sufficiently close to face-like but is not classified as one of the familiar faces, it is initially labeled as "unmatched". A noisy image or partially obstructed face would cause recognition performance to degrade. The eigenface approach does provide a practical solution that is well fitted to the problem of face recognition. It is fast, relatively simple, and has been shown to work more accurate in constrained environment.

REFERENCES

[1] M.Turk and A. Pentland, "Eigen faces for Recognition", Journal of Cognitive Neuroscience, March 1991.

[2] M.A. Turk and A.P. Pentland. “Face recognition using Eigen faces”. In Proc. of Computer Vision and Pattern Recognition, pages 586-591. IEEE, June 1991b.

[3] L.I. Smith. “A tutorial on principal components analysis”

[4] Delac K., Grgic M., Grgic S., “Independent Comparative Study of PCA, ICA, and LDA on the FERET Data Set”, International Journal of Imaging Systems and Technology, Vol. 15, Issue 5, 2006, pp. 252-260

[5] H. Moon, P.J. Phillips, “Computational and Performance aspects of PCA-based Face Recognition Algorithms”, Perception, Vol. 30, 2001, pp. 303-321

[6] Matlab Online Documentation “http://

[7] Aditya kelkar,”Face recognition using Eigen faces Approach”

[8] Dimitri Pissarenko, “Eigenface-based facial recognition”

[9] Ming-Hsuan Yang, “Recent Advances in Face Recognition”

[11] W. Zhao, R. Chellappa, P.J. Phillips and A. Rosenfeld, “ Face Recognition: A Literature Survey”

[12] Jon Shlens, “A Tutorial on Principal Component Analysis Derivation, Discusson and Singular Value Decomposition”, 25 March 2003, Version 1