3D FACE RECOGNITION UNDER EXPRESSIONS, OCCLUSIONS AND POSE VARIATION

Puri S.B.[1], Suryawanshi P.T.[2], Bhosale S.D.[3]

[1]*PG Student, Dept. of E&TC, S.T.B. College of Engineering, Tuljapur, India*
[2]*PG Student, Dept. of E&TC, S.T.B. College of Engineering, Tuljapur, India*
[3]*PG Student, Dept. of E&TC, S.T.B. College of Engineering, Tuljapur, India*

Abstract

We propose a novel geometric framework for analysing 3D faces, with the specific goals of comparing, matching, and averaging their shapes. Here we represent facial surfaces by radial curves emanating from the nose tips and use elastic shape analysis of these curves to develop a Riemannian framework for analysing shapes of full facial surfaces. This representation, along with the elastic Riemannian metric, seems natural for measuring facial deformations and is robust to challenges such as large facial expressions (especially those with open mouths), large pose variations, missing parts, and partial occlusions due to glasses, hair, and illustrates the use of radial facial curves on 3D meshes to mode facial deformation caused by expression, occlusion and variation in poses and to recognize faces despite large expression, in presence of occlusion and pose variations. Here we represent facial surface by indexed collection of radial geodesic curves on 3D face meshes emanating from nose tip to the boundary of mesh and compare the facial shapes by comparing shapes of their corresponding curves. We use elastic shape analysis for comparing shapes of facial curves because elastic matching seems natural for facial deformation and is robust to challenges such as large facial expressions (especially those with open mouths), large pose variations, missing parts, and partial occlusions due to glasses, hair, and so on. Our results match or improve upon the state-of-the-art methods on two prominent databases: GavabDB and Bosporus, each posing a different type of challenges.

Keywords: - *3D Face recognition, data restoration, Geodesic curve, neural network, shape analysis, pca method, eigenfaces, NN method.*

---

1. INTRODUCTION

Our face is primary identity, which represents a human being in social life. Each face has unique features, which differ from the other people. So face can be used to define a person’s identity. We are able to recognize a person even after we see them after years; this is because of the primary features present on a human face. May be they are with hairstyle change or with glasses or bearded still we can recognize them. Same can be done by computer to identify the faces and that can be used if any criminal records are there, also verification of identity and in social media applications like Picasa where automatic tagging in pictures is done. In addition, this automatic recognition can be used at airports, image that is captured form video or from CCTV footage.

A face recognition system can be used in two modes: verification (or authentication) and identification. A face verification system involves confirming or denying the identity claimed by a person (one-to-one matching). On the other hand, a face identification system attempts to establish the identity of a given person out of a pool of N people (one-to-N matching). When the identity of the person may not be in the database, this is called open set identification. While verification and identification often share the same classification algorithms, both modes target distinct applications. In verification mode, the main applications concern access control, such as computer or mobile device login, building gate control, digital multimedia data access. Over traditional security access systems, face verification has many advantages: the biometric signature can’t be stolen, lost or transmitted, like for ID card, token, badges or forgotten like passwords or PIN codes. In identification mode, potential applications mainly involve video surveillance (public places, restricted areas), information retrieval (police databases, multimedia data management) or human computer interaction (video games, personal settings identification). Applications of face recognition are very wide. Most commonly used is that of human computer interaction. One can make the computer locking system with his image as opening key. So that another user cannot use the personal computer of any user and the confidential data can be kept on it.

1.1 Challenges in Face Recognition System

There are many problems in face recognition system. Face Recognition System needs to overcome

1. Change of face position, size, and shooting angles
2. Change of face expression
3. Bad environment, such as low-light
4. Real-time recognition
5. Aging
6. Accessories such as glasses

2. LITERATURE SURVEY

In human face the shape and size of nose, eyes and lips varies from person to person. To extract and measure these features some land markings (most important points) are used. The method of face recognition depends upon the application which we want to apply for. For example, if we want to apply for surveillance system which may use the video camera but the image database investigations may require static intensity images taken by a standard camera. The other applications, which are access to top security domains, may even require the quality of face recognition by user to stand in front of a 3D scanner or an infra-red sensor. Therefore, depending on the face data acquisition methods face recognition techniques can be widely divided into three categories: methods those work on intensity images, those operate with video sequences, and those that need other sensory data such as 3D information or infra-red imagery.

2.1 Face Recognition from Intensity Images

Face recognition methods for intensity images are divided into two main categories: feature-based and holistic.

2.1.1 Feature Based approach

Feature based approach processes only remarkable features such as eyes, nose and mouth etc. And then computes the geometric distances of these facial points. Standard statistical pattern recognition techniques are then employed to match faces using these measurements. In the earlier phase of Facial recognition system, the feature based approach has been used by Kanade [2], he used 16 facial points and used a simple Euclidean distance measure for matching to achieve a peak performance of 75% on a database of 20 different people using 2 images per person.

The proposed method uses a Simulated Annealing-based [3] approach (SA) for range image registration with the Surface Interpenetration Measure (SIM), as similarity measure, in order to match two face images. The authentication score is obtained by combining the SIM values corresponding to the matching of our different face regions: circular and elliptical areas around the nose, forehead, and the entire face region. Then, a modified SA approach is proposed taking advantage of invariant face regions to better handle facial expressions. Comprehensive experiments were performed on the FRGC v2 database, the largest available database of 3D face images composed of 4,007 images with different facial expressions.

2.1.2 Image Based approach

Holistic approaches attempt to identify faces using global representations, i.e., descriptions based on the entire image rather than on local features of the face. There have been several methods in past some years that depend upon deforming facial surfaces into one another, under some chosen conditions, and use quantifications of these deformations as metrics for face recognition. Among these, the ones using nonlinear deformations use local stretching, compression, and bending of surfaces to match each other and are referred to as elastic methods. For an example, Kakadiaris et al. [4] utilize an annotated face model to study geometrical variability across faces. The annotated face model is deformed elastically to fit each face, thus matching different anatomical areas such as the nose, eyes, and mouth.

In [5] this paper, face recognition is performed using Principal Component Analysis followed by Linear Discriminant Analysis based dimension reduction techniques. Sequencing of this paper is preprocessing, dimension reduction of training database set by PCA, extraction of features for class separability by LDA and finally testing by nearest mean classification techniques. The proposed method is tested over ORL face database. It is found that recognition rate on this database is 96.35% and hence showing efficiency of the proposed method than previously adopted methods of face recognition systems. With some most significant Eigen features, Eigen face space projection matrix is derived and over it LDA is applied. LDA reduces dimension of applied projection matrix and increases recognition rate and reduces computation time. Classification space projection matrix, derived after LDA, can be classified with various techniques.

In [8] feed forward neural network and back propagation neural networks are used in addition with PCA. The Principal Component Analysis is used here to extract the features from the face images of human frontal face. The database is rearranged in the form of a matrix where each column represents an image. With the help of Eigen values and Eigen vectors covariance matrix is computed. Feature vector for each image is then computed. This feature vector represents the signature of the image. Signature matrix for whole database is then computed. Euclidian distance of the image is computed with all the signatures in the database.

It involves feature extraction using principal component analysis and wavelet decomposition and then classification by creation of back propagation neural network. It gives a comparative analysis of performance of back propagation neural network and single layer feed forward neural network. In this approach variable learning rate is used and its superiority over constant learning rate is demonstrated. Different inner epochs for different input patterns according to their difficulty of recognition are assigned to patterns. The effect of optimum numbers of inner epochs, best variable learning rate and numbers of hidden neurons on training time and recognition accuracy are also shown.

2.2 Existing Algorithms

2.2.1 Principal Component Analysis

Principal component analysis (PCA) is a dimensionality reduction technique which is used for compression and face recognition problems. It is also known as Eigenspace projection or Karhunen-Leave transformation [9]. PCA calculates the Eigen vectors of the covariance matrix, and projects the original data onto a lower dimensional feature space, which is defined by Eigen vectors with large Eigen values. PCA has been used in face representation and recognition where the Eigen vectors calculated are referred to as Eigen faces. PCA is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension. It is one of the more successful techniques of face recognition. The benefit of PCA is to reduce the dimension of the data. No data redundancy is found as components are orthogonal. With help of PCA, complexity of grouping the images can be reduced. The application of PCA is made in criminal investigation, access control for computer, online banking, post office, passport verification, medical records etc.

2.2.2 Elastic Bunch Graph Matching

All human faces share a similar topological structure. Faces are represented as graphs, with nodes positioned at fiducially points (eyes, nose etc.) and edges labelled with 2-D distance vectors. Each node contains set of 40 complex Gabor wavelet coefficient s at different scales and orientations (phase and amplitude). They are called "jets". Recognition is based on labelled graphs. A labelled graph is a set of nodes connected by edges, each node is labelled as jets and edges are labelled as distances.

2.2.3 Independent Component Analysis

This is a technique to extract the statistic of random variables. In this second order and higher order dependencies of the input data are minimized by the technique and the basis along which the data PCA considered image elements as random variables with Gaussian distribution and minimized second-order statistics. Clearly, for any non-Gaussian distribution, largest variances would not correspond to PCA basis vectors. Independent Component Analysis (ICA) (Bartlett et al., 2002; Draper et al., 2003) minimizes both second-order and higher order dependencies in the input data and attempts to find the basis long which the data (when projected onto them) are statistically independent. Bartlett et al. (2002) provided two architectures of ICA for face recognition task: Architecture I – statistically independent basis images and Architecture II – factorial code representation.

2.2.4 Linear Discriminant Analysis

LDA is basically popular for the classification of data into within class and between classes. LDA is widely used to find linear combinations of features while preserving class separability. It tries to maximize the between class scatter to within class scatter. It is a statistical approach for classifying samples of unknown classes based on training samples with known classes. This technique tries to maximize between-class (i.e. across users) variance and minimize within-class (i.e. within user) variance.

2.2.4.1 Class Dependent Information

This approach tries to classify the samples which lie between two classes. This will try to find the best class for the sample which lie in between the classes and try to maximize the ratio of between class variance to within class variance. The main purpose of this approach is to get the best separation between the classes.

2.2.4.2 Class Independent Information

This approach tries to maximize the overall variance of all classes in dataset to within class variance. This approach uses only one optimizing criterion to transform the data sets and hence all data points irrespective of their class identity are transformed.

2.2.5 Local Binary Pattern

LBP is very good means of feature description. The LBP operator was originally designed for texture description. The LBP operator assigns a label to every pixel of an image by thresholding the 3x3-neighborhood of each pixel with the centre pixel value and considering the result as a binary number. A local binary pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. In the computation of the LBP histogram, the patterns used are uniform so that the histogram has a separate bin for every pattern, which is uniform, and all non-uniform patterns are assigned to a single bin. The original LBP operator was using the 8 pixels being the central pixel as a threshold.

3. PROBLEM STATEMENT

3.1 Problem Definition

The face recognition problem can be formulated as follows: Given an input face image and a database of face images of known individuals, how can we verify or determine the identity of the person in the input image. The 3D Face Recognition system is used for verification of any person's identity. If any occlusions are present in the image, then it is not possible to recognize a person manually. In case of identical twins, it is very difficult to recognize the correct person manually. It will be an added expense for any organization for keeping one person for such kind of job. The automatic face recognition system can do this job in some seconds and time also be saved.

3D Face Recognition is an invention that will be useful for the authentication of identity of a certain person. If any person is misusing the belongings like ID cards of other person that is very harmful to original owner of the ID cards. So this automatic identification of face is very useful equipment which can give us the desired result of known or unknown identity of any person. Here we are going to use the ORL (Oracle Research Laboratory database) database of 400 images. Using PCA we will extract the features of the database. Features of the database calculated will be average, mean, Euclidean distance and weights of input vector. And based on the distance the face will be recognized as match found in the database.

3.2 Importance of Face Recognition

Biometric-based techniques have emerged as the most promising option for recognizing individuals in recent years since, instead of authenticating people and granting them access to physical and virtual domains based on passwords, PINs, smart cards, plastic cards, tokens, keys and so forth, these methods examine an individual's physiological and/or behavioural characteristics in order to determine and/or ascertain his identity. Passwords and PINs are hard to remember and can be stolen or guessed; cards, tokens, keys and the like can be misplaced, forgotten, purloined or duplicated; magnetic cards can become corrupted and unreadable. However, an individual's biological traits cannot be misplaced, forgotten, stolen or forged. Biometric-based technologies include identification based on physiological characteristics (such as face, fingerprints, finger geometry, hand geometry, hand veins, palm, iris, retina, ear and voice) and behavioural traits (such as gait, signature and keystroke dynamics) [10]. Face recognition appears to offer several advantages over other biometric methods, a few of which are outlined here: Almost all these technologies require some voluntary action by the user, i.e., the user needs to place his hand on a hand-rest for fingerprinting or hand geometry detection and has to stand in a fixed position in front of a camera for iris or retina identification.

However, face recognition can be done passively without any explicit action or participation on the part of the user since face images can be acquired from a distance by a camera. This is particularly beneficial for security and surveillance purposes. Furthermore, data acquisition in general is fraught with problems for other biometrics: techniques that rely on hands and fingers can be rendered useless if the epidermis tissue is damaged in some way (i.e., bruised or cracked). Iris and retina identification require expensive equipment and are much too sensitive to any body motion. Voice recognition is susceptible to background noises in public places and auditory fluctuations on a phone line or tape recording. Signatures can be modified or forged. However, facial images can be easily obtained 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. Finally, technologies that require multiple individuals to use the same equipment to capture their biological characteristics potentially expose the user to the transmission of germs and impurities from other users. However, face recognition is totally non-intrusive and does not carry any such health risks.

4. METHODOLOGY

4.1 Principal Component Analysis

Principal component analysis (PCA) was invented in 1901 by Karl Pearson. PCA involves a mathematical procedure that transforms a number of possibly correlated variables into a number of uncorrelated variables called principal components, related to the original variables by an orthogonal transformation. This transformation is defined in such a way that the first principal component has as high a variance as possible (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to the preceding components. PCA is sensitive to the relative scaling of the original variables. Depending on the field of application, it is also named the discrete Karhunen–Loève transform (KLT), or proper orthogonal decomposition (POD). The major advantage of PCA is that the Eigen face approach helps reducing the size of the database required for recognition of a test image. The trained images are not stored as raw images rather they are stored as their weights which are found out projecting each and every trained image to the set of Eigen faces obtained.

The PCA algorithm is generally used to take out the important features of an image. These important features which are called as Eigenface in face recognition system's domain. It is very efficient method which is most widely used for face recognition, image analysis etc. PCA not only reduces the dimensions of an image but it keeps some of the variations in image data also. With the help of the PCA algorithm we can transform images in the training datasets into the corresponding Eigenface. This Eigenface is very useful while reconstructing the original image or the input image. Here we have used 40 person's set of images to generate a set of Eigen faces, each person's 10 images per one set. These images are normalized to line up the eyes and mouths. They are then all resample at the same pixel resolution (say $m \times n$) here we have used 256x256 size, and then treated as mxn - dimensional vectors whose components are the values of their pixels. The eigenvectors of the covariance matrix of the statistical distribution of face image vectors are then extracted.

Since the eigenvectors are those vectors which belong to the same vector space as original face images, they can be viewed as if they were $m \times n$ pixel face images: hence the name Eigen faces. In this way the Eigenface will be the average face of all the training sample images. The Eigen face helps us to reduce the dimensions of original data by taking out the only most important features of the input images. So the compression of the data is achieved. Here we are talking about to extract the only important features of the data we have and creating the average face i.e. Eigenface. But the opposite of it is also possible that from the weights we have calculated face can also be recognized. We can subtract the mean image r average face from input image and if the Euclidean distance is minimum than the threshold then the input face is said to be known.

Therefore, using this weights one can determine two important things:

1. The given image is face image or not, if weights of given image differ too much from the face images then it is not a face image.
2. Similar looking face for e.g. twins has similar Eigen faces Similar to some degrees. If one extracts weights from all the images available, the images could be grouped to clusters. That is, all images having similar weights are likely to have similar face.

4.1.1 Block Diagram of PCA System

In the proposed system of PCA and radial curve method there are two stages testing and training as shown in fig 4.1. We first train the database of 400 images i.e. 40 different person's images and 10 images per each person. Then apply radial curves on them. We store these features which are extracted through algorithm. Then we select the input image ageing apply PCA and radial curves on them. We find the Eigen vectors. Then among these Eigen vectors the minimum distance matching with the test image we find match. The face with the minimum Euclidean distance will be our result.

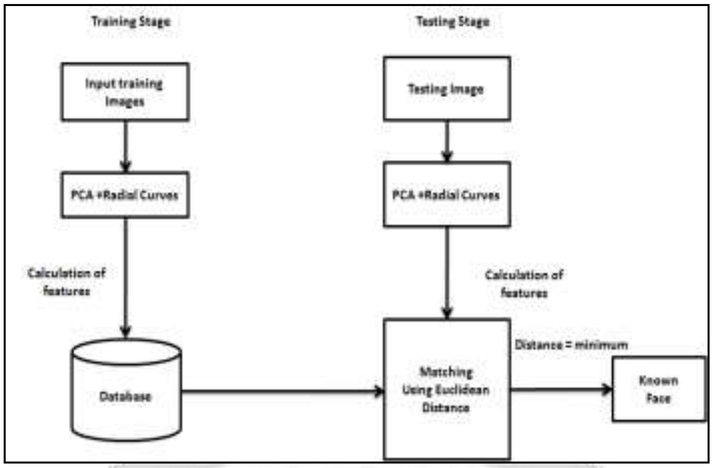

Fig-4.1: Block diagram of PCA system

4.1.2 Algorithm of a PCA with RBC system

1. Load database of Training Images.
2. Pick an image to be searched.
3. Take a mean of all images except the one which is an input.
4. Calculate the covariance matrix.
5. Apply radial curves.
6. Calculate weight vector of input image.
7. Calculate distance between input weight vectors training dataset.
8. If distance (Euclidian) is minimum among all the distance of training images, then the result will be displayed as a "Found".

4.1.3 Eigen Face Approach

It is adequate and efficient method to be used in face recognition due to its simplicity, speed and learning capability. Eigen faces are a set of Eigen vectors 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. The Eigen faces are Principal Components of a distribution of faces, or equivalently, the Eigen vectors of the covariance matrix of the set of the face images, where an image with N by N pixels is considered a point in N^2 dimensional space. Previous work on face recognition ignored the issue of face stimulus, assuming that predefined measurement was relevant and sufficient. This suggests that coding and decoding of face images may give information of face images emphasizing the significance of features. These features may or may not be related to facial features such as eyes, nose, lips and hairs. We want to extract the relevant information in a face image, encode it efficiently and compare one face encoding with a database of faces encoded similarly. A simple approach to extracting the information content in an image of a face is to somehow capture the variation in a collection of face images. We wish to find Principal Components of the distribution of faces or the Eigen vectors of the covariance matrix of the set of face images. Each image location contributes to each Eigen vector, so that we can display the Eigen vector as a sort of face. Each face image can be represented exactly in terms of linear combination of the Eigen faces. The number of 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 which therefore account for most variance between the set of face images. The primary reason for using fewer Eigen faces is computational efficiency.

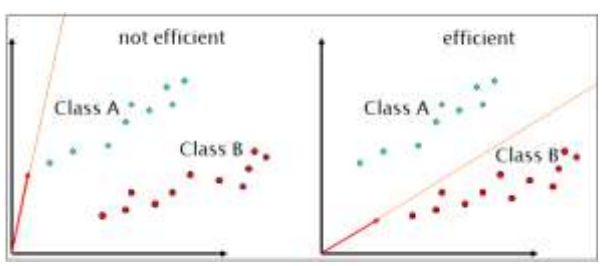

Fig - 4.2: Data representation with help of PCA

The approach to face recognition involves the following initialisation operations:

1. Acquire an initial set of N face images (training images).

2. Calculate the Eigen face from the training set keeping only the M images that correspond to the highest Eigen values. These M images define the “facespace”. As new faces are encountered, the “Eigen faces” can be updated or recalculated accordingly.

3. Calculate the corresponding distribution in M dimensional weight space for each known individual by projecting their face images onto the “face space”.

4. Calculate a set of weights projecting the input image to the M “Eigen faces”.

5. Determine whether the image is a face or not by checking the closeness of the image to the “face space”.

6. If it is close enough, classify, the weight pattern as either a known person i.e. we get result as found in GUI.

7. If it is close enough then the recognition successful and also if sometimes the different face is also shown as a match i.e. wrong person image appears as a output. We call it as a false acceptance by a system.

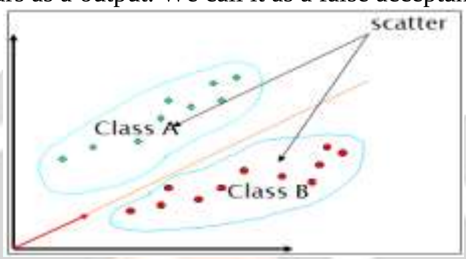

Fig - 4.3: Data classification after applying PCA Algorithm

As shown in fig the total scatter of the data is minimized by Principal Component Analysis.

4.1.4 Mathematical approach

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

$AX = \lambda X,$ (1)

where A is a vector function.

$(A - \lambda I)X = 0,$ (2)

where I is the identity matrix.

This is a homogeneous system of equations and form fundamental linear algebra. We know a non-trivial solution exists if and only if-

$\mathrm{Det}(A - \lambda I) = 0$ (3)

where - det denotes determinant.

When evaluated becomes a polynomial of degree n. This is called characteristic polynomial of A. If A is N by N then there are n solutions or n roots of the characteristic polynomial. Thus there are n Eigen values of A satisfying the equation.

$AX_i = \lambda_i X_i$ (4)

where i = 1,2,3,.....n

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

4.1.5 Face Image Representation

Training set of m images of size N×N are represented by vectors of size N^2. Each face is represented by $\Gamma_1, \Gamma_2, \Gamma_3, \ldots\ \Gamma_m$.

Feature vector of a face is stored in a N×N matrix. Now, this two dimensional vector is changed to one dimensional vector [13].

For Example $\begin{bmatrix}1 & 2\\2 & 1\end{bmatrix} = \begin{bmatrix}1\\2\\2\\1\end{bmatrix}$

Fig - 4.4: Face image conversion into one-dimensional vector

Each face image is represented by the vector Γ_i

$$\Gamma_1=\begin{bmatrix}1\\-2\\1\\-3\end{bmatrix},\ \Gamma_2=\begin{bmatrix}1\\3\\-1\\2\end{bmatrix},\ \Gamma_3=\begin{bmatrix}2\\1\\-2\\3\end{bmatrix},\ldots\ldots\ldots\ldots\ldots\ldots\ \Gamma_m=\begin{bmatrix}1\\2\\2\\1\end{bmatrix} \tag{5}$$

4.1.6 Mean and Mean Centered Images

Average face image is calculated by

$$\mu = \frac{1}{M} \quad \sum_{N-1}^{M} \Gamma_m \tag{6}$$

$$\begin{bmatrix}1\\-2\\1\\-3\end{bmatrix}+\begin{bmatrix}1\\3\\-1\\2\end{bmatrix}+\begin{bmatrix}2\\1\\-2\\3\end{bmatrix}+\ldots\ldots\ldots\ldots\ldots\ldots+\begin{bmatrix}1\\2\\2\\1\end{bmatrix}\rightarrow\begin{bmatrix}-1\\-1\\2\\-3\end{bmatrix}$$

$$\mu=\left(\Gamma 1, + \Gamma 2 + \Gamma 3 + \cdots + \Gamma \text{m}\right) \tag{7}$$

Each face differs from the average by $\emptyset_i = \delta_i - \mu_i$ which is called mean centered image.

$$\phi\, 1=\begin{bmatrix}2\\-1\\-1\\0\end{bmatrix}\ \phi\, 2=\begin{bmatrix}2\\4\\-3\\5\end{bmatrix}\ \phi\, 3=\begin{bmatrix}3\\2\\-4\\6\end{bmatrix}\ldots\ldots\ldots\ldots\ldots\ldots\ \phi\, \text{M}=\begin{bmatrix}2\\3\\0\\4\end{bmatrix}$$

4.1.7 Covariance Matrix

A covariance matrix is constructed as:

$$C = AA^T \tag{8}$$

where A = [ϕ_1, ϕ_2,...... , ϕ_m] of size $N^2 \times N^2$.

$$A=\begin{bmatrix}2 & 3\\-1 & -2\\-1 & 1\\0 & 2\end{bmatrix} \quad A^T=\begin{bmatrix}2 & -1 & -1 & 0\\3 & -2 & 1 & 2\end{bmatrix}$$

Size of covariance matrix will be $N^2 \times N^2$.

Eigen vectors corresponding to this covariance matrix is needed to be calculated, but that will be a tedious task therefore. For simplicity we calculate A^TA which would be a 2 × 2 matrix in this case.

$A^TA = \begin{bmatrix}6 & 7\\7 & 18\end{bmatrix}$ size of this matrix is M×M.

Consider the eigenvectors vectors of A^TA such that

$$A^TA\, X_i = \lambda_i X_i \tag{9}$$

The eigenvectors v_i of A^TA is X_1 and X_2 which are 2 ×1. Now multiplying the above equation with A both sides we get-

$$AA^T AX_i = A\, \lambda_i X_i$$

$AA^T (A X_i) = \lambda_i (A X_i)$ (10)

Eigen vectors corresponding to AA^T can now be easily calculated now with reduced dimensionality where AXi is the Eigen vector and λ_i is the Eigen value.

The Eigen vectors of the covariance matrix AAT are AXi which is denoted by U^i. U^i resembles facial images which look ghostly and are called Eigen faces. Eigen vectors correspond to each Eigen face in the face space and discard the faces for which Eigen values are zero thus reducing the Eigen face space to an extent. The Eigen faces are ranked according to their usefulness in characterizing the variation among the images.

A face image can be projected into this face space by

$$\Omega_k = U^T (\Gamma_k - \mu) \quad (11)$$

k=1,....,M, where $(\Gamma_k - \mu)$ is the mean centered image.

Hence projection of each image can be obtained as Ω_1 for projection of image1 and Ω_2 for projection of image2 and hence forth.

4.1.8 Face Recognition Step

We first train the database of 400 images i.e. 40 different person's images and 10 images per each person. Then apply radial curves on them. We store these features which are extracted through algorithm. Then we select the input image and apply PCA & radial curves on them. We find the Eigen vectors. Then among these Eigen vectors the minimum distance matching with the test image we find match. The face with the minimum Euclidean distance will be our result. Thus we get the matching face not the exact face in PCA method. We have used 10 images of each person so out of 10 any face matching with input will be our output.

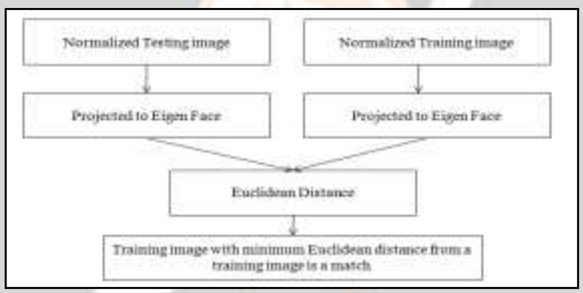

Fig - 4.5: Face recognition process

4.3 Advantages and Limitations of PCA

There are many advantages with the PCA algorithm like accuracy, dimension reduction, simplicity to implementation in coding etc. We have enlisted as below. But as it is slow process this becomes time consuming process.

4.3.1 Advantages of PCA

1. It's the simplest approach which can be used for data compression and face recognition.
2. Operates at a faster rate.

4.3.2 Limitations of PCA

1. Requires full frontal display of faces.
2. Not sensitive to lighting conditions, position of faces.
3. Considers every face in the database as a different image. Faces of the same person are not classified in classes.

4.4 Neural Network Introduction

In the human brain, a typical neuron collects signals from others through a host of fine structured called dendrites. The neuron sends out spikes of electrical activity through a long, thin stand known as an axon, which splits into thousands of branches.

At the end of each branch, a structure called a synapse converts the activity from the axon into electrical effects that inhibit or excite activity from the axon into electrical effects that inhibit or excite activity in the connected neurones. When a neuron receives excitatory input that is sufficiently large compared with its inhibitory input, it sends a spike of electrical activity down its axon. Learning occurs by changing the effectiveness of the synapses so that the influence of one neuron on another changes.

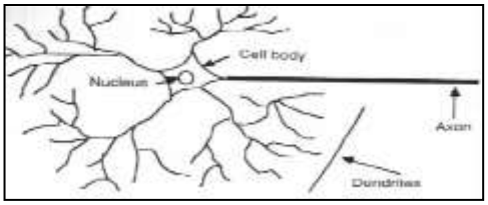

Fig - 4.6: Components of a neuron

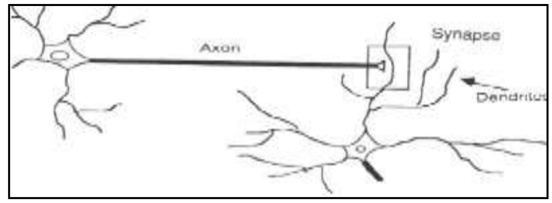

Fig - 4.7: The synapse

An artificial neuron is a device with many inputs and one output. The neuron has two modes of operation; the training mode and the using mode. In the training mode, the neuron can be trained to fire (or not), for particular input patterns. In the using mode, when a taught input pattern is detected at the input, its associated output becomes the current output. If the input pattern does not belong in the taught list of input patterns, the firing rule is used to determine whether to fire or not.

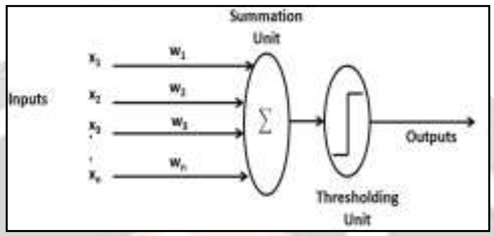

Fig - 4.8: A simple model of Artificial Neuron

Artificial neural networks are a popular tool in face recognition. They have been used in pattern recognition and classification. Kohonen [8] was the first to demonstrate that a neuron network could be used to recognize aligned and normalized faces. Many different methods based on neural network have been proposed since then. Some of these methods use neural networks just for classification. One approach is to use decision-based neural networks, which classifies pre-processed and sub sampled face images. There are methods which perform feature extraction using neural networks. For example, Intrator et. al proposed a hybrid or semi-supervised method. They combined unsupervised methods for extracting features and supervised methods for finding features able to reduce classification error. They used feed-forward neural networks (FFNN) for classification. They also tested their algorithm using additional bias constraints, obtaining better results. They also demonstrated that they could decrease the error rate training several neural networks and averaging over their outputs, although it is more time-consuming that the simple method.

4.5 Neural Network Architectures

An artificial neural network is defined as a data processing system consisting of large no of simple highly interconnected processing elements (artificial neurons) in an architecture by the structure of the cerebral cortex of the brain. Generally, an ANN structure can be represented using a directed graph. A graph G is an ordered 2-tupe l(V, E) consisting of a set V of vertices and a set of E of edges. When each edge is assigned an orientation the graph is directed and is called graph or a diagraph.

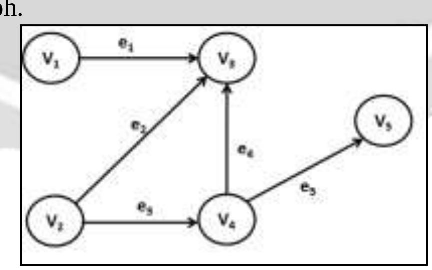

Fig - 4.9: An example of digraph

There are several classes of NN, classified according to their learning mechanisms. However, we identify three fundamentally different classes of network s. All these classes employ diagraph structure for their representation.

4.5.1 Single Layer Feed Forward Network

This type of network comprises of two layers, namely the input layer and the output layer. The input layer neurons receive the input signal and the output layer neurons receive the output signals. The synaptic links carrying the weights connect every input neuron to output neuron but not vice versa. Such a network is said to be a feed forward in type or acyclic in nature. Despite the two layers, the network is termed as single layer since it is the output layer, alone which performs computation. The input layer merely transmits the signals to output layer. Hence, the name single layer feed forward network. Fig.4.10 Below illustrates an example of this network.

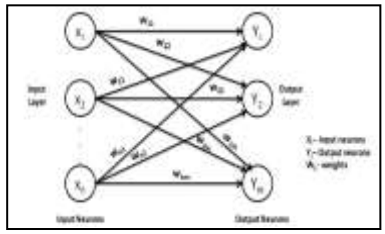

Fig - 4.10: Single layer feed forward network

4.5.2 Multilayer Feed Forward Network

This network as its name suggests is made up of multiple layers. Thus, architecture of this class besides possessing an input and an output layer also have one or more layer called as hidden layers. The computational units of hidden layers are called as the hidden neurons or hidden units. The hidden layers help in performing the useful intermediary computations before directing the input to output layer. The input layer neurons are linked to hidden neurons and weights on these links are referred to as the input- hidden layer weights. Again the hidden layer neurons are linked to the output layer neurons and weights on these links are known as the hidden-output layer weights.

Here x_i= input to the neural network at input layer

w_i= weights of the neural network

y_i= output of neural network

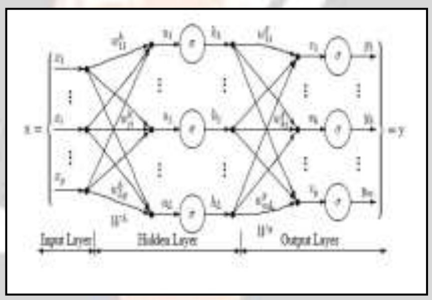

Fig - 4.11: Multilayer neural network

4.6 Types of learning methods in Neural Networks

Face recognition is nothing but the ability of machine to successfully categorize a set of images based on certain discriminatory features. Classification or pattern recognition can be a very difficult problem and still a very active field of research. The aim of this project is to automatically recognize the input image entered with the existing database. Classification by machine requires two steps: First, the properties the properties that distinguish the element from the class have to be identified. Secondly, the machine has to be trained to know how to discriminate between classes by defining a learning model. The learning model describes the procedure that has to be used for the actual training. Basically there are two types of learning supervised learning and unsupervised learning.

4.6.1 Supervised Learning

In Supervised learning, the trainer feeds the machine a sample and the machine classifies it. The trainer then tells the machine whether it has classified it correctly or not. In the case that it has misclassified the sample, the machine has to adjust the classifier parameters to better classify the given sample or an input sample. Supervised learning is illustrated Fig 4.12. Examples of this include Bayes classifier and Neural Networks. In supervised training, both the inputs and the outputs are provided. The network then processes the inputs and compares its resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights which control the network. This process occurs over and over as the weights are continually tweaked. The set of data which enables the training is called the training set. During the training of a network the same set of data is processed many times as the connection weights are ever refined.

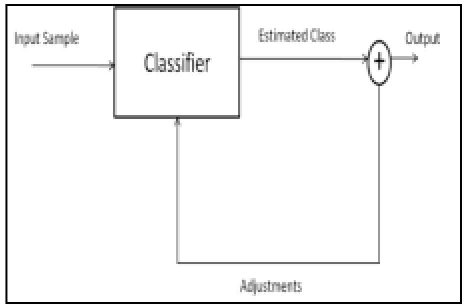

Fig - 4.12: Supervised Learning

4.6.1.1 Advantages

1. Analyst has control over the selected classes tailored to the purpose
2. Have specific classes of known identity
3. Can detect serious errors in classification if training areas are misclassified

4.6.2 Unsupervised Learning

In unsupervised or self-organized learning there is no external trainer to observe the learning process. In unsupervised learning the net is autonomous: it just looks at the data it is presented with, finds out about some of the properties of the data set and earns to reflect these properties in its output. What exactly these properties are, that the network can learn to recognise, depends on the particular network model and learning method. In unsupervised training, the network is provided with inputs but not with desired outputs. The system itself must then decide what features it will use to group the input data. This is often referred to as self-organization or adaption.

4.6.2.1 Advantages

1. Requires no prior knowledge of the region.
2. Human error is minimized.
3. Unique classes are recognized as distinct units.

4.6.2.2 Disadvantages

1. Classes do not necessarily match informational categories of interest.
2. Limited control of classes and identities.

4.6.3 Reinforcement Learning

In reinforcement learning, data x is usually not given, but generated by an agent's interactions with the environment. At each point in time t, the agent performs an action y_t and the environment generates an observation x and an instantaneous cost c_t according to some (usually unknown) dynamics. The aim is to discover a policy for selecting actions that minimizes some measure of a long-term cost, e.g., the expected cumulative cost. The environment's dynamics and the long-term cost for each policy are usually unknown, but can be estimated.

The neural network classifier is a more advanced alternative to the Euclidian distance method. The primary advantage of the neural network is that no assumptions need to be made as to how clustering will occur in face-space. Given the correct network configuration, situations like the tongue out versus normal pose problem should be handled automatically by the system, with no additional design consideration.

It should be noted that this network configuration is designed to accept the weight values that are obtained by projecting a test face into face-space. This projection is accomplished in the same way as described in the Eigenface section, and is the same way used in the Euclidian distance classification method. Other possible network configurations include an additional layer, which takes the place of the projection into face-space. In this configuration, the projection is learned along with the rest of the classification. While it is possible that this setup might allow the system to degrade more gracefully with error, we have not implemented it for our experiments because of the time required to train the network. With the inputs to the network being the individual pixels of the image, we would expect significantly longer training time than for a network accepting the weights as data, which was already a significant amount of time.

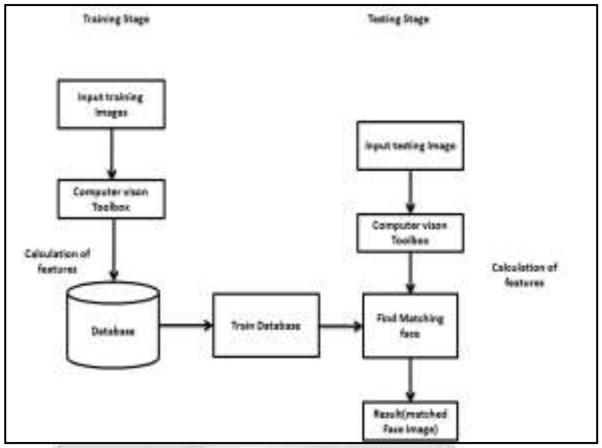

Fig - 4.13: Block diagram of neural network FR system

4.7 Algorithm for Neural Network Face recognition System

1) Training

1. Select folder of database of training images.
2. Apply computer vision toolbox on each image to detect eye pair, nose, mouth, etc.
3. Calculate features of all images.
4. Store it into database.

2) Testing

1. Load database.
2. Select input image.
3. Use computer vision toolbox for calculating eye pair, nose and mouth regions.
4. Calculate the features.
5. Train the database into neural network using BPNN to get the network.
6. Use that network and input feature to calculate similar image to find the match.

4.8 Viola Jones Algorithm for cascade vision detection process

The main characteristics of Viola–Jones algorithm that makes it a good detection algorithm are:

1. Robust – very high detection rate (true-positive rate) & very low false-positive rate always.
2. Real time – For practical applications at least 2 frames per second must be processed.
3. Face detection and not recognition - The goal is to distinguish faces from non-faces (face detection is the first step in the identification process)

The algorithm has mainly 4 stages:

1. Haar Features Selection
2. Creating Integral Image
3. Adaboost Training algorithm
4. Cascaded Classifiers

The features employed by the detection framework universally involve the sums of image pixels within rectangular areas. As such, they bear some resemblance to Haar basis functions, which have been used previously in the realm of image-based object detection. However, since the features used by Viola and Jones all rely on more than one rectangular area, they are generally more complex. The figure at right illustrates the four different types of features used in the framework. The value of any given feature is always simply the sum of the pixels within clear rectangles subtracted from the sum of the pixels within shaded rectangles. As is to be expected, rectangular features of this sort are rather primitive when compared to alternatives such as steerable filters. Although they are sensitive to vertical and horizontal features, their feedback is considerably coarser.

Fig - 4.14: 1st and 2nd Haar features example

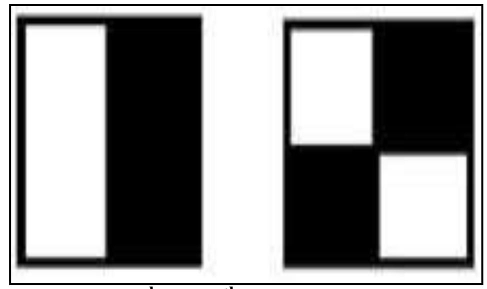

Fig - 4.15: 3rd and 4th kind of Haar Feature

Haar Feature that looks similar to the bridge of the nose is applied onto the face. Haar Feature that looks similar to the eye region that is darker than the upper cheeks is applied onto a face.

Haar Features – All human faces share some similar properties. This knowledge is used to construct certain features known as Haar Features. The properties that are similar for a human face are:

1. The eyes region is darker than the upper-cheeks.
2. The nose bridge region is brighter than the eyes.

That is useful domain knowledge:

1. Location - Size: eyes & nose bridge region
2. Value: darker / brighter

The four features applied in this algorithm are applied onto a face and shown on the left.

Rectangle features:

$$\text{Value} = \Sigma \text{ (pixels in black area)} - \Sigma \text{ (pixels in white area)} \tag{12}$$

Three types of features are there two-, three-, four-rectangles; Viola & Jones used two-rectangle features. For example, the difference in brightness between the white & black rectangles over a specific area. Each feature is related to a special location in the sub-window However, with the use of an image representation called the integral image, rectangular features can be evaluated in constant time, which gives them a considerable speed advantage over their more sophisticated relatives. Because each rectangular area in a feature is always adjacent to at least one other rectangle, it follows that any two-rectangle feature can be computed in six array references, any three-rectangle feature in eight, and any four-rectangle feature in just ten. The integral image at location (x, y), is the sum of the pixels above and to the left of (x, y), inclusive.

4.9 Back Propagation Neural Network

The Back propagation neural network is a multi-layered, feed forward neural network and is by far the most extensively used. It is also considered one of the simplest and most general methods used for supervised training of multi-layered neural networks. Back propagation works by approximating the non-linear relationship between the input and the output by adjusting the weight values internally. It can further be generalized for the input that is not included in the training patterns (predictive abilities).

Generally, the Back propagation network has two stages, training and testing. During the training phase, the network is "shown" sample inputs and the correct classifications. For example, the input might be an encoded picture of a face, and the output could be represented by a code that corresponds to the name of the person.

A further note on encoding information - a neural network, as most learning algorithms, needs to have the inputs and outputs encoded according to an arbitrary user defined scheme. The scheme will define the network architecture so that once a network is trained; the scheme cannot be changed without creating a very new net. Similarly, there are many forms of encoding the network response.

The following figure shows the topology of the Back propagation neural network that includes and input layer, one hidden layer and an output layer. It should be noted that Back propagation neural networks can have more than one hidden layer.

Once the error signal for each node has been determined, the nodes to update the values for each connection weights until the network converges to a state that allows all the training patterns to be encoded then use the errors. The Back propagation algorithm looks for the minimum value of the error function in weight space using a technique called the delta rule or gradient descent. The weights that minimize the error function is then considered to be a solution to the learning problem.

The network behaviour is analogous to a human that is shown a set of data and is asked to classify them into predefined classes. Like a human, it will come up with "theories" about how the samples fit into the classes. These are then tested against the correct outputs to see how accurate the guesses of the network are. Radical changes in the latest theory are indicated by large changes in the weights, and small changes may be seen as minor adjustments to the theory.

There are also issues regarding generalizing a neural network. Issues to consider are problems associated with under-training and over-training data. Under-training can occur when the neural network is not complex enough to detect a pattern in a complicated data set. This is usually the result of networks with so few hidden nodes that it cannot accurately represent the solution, therefore under-fitting the data.

On the other hand, over-training can result in a network that is too complex, resulting in predictions that are far beyond the range of the training data. Networks with too many hidden nodes will tend to over-fit the solution. The aim is to create a neural network with the "right" number of hidden nodes that will lead to a good solution to the problem.

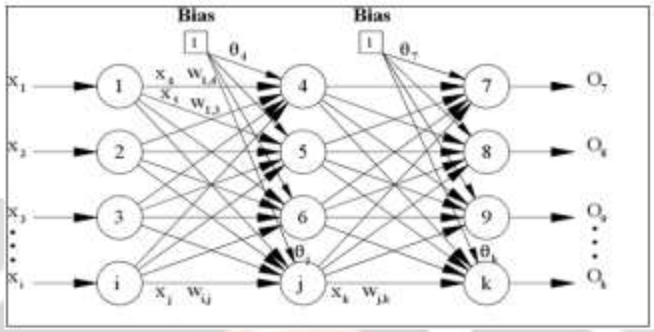

Fig - 4.16: Back propagation Neural Network with one hidden layer

4.10 Back Propagation Algorithm

Using fig.4.13, the following describes the learning algorithm and the equations used to train a neural network. The back propagation neural network has two stages first is feed forward and second is feedback. In feedback stage the error is provided back to hidden layer and it is adjusted to get the best output.

4.10.1 Feed forward Network

When a specified training pattern is fed to the input layer, the weighted sum of the input to the j^{th} node in the hidden layer is given by

$$\text{Net}_j = \sum w_{ij}\, x_j + \theta_j \qquad (12)$$

Equation (12) is used to calculate the aggregate input to the neuron. The term θ_j is the weighted value from a bias node that always has an output value of 1. The bias node is considered a "pseudo input" to each neuron in the hidden layer and the output layer, and is used to overcome the problems associated with situations where the values of an input pattern are zero. If any input pattern has zero values, the neural network could not be trained without a bias node.

To decide whether a neuron should fire, the "Net" term, also known as the action potential, is passed onto an appropriate activation function. The resulting value from the activation function determines the neuron's output, and becomes the input value for the neurons in the next layer connected to it.

Since one of the requirements for the Back propagation algorithm is that the activation function is differentiable, a typical activation function used is the Sigmoid

$$O_j = x_k = \frac{1}{1+e^{-Net_j}} \qquad (13)$$

It should be noted that many other types of functions can, and are, used hyperbolic tan being another popular choice. Similarly, equations (17) and (18) are used to determine the output value for node k in the output layer.

4.10.2 Error Calculations and Weight Adjustments

If the actual activation value of the output node, k, is O_k, and the expected target output for node k is t_k, the difference between the actual output and the expected output is given by:

$$\Delta_k = t_k - O_k \qquad (14)$$

Where t_k= expected target output

O_k= actual activation value of the output node or obtained value of output

The error signal for node k in the output layer can be calculated as

$$\delta_k = \Delta_k O_k (1 - O_k) \qquad (15)$$

where the $O_k(1-O_k)$ term is the derivative of the Sigmoid function.

With the delta rule, the change in the weight connecting input node j and output node k is proportional to the error at node k multiplied by the activation of node j.

The formulas used to modify the weight, $w_{j,k}$, between the output node k and the node j is

$w_{j,k} = w_{j,k} + \Delta\, w_{j,k}$ (16)

where $\Delta\, w_{j,k}$ = the change in the weight between nodes j and k,

l_r= the learning rate.

The learning rate is a relatively small constant that indicates the relative change in weights. If the learning rate is too low, the network will learn very slowly, and if the learning rate is too high, the network may oscillate around minimum point, overshooting the lowest point with each weight adjustment, but never actually reaching it. Usually the learning rate is very small, with 0.01 not an uncommon number. Some modifications to the Back propagation algorithm allow the learning rate to decrease from a large value during the learning process. This has many advantages. Since it is assumed that the network initiates at a state that is distant from the optimal set of weights, training will initially be rapid. As learning progresses, the learning rate decreases as it approaches the optimal point in the minima. Slowing the learning process near the optimal point encourages the network to converge to a solution while reducing the possibility of overshooting. If, however, the learning process initiates close to the optimal point, the system may initially oscillate, but this effect is reduced with time as the learning rate decreases.

It should also be noted that, in equation (17), the x_k variable is the input value to the node k, and is the same value as the output from node j. To improve the process of updating the weights, a modification to equation (16) is made:

$\Delta w_{j,k}^{n} = l_r\, \delta_k x_k + \Delta w_{j,k}^{(n-1)} \mu$ (17)

Here the weight update during the nth iteration is determined by including a momentum term (μ), which is multiplied to the (n-1)th iteration of the $\Delta w_{j,k}^{n}$.

The introduction of the momentum term is used to accelerate the learning process by "encouraging" the weight changes to continue in the same direction with larger steps. Furthermore, the momentum term prevents the learning process from settling in a local minimum. by "over stepping" the small "hill". Typically, the momentum term has term has a value between 0 and 1.

Advantages of Neural Network over PCA method

1. In NN training requires time, but less than PCA
2. Results are faster in NN.
3. Accuracy of NN is better than PCA.
4. False Rejection Ratio is more in NN as compared to PCA.

5. EXPERIMENTATION AND RESULTS

5.1 The Database

As mentioned above in the previous section, input images are preprocessed and the features extracted to remove redundant information and reduce the complexity of networks structure and also improve the efficiency of training networks. Interpolation algorithm are applied to face image compression and the grey value of compressed image is normalized to [0,1]. In this experiment, the Oracle Research Laboratory database (ORL database) of faces is used. The database was used in the context of a face recognition project carried out in collaboration with the Speech, Vision and Robotics Group of the Cambridge University Engineering Department [11]. There are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The files are in PGM format, and size of each image is 92x112 pixels, with 256 grey levels per pixel. The images are organized in 40 directories (one for each subject).

In each of these directories, there are ten different images of that subject, which have names of the form Y.pgm, where Y is the image number for that subject (between 1 and 10). The database can be retrieved from:

http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.zip

Some example images from the ORL database is shown in the following fig 5.1:

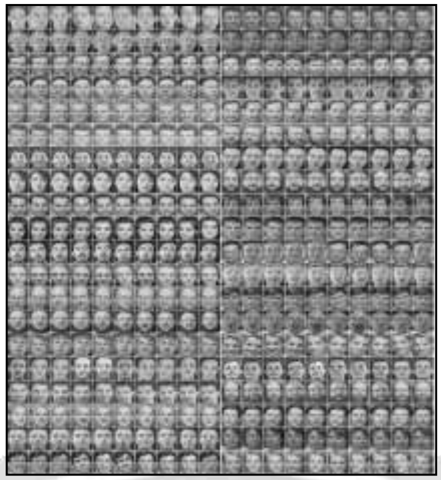

Fig - 5.1: The ORL Database (Oracle Research Laboratory database)

5.2 Normalized images

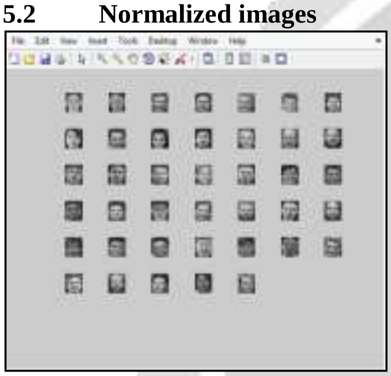

Fig - 5.2: Normalized Images

5.3 Mean Image

Fig - 5.3: Mean Image of all the face images

5.4 Eigen Faces

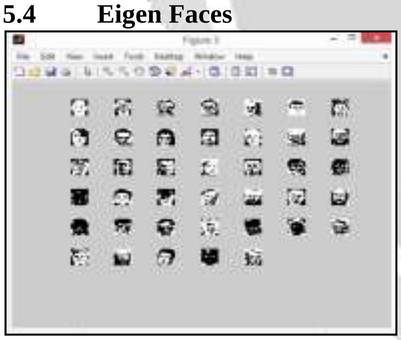

Fig - 5.4: Eigen faces

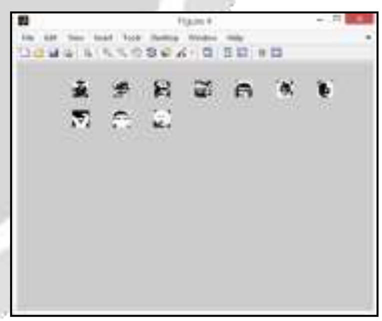

Fig - 5.5: Eigen faces

5.5 Results of PCA Method

Fig - 5.6: GUI for PCA Method

Fig - 5.7: PCA method – checking face for matching with input

Fig - 5.8: PCA method – Match found

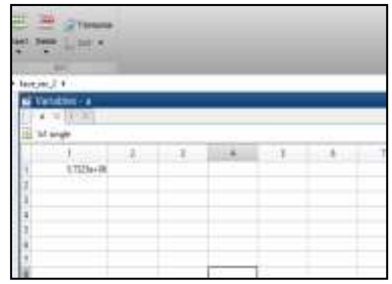

Fig - 5.9: Matrix with minimum distance

Figures 5.7 and 5.8 show the output of the PCA with radial curves method for face recognition. In this we create the database first by selecting the folder which contains total 400 no of images which is comprised of the 10 images of 40 persons.

After selecting the database folder and creating the database with the help of GUI we need to select the input image folder i.e. the image to be searched for recognition. Then we will run our PCA algorithm and will find the result. The PCA algorithm will check all the 400 images one by one and this is time consuming process as compared to the neural network algorithm proposed in this paper. In neural network time required is very less and due to this we can suggest this system to be used at the real time environment such as airport or day-care centres, office premise or the bank ATM centres. We can see in figure 5.7 that the face is being checked during the testing phase of our PCA and RBC algorithm. When the checking has been completed by our system the result will be shown as in figure 5.8 as "Found". This completes the procedure of the recognition of an input image.

5.6 Results OF NN Method

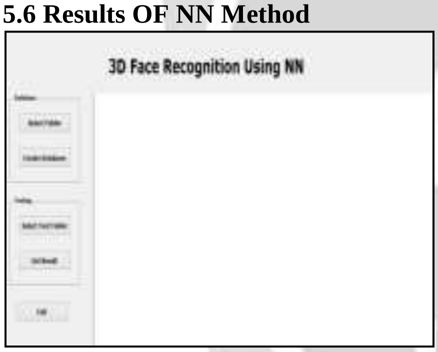

Fig - 5.10: GUI for NN Method

Fig - 5.11: NN method – checking the face for matching with input image

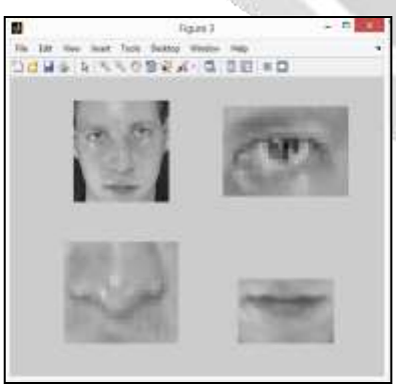

Fig - 5.12: Features extraction

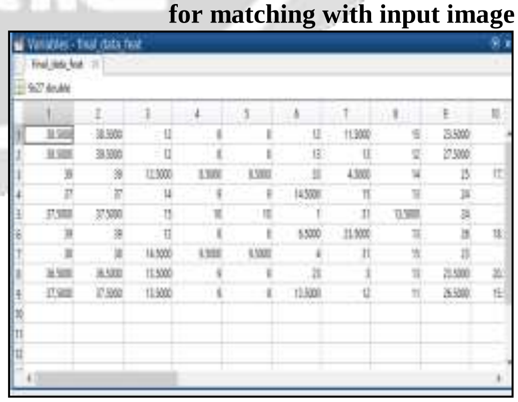

Fig - 5.13: Features of matching image

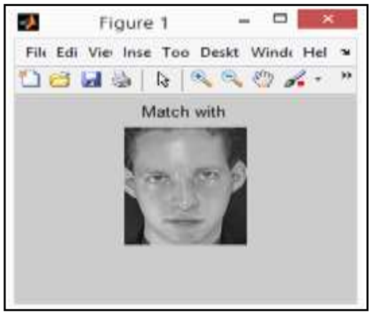

Fig - 5.14: NN method - Output of matched face

Here we have compared the acceptance ratio and rejection ratios of both the algorithms for the analysis of PCA-RBC and NN using BPNN. PCA method is giving wrong match for certain face images. We call the match of wrong output image with the input image as the false acceptance by the face recognition system.

Fig - 5.15: False Acceptance

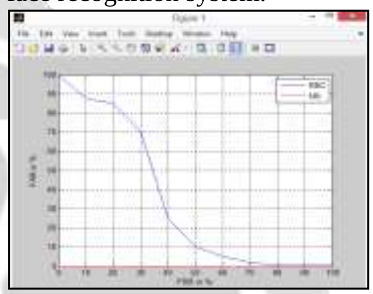

Fig - 5.16: Graph of FRR and FAR for RBC and NN

Figure 5.16 is the graph of False Acceptance Ratio (FAR) and False Rejection Ratio (FRR). When the no of images in dataset are less the false acceptance ratio is high and false rejection ratio is low. Whereas when the no of images in database are increased gradually the false acceptance ratio decreases. Also PCA with RBC method has the higher FAR than NN for less than 40% and Neural Network show good performance for almost all the images. So we can state that the NN has better results for the face recognition system. But one interesting thing is that the PCA system can match with the similar face not the same exact image of that person in database. NN matches the exact face image in database. As shown in figure below the graph of accuracy is plotted against the no of images used for training the database. We can see that he accuracy of PCA is good and it decreases slightly when no of images are increased above 100.

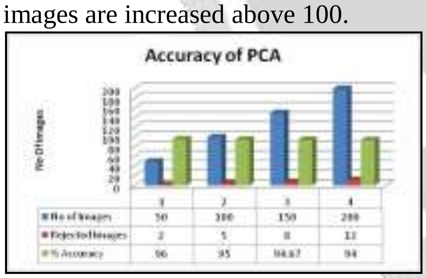

Fig - 5.17: Graph of Accuracy of PCA Algorithm

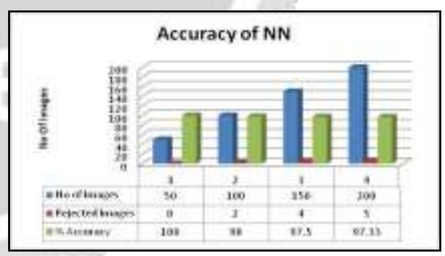

Fig - 5.18: Graph of Accuracy Of NN

From the graph in fig 5.17 it can be easily noted that when the database images are less the performance of face recognition system is very good. Also the accuracy is better it is 96% for 50 images. As no of images goes on increasing the chance to getting incorrect result also increases. And the accuracy of our PCA based face recognition system starts decreasing. For 100 images, it is 96%, for 150 images, it is 94.67% and for 200 images, it is 94%.

As we can see in the graph fig 5.18, accuracy of NN is very good as compared to the PCA method; NN gives accuracy of 97.33% when tested on 200 images.

6. CONCLUSION AND FUTURE SCOPE

In this work we have implemented a system for face detection using the PCA algorithm and radial curves which were used for the better and accurate classification of features. In this method we first input the testing image and match with the training images with help of PCA algorithm and radial curves. This method gives accuracy of 96% when tested on 50 training images and 94.67% when tested on 200 training images. We have used the PCA for the feature selection as it considers all the selective face features for checking the image. Also with use of radial curves it will take the best features for the mapping of input image with the training images and speed will be increased as instead of whole face only best features are selected.

We have calculated the important features such as Normalized images, Eigen faces and mean image. Based on these features we have calculated the minimum Euclidean distance between the test image and training image.

In the second method we have used the vision. Cascade detect process based on Viola –Jones Algorithm. This we have used to automatically detect the eye pair, nose and mouth as these are most distinct and unique features of a human face. Then these features of test image are matched with the training set face images. Here we have used Back propagation as a classifier of the generated data of features. We have used BPNN as it is a unsupervised method of learning algorithm in neural networks. It requires defining the target vector of output. It uses the features of input matrix as a target vector as a measure to calculate the matching scores with the training images. More is matching score with predefined target the more are chances to face to be recognized as known face from the database. This method gives very good results than PCA-RBC method. Time required to give the result is less in NN method as compared to the PCA- RBC method. Also NN has less false acceptance rate. So it is more reliable than PCA-RBC method. The algorithm takes into account the front images for the feature selection and classification. We can further extend this algorithm for the face detection in presences of occlusions like sunglasses, hair.

Also this algorithm can be used for the office attendance systems when the office database is stored already in a system. We can use this system for verification of person's identity at immigration check at airport. This includes one to one matching process; the person's passport photo will be checked with real time image of that persons.

REFERENCES

[1] Anil K. Jain, Robert P.W. Duin, and Jianchang Mao. Statistical Pattern Recognition: AReview. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1):4 37, January 2000.

[2] T. Kanade, "Picture Processing System by Computer Complex and Recognition of Human Faces," Kyoto University, Japan, PhD. Thesis 1973.

[3] Chaua˜ C. Queirolo, Luciano Silva, Member, Olga R.P. Bellon, and Maurı´cio Pamplona Segundo "3D Face Recognition Using Simulated Annealing and the Surface Interpenetration Measure" IEEE Transactions On Pattern Analysis And Machine Intelligence, VOL. 32, NO. 2, Feb 2010

[4] I.A.I. Kakadiaris, G. Passalis, G. Toderici, M.N.M. Murtuza, Y. Lu,N. Karampatziakis, and T. Theoharis, "Three-Dimensional Face Recognition in the Presence of Facial Expressions: An Annotated Deformable Model Approach," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 29, no. 4, pp. 640-649, Apr. 2007.

[5] Tomesh Verma, Raj Kumar Sahu "PCA-LDA Based Face Recognition System & Results Comparison By Various Classification Techniques" Proceedings of 2013 International Conference on Green High Performance Computing March 14-15, 2013, India 978-1-4673-2594-3/13/$31.00 ©2013 IEEE

[6] Yee Wan Wong, Kah Phooi Seng, and Li-Minn Ang "Radial Basis Function Neural Network With Incremental Learning for Face Recognition" IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, VOL. 41, NO. 4, AUG 2011

[7] A. K. Jain, R. Bolle, and S. Pankanti, "Biometrics: Personal Identification in Networked Security," A. K. Jain, R. Bolle, and S. Pankanti, Eds.: Kluwer Academic Publishers, 1999.

[8] M.Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Congnitive Neuroscience, 3(1), pp.71-86, 1991.

[9] S. Kung and J. Taur. "Decision-based neural networks with signal/image classification applications" IEEE Transactions on Neural Networks, 6(1):170–181, 1995.

[10] N. Intrator, D. Reisfeld, and Y. Yeshurun. Face recognition using a hybrid supervised/unsupervised neural network. Pattern Recognition Letters, 17:67–76, 1995.

[11] Abhishek Singh Face Recognition Using PCA and Eigen Face Approach Department of Computer Science and Engineering National Institute of Technology Rourkela

[12] Tan, X.; Chen, S.; Zhou, Z. & Zhang, F. (2005). Recognizing Partially Occluded, Expression Variant Faces from Single Training Image per Person with SOM and Soft k-NN Ensemble, IEEE Trans. on Neural Networks, Vol. 16, No. 4, pp. 875-886

[13] Martinez, A. M. (2002). Recognizing Imprecisely Localized, Partially Occluded, and Expression Variant Faces from a Single Sample per Class, IEEE Trans. on Pattern Analysis Machine Intelligence, Vol. 25, No. 6, pp. 748-763

[14] SIMON HYKIN, “Neural Networks”,Second Edition, A comprehensive Foundation, Prentice Hall of India Private Limited.

[15] S. Rajasekaran, G. A. Vijayalakshmi Pai “Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications” PHI publications.