Two Level Security System Using MATLAB

Sunil Kumar Sharma[1], Sharad Chouhan[2]

[1]Student, M. Tech, ESSEAR, Ambala

[2]Assistant Professor, Dept. of CSE, E-Max group of Institutions, Ambala

Abstract—--- **Two levels security system include Face Recognition and Password based entry control. Two level security(Face recognition and PIN) can provide hi-tech security at public place including International Airport, Metro-Station, Shopping Mall, banking ATM control and home etc. It increase human computer interaction which is fast and secure. To *avoid the misuse of property and to counter terrorism and thefts etc. security devices are implemented. So this paper discussed here two levels Security system.* This paper addresses first level i.e. the building of face recognition system by using Principal Component Analysis (PCA) method and second level Edsim51Di simulator for keypad entry and finally controlling the motor. A PCA algorithm is written on MATLAB. PCA [31], [33], [37] is a statistical approach used for reducing the number of variables in face recognition. As extracting the most information (feature) contained in the images (face). In PCA, every image in the training set can be represented as a linear combination of weighted eigenvectors called as "Eigen faces". These eigenvectors are obtained from covariance matrix of a training image set called as basis function. Recognition is performed by projecting a new image (test image) onto the subspace spanned by the Eigen faces and then classification is done by distance measure methods such as Euclidean distance. Finally, when the face is matched, MATLAB program generate the 8bit hex code which is received at the port of 8051, further this project is simulated with the help of Edsim51di, first matches face code from database and secondly password entered by keypad which is interface with 8051 to match with the stored database and motor is rotated to allow the entry. If any one condition not satisfies it display "No Entry" at LC*D.***

Keywords— **Principal Component Analysis (PCA), , Eigen faces, Euclidean distance.**

I. INTRODUCTION

The face recognition is ideal for high traffic areas which are open to the general public for e.g. airports and railway stations, ATM's, public transportationThese days' numbers of terrorism and thefts cases are increasing day by day all over the nation and world. Securityat the public places like Metro station, car parking, ATM machine, markets homes have become the prime concern. In given scenario it becomes the need of the time to use such type of security device which are automated, fast and secure. On the basis of existing studies as discussed in previous sections most of the people have worked either on biometrics, "Face Recognition" [11], [12], [13] or numerically controlled personal identification number (PIN) based security systems [38]. There are only few studies which have considered the hybridized aspects of security systems. This was the prime motivation to design a hybrid security system combining biometric and numeric password based approach for security purpose.

There are many algorithms to detect the unknown face from the data base. These algorithms are:

1. Principal Component Analysis
2. Discrete Cosine Transform
3. 3D recognition methods
4. Gabor Wavelets method
5. Hidden Markov Models
6. Kernel methods

These algorithms are further categorized on the basis of image mode face recognition characteristics. A detail of this categorization has been shown in figure 2.1

There are many characteristics to be considered for choosing a face recognition method. The keys ones are:

1. Accuracy
2. Time limitations
3. Process speed
4. Availability

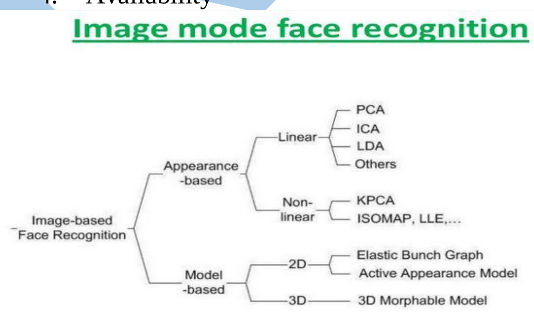

Fig 2.1 Image Mode Face recognition.

Out of all the above face recognition algorithms and characteristics, Principle Component Analysis (PCA) [30-35] is effective feature extraction method based on face as a global feature. It reduces the dimension of image effectively and holds the primary information at the same time.With these in mind the PCA method of face recognition was selected for this project because:

1. Simplest and easiest method to implement – due to project deadlines this method seemed the most practical.
2. Very fast computation time.
3. Accurate – this method is definitely not the most accurate of face recognition algorithms but considering the requirements of this project it was judged to be accurate enough.

II. BASIC THEORY OF PCA:

Principal component analysis (PCA) was worked out by Karl Pearsonin 1901[31].The Principal Component Analysis (PCA) is one of the most successful techniques that have been used in image recognition and compression. PCA is a statistical method under the broad title of factoranalysis. The jobs which PCA can do are prediction, redundancy removal, feature extraction, data compression, etc. Because PCA is a classical technique which can do something in the linear domain, applications having linear models such as signal processing, image processing, system and control theory, communications, etc.are suitable for analysis.

PCA [31], [33], [37] is a statistical approach used for reducing the number of variables in face recognition.It is a variable reduction procedure and useful when obtained data have some redundancy. While extracting the most relevant information (feature) contained in the images (face). This will reduce the features of an image into reduced number of variables which are called Principal Components. These principal components represent the most variance of the observed variable.In PCA, every image in the training set can be represented as a linear combination of weighted eigenvectors called as "Eigen faces". These eigenvectors are obtained from covariance matrix of a training image set called as basis function. The weights are found out after selecting a set of most relevant Eigen faces. Recognition is performed by projecting a new image (test image) onto the subspace spanned by the Eigen faces and then classification is done by distance measure methods such as Euclidean distance.

Face recognition has many applicable areas. Moreover, it can be categorized into face identification, face classification. The most useful applications contain crowd surveillance, video content indexing, personal identification (ex. driver's license), entrance security, etc.

In using PCA for face recognition the image data is statistically interpreted as follow:

- PCA projects the data along the directions where the data varies the most.
- These directions are determined by the eigenvectors of the covariance matrix corresponding to the largest eigenvalues.
- The magnitude of the eigenvalues corresponds to the variance of the data along the eigenvector directions.

III PCA Algorithms

The PCA algorithm is implemented in following six steps.

***Step1:*Face Image Representation**

A *2-D* facial image can be represented as *1-D* vector by concatenating each column into a long thin vector. Training sets of images is construct in such a way that different person have their five different facial expressions each. M is total no. of training images. Training set consists of *M*differentimages of size *PxQ*isconverted into a vector of size*PQx1*.

Example: $2 * 2$Matrixis converted into column matrix as shown in (1) below.

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}_{2\,X\,2} \rightarrow \begin{bmatrix} 2 \\ 1 \\ 1 \\ 2 \end{bmatrix}_{4\,X\,1} \quad \ldots.(1)$$

Each face of training set is represented by vectorX_1; X_2; X_3; …; X_M.Actual size of each image is taken fixed i.e. $P * Q$. Here only for demonstration size of matrix is taken as $2*2$. Pixel vector $X_1, X_2 \ldots X_M$generated from training images are shown by equation (2).

$$X_1 = \begin{bmatrix} 1 \\ -2 \\ 1 \\ -3 \end{bmatrix}_{4\,X\,1}, \; X_2 = \begin{bmatrix} 1 \\ 3 \\ -1 \\ 2 \end{bmatrix}_{4\,X\,1}, \; X_3 = \begin{bmatrix} 2 \\ 1 \\ -2 \\ 3 \end{bmatrix}_{4\,X\,1}, \ldots X_M = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}_{4\,X\,1} \quad \ldots (2)$$

Pixel vector of a training image (face) is stored in a matrixX_i

$$X_i = [X_1 X_2 X_3 \ldots X_M]_{PQxM}, \; i = 1..M \quad \ldots (3)$$

Let consider a gray-Scale image of size *112*92* as shown in fig2.2 which is represented in the form of matrix and further is converted into a column matrix. This transformation is shown in the following equation (4).

Fig 2.2 an image (Face)

$$\begin{bmatrix} 222 & \cdots & 233 \\ \vdots & \ddots & \vdots \\ 167 & \cdots & 211 \end{bmatrix}_{112\,X92} \begin{bmatrix} 223 \\ \vdots \\ 211 \end{bmatrix}_{10304\,X\,1} \longrightarrow \quad \ldots\ldots (4)$$

Step2:Calculation of Mean and Mean Centered Images

In this step the column matrix obtained from different training images (facial images) in step1 is operated upon to compute mean and mean centered images as shown below.

2.1 Computing *mean* face vector *m*:

$$m = \frac{1}{M}\sum_{i=1}^{M} X_i \ldots (5)$$

Or

$$m = (X_1 + X_2 + X_3 + \cdots \ldots + X_M)/M \ldots (6)$$

$$m = 1/M \left[\begin{bmatrix} 1 \\ -1 \\ 1 \\ -3 \end{bmatrix} + \begin{bmatrix} 1 \\ 3 \\ -1 \\ 2 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \\ -3 \\ 1 \end{bmatrix} + \cdots \ldots \ldots \ldots \ldots \begin{bmatrix} 2 \\ 1 \\ 2 \\ 2 \end{bmatrix} \right] \ldots (7)$$

M is total number of training images, m is mean and $X_i$is the pixel Vector of M training images.

2.2 Computingmean centered image

Mean calculated above is subtracted from each training image of training set to make new mean centered images.

The mean centered imagesΦ_i are computed as

$$\Phi_i = X_i - m \;\; ; i = 1 \ldots. M \ldots (8)$$

This Mean centered images would be used to construct the co-variance matrix. Covariance Matrix concept is explains in next step.

***Step3:*Calculatingcovariance matrix:**

Definition: Covariance is a measure of how much tworandom variables (X, Y) change together [39].Or it is measure of the spread of data in a data set.

General covariance is

$$Cov(X,Y) = \frac{1}{(n-1)}\sum_{i=0}^{n}(Xi - \bar{X})(Yi - \bar{Y}) \ldots. (9)$$

Where X, Y are data representing in two dimensional matrix rows and column, n is length of data elements.

A covariance C matrix is calculated as:

$$C = A * A^T \ldots (10)$$

$$\text{Where} A = [\Phi_1, \Phi_2, \Phi_3 \ \ldots. \Phi_M]_{PQ*M} \ldots (11)$$

$\Phi_1, \Phi_2, \Phi_3 \ \ldots. \Phi_M$ are mean centered images.

The Calculation of Co-variance of a matrix is demonstrated below

$$\text{Let } A = \begin{bmatrix} 2 & 1 \\ 3 & 1 \\ 3 & 4 \end{bmatrix}_{3\,x\,2}, \text{ than} A^T = \begin{bmatrix} 2 & 3 & 3 \\ 1 & 1 & 4 \end{bmatrix}_{2\,x\,3} \ldots (12)$$

$$C = A * A^T = \begin{bmatrix} 2 & 1 \\ 3 & 1 \\ 3 & 4 \end{bmatrix} * \begin{bmatrix} 2 & 3 & 3 \\ 1 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 5 & 7 & 10 \\ 7 & 10 & 13 \\ 10 & 13 & 25 \end{bmatrix}_{3\,x\,3} \ldots (13)$$

Here size of covariance matrix is *3 x 3*, but in actual formation of Co-variance matrix size of matrix will be *PQ x PQ*.

Eigen vectors corresponding to this covariance matrix $C = A * A^T$ is needed to be calculated, but that will be a tedious task, because actual size of matrix would be very large therefore,

- For simplicity we calculate $A^T * A$ which would be a *2 x 2* matrix in this case but in actual case size of Matrix would be *MxM*

Covariance matrix is determined as $C = A^T * A$ instead of $C = A * A^T$ which will give a *2 x 2* matrix as shown below in (14).

$$A^T * A = \begin{bmatrix} 22 & 17 \\ 17 & 18 \end{bmatrix}_{2\,x\,2} \ldots (14)$$

Calculation of Eigen-vector and Eigen value of the Co-Variance matrix is explained in following step.

Step 4: Eigen-Vector of Covariance Matrix

In this step the above covariance matrix is used to finda set of eigenvectors which described the best principal information in each of the mean centered images. The set of eigenvectors (V_i)with the highest eigenvalues (λ_i) are required desired ones.

Definition:

$$AV = \lambda V \qquad \ldots (15)$$

A is square matrix, V is Eigen vector and $\lambda$is Eigen Value.

- In linear algebra, the eigenvectors of a linear operator are non-zero vector.. Scalar is then called Eigen value (λ) associated with the eigenvector (V).
- An eigenvector of a matrix is a vector such that, if multiplied with the matrix, the result is always an integer multiple of that vector.
- Eigen vector is a vector that is scaled by linear transformation. ..

Eigenvectors possess following properties:

- It can be determined only for square matrices
- There are n eigenvectors (and corresponding eigenvalues) in an $n \times n$Matrix.
- All eigenvectors are right angle with each other.

Here in figure *2.3*show projection of relevant data. Two Eigen vector u_1, u_2 are shown perpendicular to each other are generated from two dimensional data to largest Eigen value that describes the direction where most the data varies.

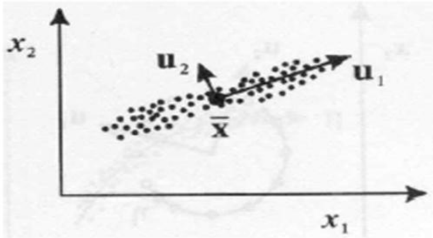

Fig 2.3 diagram showing projection of relevant data

Calculation of Eigen vector:

For Example:

From the definition given above, Eigen vector of matrix is defined by (16)

$$AV = \lambda V \qquad \ldots (16)$$

Where A is a square matrix, V is Eigen vector, λ is Eigen value.

$$(A - \lambda I)V = 0 \ldots (17)$$

Where I is the identity matrix

This is a homogeneous system of equations and form fundamental linear algebra.

$$|A - \lambda I| = 0. \quad \ldots (18)$$

When evaluated becomes a degree of polynomial

In this A *is* n *x* n then there are n solutions or n roots of polynomial. so there are n Eigen values of A the equation.

$$A * V_i = \lambda_i * V_i \ldots (19)$$

Where $i = 1, 2, 3 \ldots, n$.

Eigen values, are n associated linearly independent eigenvectors if they are distinct, whose directions are unique, which span an n dimensional Euclidean space.

Consider the eigenvectors $V_i$of$A^T * A$ such that

$$(A^T * A)\, V_i = \lambda_i V_i \qquad \ldots (20)$$

Characteristics equation is formed of above equation, for calculating the Eigen value and Eigen vector.

Co-variance matrix (C)constructed in (14) is used for calculating Eigen vector and Eigenvalue using equation (18) shown in (21-22) respectively.

$$V_1 = \begin{bmatrix} 0.6645 \\ -0.7473 \end{bmatrix}_{2\,x1}, V_2 = \begin{bmatrix} 0.7473 \\ -0.6645 \end{bmatrix}_{2\,x1} \ldots (21)$$

$$\lambda_1 = 2.8828, \ \lambda_2 = 37.1172 \quad \ldots (22)$$

.

Step: 5.1Eigen Face Approach

Eigen Face Approach is an adequate and efficient method to be used in face recognition due to, speed , learning capability and simplicity. Eigen faces are used in the Computer Vision problem of human face recognition. This 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.

Eigen faces are Principal Components of a distribute the faces, or equivalently, the Eigen vectors of the matrix of the set of the face images, where an image with *P x Q* pixels is considered a point in *PQ*

We wish to find Principal Components of the distribution of faces, or the Eigen vectors of the matrix of the set of face images and image location can contribute to each Eigen vector, so that it can display the Eigen vector as face image is represented in exactly terms of linear combination of the

Eigen faces. Possible Eigen faces is equal to the number of face image in the training set. The faces can also be approximated by using best Eigen face, those that have the largest Eigen values, and therefore account for most variance between the set of face images. Primary reason for using Eigen faces is computational efficiency.

Step 5.2 Eigen Face Space:

The Eigen vectors of the covariance matrix $C = A^T * A$ are V_i. $V_i$resembles facial images which look ghostly and. Eigen vectors correspond to each Eigen face in the face space and.The Eigen faces are ranked according to their usefulness in characterizing the variation among the images.

Face image is projected into the face space by:

$$\Omega_i = V_i^T(X_i - m);\ i = 1 \ldots . M \ldots (23)$$

Where$(X_i - m)$ can be mean center image,V_i^T is Transpose of V_i.Hence projection of each image can be obtained as $\Omega_1$for projection of image1 and$\Omega_2$for projection of image2 and hence forth.

Step6: Recognition Step

The test image X is projected into the face space to obtain a vector, Ω(omega) as

$$\Omega = V_i^T(X - m) \quad \ldots (24)$$

The distance ofΩ to each face is called **Euclidean distance**(E_d) and defined by

$$E_d^2 = \|\Omega - \Omega_i\|^2;\ i = 1 \ldots\ldots M \ldots (25)$$

Where, Ω_i is a vector describing the i[th]face class.

A face is classified as belonging to class i when the minimum $E_d$is below some chosen **threshold** (T_h). otherwise the face is classified as unknown.

$T_h$is half the largest distance between any two face images:

$$T_h = \left(\frac{1}{2}\right) \max_j i \left\|\Omega_j - \Omega_i\right\| \quad \ldots (26)$$

Where *i, j=1…….M*

We have to find the distance E between the original Test image X and its reconstructed image from the Eigen face X_i^f

$$E = \left\|X - X_i^f\right\| \quad \ldots (27)$$

$$\text{Where} X_i^f = V_M * \Omega_i + m \quad \ldots (28)$$

Step7:Result

1. If $E >= T_h$then input image is not even a face image and not recognized.
2. If $E < T_h$ and $E_d >= T_h$ for all $i$then input image is a face image but it is recognized as unknown face.
3. If $E < T_h$and $E_d < T_h$for all i then input images are the individual face image associated with the class vector Ω_i.

IV TOOL USED

Two levels security system of MATLAB will be used in my dissertation to implement the algorithm.MATLAB (Matrix Laboratory) was invented in late 1970s by Cleve Moler. It is a high-level language and its interactive environment helps us to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and FORTRAN. Another important feature of MATLAB . We can easily build models from scratch, or take an existing model and add to it. We have instant access to all the analysis tools in MATLAB, so we can take the results and analyze and visualize them. , an environment that encourages us to pose a question, model it, and see what happens. However today it is much more powerful.

V CONCLUSION AND FUTURE SCOPE

PCA based face recognition and increase the number of Eigen value will increase the recognition rate However, the recognition rate saturates after a certain amount of increase in the Eigen value.. Simultaneously 8 bit password based system make working at edsim51 simple and fast. Increase in the no. of bit or digits make little complex and time consuming. At last hybrid combination first face recognition and second password based security able to work properly and are efficient and economy to person. [37]

Future Scope: -

As two level security (Face recognition and Keypad entry), is automated with computer.So, it would be used where less man are required for surveillance.The face recognition is ideal for high traffic areas which are open to the general public for e.g. airports and railway stations, ATM's, public transportation, businesses of all kinds and home. It would increase security to electronic gadgets like Mobile, Personal computer, PDA etc. As Face recognition technology has been already working among major cities. But this hybrid technique (face recognition and password) bring advancement in security system. So, it would increase the human–computer interaction. Also it is fast and secure so it consumes less time and increase efficiency

REFERENCES

[1]. Craw, D. Tock, and A. Bennett, "Finding face features," Proc. of 2nd European Conf. Computer Vision. pp. 92-96, 1992.

[2]. A. Lanitis, C. J. Taylor, and T. F. Cootes, "An automatic face identification system using flexible appearance models," Image and Vision Computing, vol.13, no.5, pp.393-401, 1995.

[3]. T. K. Leung, M. C. Burl, and P. Perona, "Finding faces in cluttered scenes using random labeled graph matching," Proc. 5th IEEE int'l Conf. Computer Vision, pp. 637-644, 1995.

[4]. B. Moghaddam and A. Pentland, "Probabilistic visuallearning for object recognition," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no.7. pp. 696-710, July, 1997.

[5]. M. Turk and A. Pentland, "Eigenfaces for recognition," J. of Cognitive Neuroscience, vol.3, no. 1, pp. 71-86, 1991.

[6]. M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces,"IEEE Trans. Pattern Analysis and Machine Intelligence, vol.12, no.1, pp. 103-108, Jan. 1990.

[7]. I. T. Jolliffe, Principal component analysis, New York: Springer-Verlag,1986.

[8]. T, Agui, Y. Kokubo, H. Nagashi, and T. Nagao,"Extraction of face recognition from

monochromatic photographs using neural networks," Proc. 2nd Int'l Conf. Automation, Robotics, and Computer Vision, vol.1, pp. 18.81-18.8.5, 1992.

[9]. Krestinin, I.A., Seredin, O.S.: "Excluding cascading classifier for face detection." Proc. of the 19th Int. Conf. on Computer Graphics and Vision, 380–381 (2009).

[10]. R. Chellappa., C. L. Wilson., S. Sirohey: "Human and machine recognition of faces: a survey". Proc. IEEE, 83(5), pp. 705—740, 1995.

[11]. S. Z. Li., A. K. Jain.: "Handbook of Face Recognition", Springer, New York, 2005.

[12]. J. R. Solar, P. Navarreto, " Eigen space-based face recognition: a comparative study of different approaches, IEEE Tran., Systems man And Cybernetics- part c: Applications, Vol. 35,No. 3, 2005.

[13]. Sahoolizadeh, H.; Ghassabeh, Y.A.;" Face recognition using eigen-faces, fisher-faces and neural networks, Cybernetic Intelligent Systems, 2008. CIS 2008.7th IEEE International Conference on 9-10 Sept. 2008

[14]. M.Turk., A. Pentland.: "Eigenfces for recognition". Journal of Cognitive Neuroscience, 3(1), pp.71--86, 1991

[15]. A. A. Thomas and M. Wilscy, "Face Recognition Using Simplified Fuzzy ARTMAP",J.Signal and Image Processing, Vol.1, No.2, pp 134-136, December 2010.

[16]. M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, Face Recognition by Independent Component Analysis, IEEE Trans. on Neural Networks, Vol. 13, No. 6, November 2002, pp. 1450-1464 .

[17]. C. Liu, H. Wechsler, Evolutionary Pursuit and Its Application to Face Recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, No. 6, June 2000, pp. 570-582.

[18]. L. Wiskott, J.-M. Fellous, N. Krueuger, C. von der Malsburg, Face Recognition by Elastic Bunch Graph Matching, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, 1997, pp. 775-779 .

