Face Recognition System Using Principle Component Analysis

Project Report submitted in partial fulfillment of the

degree of

Bachelor of Technology.

in

Electronics and Communication Engineering

May 2014

under the Supervision of

Mr. Kaushlendra Kumar Pandey

By

DIVYAN RAWAL(101074)

SHIVA AGGARWAL(101092)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
WAKANAGHAT

CERTIFICATE

This is to certify that project report entitled **" Face Recognition System Using principle Component Analysis "** submitted by **Divyan Rawal (101074)** and **Shiva Aggarwal (101092)** in partial fulfillment for the award of degree of **Bachelor of Technology** of Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Signature of Guide:

Name of Guide: **Mr. Kaushlendra Kumar Pandey**

Designation: **Assistant Professor (Grade-II)**

Date: **24th May 2014**

ACKNOWLEDGEMENT

The project in this report is an outcome of continual work over the period of one year & intellectual support from various sources. It involves a lot of direct and indirect contribution from a lot of people. It is difficult to express adequately the debts owed to all of them who have been instrumental in imparting this work a successful status.

We are extremely indebted to our esteemed mentor and supervisor **Prof. Kaushlendra Kumar Pandey** who was always there to tell us how to go about our project in a systematic manner and who always took out time to help us with our technical and non-technical doubts at various stages of the project.

Equally important was the contribution of Mr. Mohan , our Lab Assistant, who kept faith in our ability to complete the project well and on time.
We would also like to express our thanks to all the staff in general who cooperated with us in this project. And last but not the least we would like to thank our friends to help us at various times of our difficulties .

Date: 24th May 2014

Divyan Rawal (101074)
Shiva Aggarwal (101092)

TABLE OF CONTENTS

Chapter No.	Topics	Page No
	Summary	5
	List Of Figures	6
Chapter 1.)	Introduction	7-9
	1.1 Challanges	10
	1.2 Problem Statement	11
	1.3 Motivation	12
Chapter 2.)	Literature Review	13-28
	2.1 Face Recognition	14-15
	2.2 MATLAB Overview	16-19
	2.3 Different type of Approaches for Face Recognition	20-21
	2.4 PCA Approach for Face Recognition	22-25
	2.5 Eigen Values And Eigen Faces	26-28
Chapter 3.)	Problem Solution And Work Done	29-48
	3.1 Block Diagram	30
	3.2 Algorithm	31
	3.3 MATLAB Codes	32-46
	• Test Image	32-35
	• Training Database	36-38
	• Face Recognition Technique	39-46
Chapter 4.)	Conclusion And Future Scope	47
Chapter 5.)	References	48

SUMMARY

Security and authentication of a person is a crucial part of any industry. There are many techniques used for this purpose. One of them is face recognition. Face recognition is an effective means of authenticating a person. A **facial recognition system** is a computer application for automatically identifying or verifying a person from a digital image or a video frame from a video source. A face recognition system has to associate an identity or name for each face it comes across by matching it to a large database of individuals. One of the ways to do this is by comparing selected facial features from the image and a facial database. The advantage of this approach is that, it enables us to detect changes in the face pattern of an individual to an appreciable extent. The recognition system can tolerate local variations in the face expression of an individual. Hence face recognition can be used as a key factor in crime detection mainly to identify criminals. There are several approaches to face recognition of which Principal Component Analysis (PCA) have been incorporated in our project. The system consists of a database of a set of facial patterns for each individual. The characteristic features called ‘eigenfaces’ are extracted from the stored images using which the system is trained for subsequent recognition of new images.

Divyan Rawal
Shiva Aggarwal

Kaushlendra Kumar Pandey
Project Supervisor

List of Figures

1. Figure 1 Generic representation of face recognition system

2. Figure 2 RGB image components

3. Figure 3 Block Diagram of PCA based Face Recognition

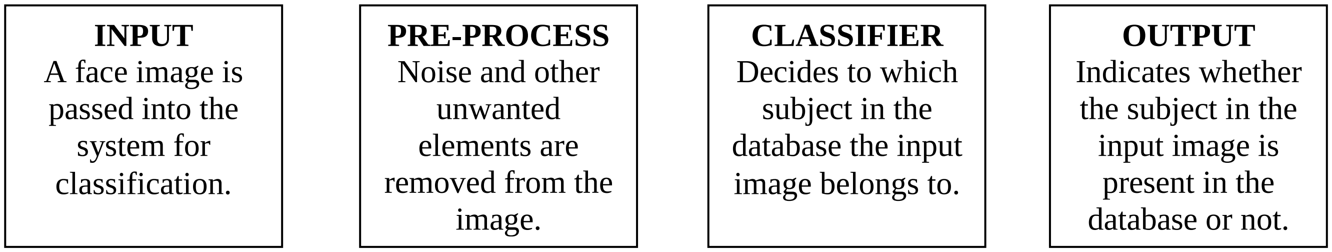
4. Figure 4 Algorithm Chart

Chapter :1

INTRODUCTION

Human face recognition systems have gained a considerable attention during last decade due to its vast applications in the field of computer and advantages over previous biometric methods. There are many applications with respect to security, sensitivity and secrecy. Human face detection suffers from various challenges due to variation regarding image conditions, size, resolution, poses and rotation. Its accurate and robust detection has been a great task for the researchers. There exist various numbers of methods and techniques for face detection but none can guarantee successful in all conditions for all kinds of faces and images. Some methods are exhibiting good results in certain conditions and others are good with different kinds of images.

Face Recognition has become a very active area of research in recent years mainly due to security demands and its potential commercial and law enforcement applications. The last decade has shown dramatic progress in this area, with emphasis on such applications as human-computer interaction (HCI) , biometric analysis, content-based coding of images and videos, and surveillance . Computational model of face recognition are interesting because they can contribute not only the theoretical insights but also to practical applications. Computer that recognize face could be apply to a lot of wide variety of problems, including criminal identification, security systems, image and film processing, and human computer interaction. For example, to be able to model particular face distinguish it from a large number of stored models; this makes it possible to improve criminal identification. Although a trivial task for the human brain , face recognition has proved to be extremely difficult to imitate artificially, since although commonalities do exist between faces, they vary considerably in terms of age, skin color and gender. The problem is further complicated by differing image qualities, facial expression, facial features, background, and illumination conditions. A generic representation of Face Recognition system is shown in Fig. 1.



INPUT
A face image is passed into the system for classification.

PRE-PROCESS
Noise and other unwanted elements are removed from the image.

CLASSIFIER
Decides to which subject in the database the input image belongs to.

OUTPUT
Indicates whether the subject in the input image is present in the database or not.

Fig.1) Generic representation of face recognition system

In our Project , we propose a new method for face recognition using PCA. The face features are extracted by the PCA method, reducing the dimensionality of input space. It has been seen that variations in pose, orientation, etc. are quite high. Principle component analysis aims to catch the total variation in the set of the training faces, and to explain the variation by a few variables. In fact, observation described by a few variables is easier to understand than it was defined by a huge amount of variables and when many faces have to be recognized the dimensionality reduction is important .Therefore, to achieve high recognition rate, structural information of face images of the same subjects is considered for classification process.

The scheme is based on an information theory approach that decomposes face images into a small set of characteristic feature images called 'eigenfaces', which are actually the principal components of the initial training set of face images. Recognition is performed by projecting a new image into the subspace spanned by the eigenfaces ('face space') and then classifying the face by comparing its position in the face space with the positions of the known individuals.

1.1

CHALLANGES

Face detection method faces are various challenges such as pose, imaging condition.

Pose: The image of a face vary due to relative camera-face pose and some facial feature becomes partially or completely occluded.

Image orientation: face images directly vary for different rotation about the camera's optical axis. Image orientation directly affects the angle of the face

Illumination: This problem is mainly due to lighting, makes a larger difference with the same face as compare to difference within different faces while comparing .

Occlusion: Sometime, faces in the image are occluded with other objects such as moustaches, beard, optical lenses and other types of object make it very difficult to find accurate image.

Facial expression: Some persons may have different expression at different times, this also contributes to challenge in face detection.

Face size: Size of the face also make difficult to automate a system for face detection and recognition .

Image condition: This problem includes factors such as intensity, resolution, camera lighting, background, characteristics of image capturing device and distance between camera and person, plays an important role in the process of face detection and recognition.

1.2

Problem Statement

Face recognition is important for the interpretation of facial expressions in applications such as intelligent man-machine interface and communication, intelligent visual surveillance, teleconference and real-time animation from live motion images. There have been many computer models proposed for machine-based recognition of face images. Among them, the eigenface approach, initially introduced for face image coding and then for classification, extracts principal local and global hidden “face stimuli” that are significant in recognition. The distinctive features of this method are: the eigenvectors reflect the statistical properties of face images they represent; they capture more global “signatures” of the faces and therefore, more tolerant and immune to local variations. Because face recognition is commonly subject to a wide range of changes in viewing angle and facial hair as well as to partial occlusion and blurring, the eigenface method is computationally more robust and biologically more plausible than other template matching techniques that are based on the detection of visible local facial features and the representation of face models by geometric measures of such features.

Eigenfaces also provide an attractive mechanism for the transmission of coded image sequences through networks. However, there is a fundamental problem with the existing eigenface models. Current eigenface models only recognize single face images that have been taken from a strict narrow view angle, most commonly a frontal view.

1.3

Motivation

This project has been developed to get people interest in computer vision and face recognition. Through the development of techniques like Principle Component Analysis computers can now outperform humans in many face recognition task, particularly those in which large database of faces must be searched.

Face recognition has recently received a blooming attention and interest from the scientific community as well as from the general public. The interest from the general public is mostly due to the recent events of terror around the world, which has demand for useful security systems, This work has been done to find out the various problems relating to face recognition and possible solution to these problems.

It can further be enhanced as a live project for our university as well. It can be used for the attendance system, which will ease out the process of taking attendance as well as save time.

Chapter-2

LITERATURE REVIEW

2.1

Face Recognition

Human often use faces to recognize individuals and the advancement in computing capability over the past few decades now enable similar recognition automatically. Early face recognition algorithms are simple geometric models, but the recognition process has now matured in to a science of sophisticated mathematical representations and matching processes. The proposed system for face recognition required the administrator to locate features on the photographs before it calculated distance and ratios to a common reference point, which were then compared to reference data.

Face Recognition Process :

A facial recognition system is a computer application to automatically identifying a person from a digital image or a video frame. One way to achieve this is by comparing selected facial features from the image to a facial database. It is typically used in security systems and can be compared to other biometrics such as fingerprint or human iris. Currently, developers came up with the design that is capable of extracting and picking up faces from the crowd and have it compared to an image source - database. The software has the ability to know how the basic human face looks like in order for it to work accordingly. Thus, developers designed these programs (by storing commands) to pinpoint a face and measure its features.

Face recognition is a biometric approach that employs automated methods to verify or recognize the identity of a living person based on his/her physiological characteristics. In general, a biometric identification system makes use of either physiological characteristics (such as a fingerprint, iris pattern, or face) or behaviour patterns (such as hand-writing, voice, or key-stroke pattern) to identify a person. Because of human inherent protectiveness of his/her eyes, some people are reluctant to use eye identification systems. Face recognition has the benefit of being a passive, non intrusive system to verify personal identity in a “natural” and friendly way.

There are different methods of facial recognition which involve a series of steps that serve to capturing, analyzing and comparing a face to a database of stored images. Some related software was designed to recognize similarities through pattern recognition. Pattern recognition is often used under the names of diagnosis and clarifications. Each of this software varies on how it is designed to work yet the function and concept is still the same that is identifying on facial landmarks. Because of these, facial recognition is hard to fool since it compares specific proportions and angles of the defined facial features.

Facial recognition software falls into a larger group of technologies known as biometrics. Biometrics uses biological information to verify identity. The basic idea behind biometrics is that our body contains unique properties that can be used to distinguish us from other persons.

Face recognition has a number of advantages over other biometrics. Firstly, it is non-intrusive. While many biometrics require the subject's co-operation and awareness in order to perform identification, such as looking into an eye scanner or placing their hand on a fingerprint reader, face recognition could be performed even without the subject's knowledge. Secondly, the biometric data used to perform recognition is in a format that is readable and understood by humans. This means that a potential face recognition system can always be backed up and verified by a human. For example, supposing a person was falsely denied access to a site by a face recognition system. That decision could easily be corrected by a security guard that would compare the subject's face with the stored image, whereas this would not be possible with other biometrics such as iris.

Benifits of Face Recognition :

- Accurate
- Cost-effective
- Non-invasive
- Uses legacy data
- Often is the only suitable biometric
- Built in human back up mechanism

2.2

MATLAB OVERVIEW

MATLAB (is the abbreviation of MATrix LABoratory) is special purpose computer program to perform engineering and scientific calculation. It is used for simple mathematical calculation such as mathematics, computation as well as to solve complex problems such as for understanding and teaching engineering concepts, data analysis, exploration, visualization, prototyping, analyzing and simulating of actual electrical and power systems.

MATLAB is a high-level language and interactive environment chosen to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and Fortran .

MATLAB can be used in wide range of application such as signal processing, image processing, control design , biological computation, finance modeling and analysis. The most interesting feature of MATLAB is that integration of MATLAB code with other languages and applications. In 1970s MATLAB was produced as a technical tool, because of its huge application it has expanded and now used for car design, aerospace and many advance researches.

Advantages of MATLAB : MATLAB has many advantages over conventional computer language, some of them are :

1. User friendly: Programs written in MATLAB can be easy modified with the built-in integrated development environment and corrected with the MATLAB debugger and hence used for rapid prototyping of new program.

2. Platform independence: MATLAB runs on many computer operating systems thus providing a high range of platform independence.

3. Predefined Function: MATLAB has a large numbers of functions that are predefined that gives tested and pre-packaged solutions to many basic technical problems such as mean, square root, standard deviation etc. Also one more special purpose tool of MATLAB to solve the complex problems in specific areas are Toolboxes.

4. Device-Independent Plotting: MATLAB has integral plotting and imaging commands which is not found in other languages of computer. This plots and images can be visualized on any graphical output device installed on computer on which MATLAB is present.

5. Graphical User Interface: MATLAB comprises of tool that permit the user to construct a graphical user interface. With this powerful mechanism of MATLAB, even an inexperience user can design data analysis program.

6. MATLAB Compiler: Flexibility of MATLAB lies in compiling MATLAB program into a device-independent p-code after that interpreting the p-code instruction at runtime which effects the execution of program since the code id interpreted instead of compiled.

Image Formats and Types Supported by MATLAB:

The following image formats are supported by MATLAB:

- BMP (Bitmap)
- HDF (Hubble Deep Field)
- JPEG (Joint Photographic Experts Group)
- PCX (Picture Exchange Format)
- TIFF (Tagged Image File)
- XWB

MATLAB supports four types of images

- Binary images
- Indexed Images
- Grayscale Images
- Truecolour Images

Color Models: The purpose of color model is to ease the specification of color in some standard. In general a color model is a specification of a coordinate system, subspace present in color space is used to represent each color by a single point.

RGB Color Space: RGB model is based on Cartesian coordinate system which is cube aid is shown below. The cube has RGB values at the three corners, colors like cyan, magenta and yellow and other three corners, black is at the origin and white is at farthest corner from the origin. The gray scale lies on the line joining black and white. They are termed as ‘’additive primaries’’ because the colors are added together to produce the desired color. Due to high correlation between color components: red, green and blue also each of the component is subjected to luminance effect from the light intensity of the environment, thus this suffers dissatisfaction from many image processing applications. Also practically, this model is not well suitable for describing colors in terms of human interpretation.

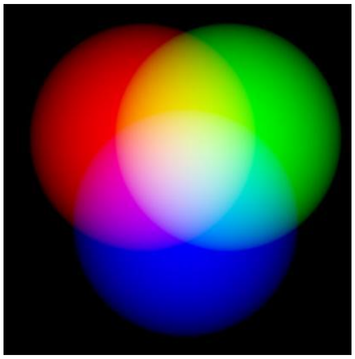


Fig.2) RGB image components

YC_rC_b Color Space: This color space has been defined to meet the increasing demand of digital algorithms in handling video information and has become the widely used color space in digital videos. It has three components, two of them is of chrominance and one is of luminance. YC_rC_b color model was developed in order to allow to transmit color information onto the television sets keeping in mind that existing black and white television sets continue to display the images in tones of grey. And got the characteristic of isolating luminance and color information, thus used in many applications such as compression.

Application areas of MATLAB:

- Technical computing
- Control system
- FPGA design
- Image and video processing
- Embedded systems
- Digital signal processing
- Computational finance

2.3

DIFFERENT TYPE OF APPROACHES FOR FACE RECOGNITION SYSTEM

- PCA (Principle Component Analysis)
- ICA (Independent Component Analysis)
- LBP (Local Binary Patterns)
- LDA (linear Discriminant Analysis)
- EBGM (Elastic Bunch Group Mapping)
- KERNEL Methods
- Trance Transform
- AAM (Active Appearance Model)
- 3-D Morphable Model

Main Approaches Of Face Recognition:

PCA : Principle component analysis (PCA) method used for global feature extraction is a powerful technique for extracting global structures from high-dimensional data set and has been widely used to reduce dimensionality and extract abstract features of faces for face recognition. It can also be used to identify patterns in data, and expressing the data in such a way as to highlight their similarities and differences. This provides an effective technique for dimensionality reduction. The PCA method has been extensively applied for the task of face recognition. Approximate reconstruction of faces in the ensemble was performed using a weighted combination of eigenvectors (Eigen pictures), obtained from that ensemble. The weights that characterize the expansion of the given image in terms of Eigen pictures are seen as global facial features.

LDA : Linear Discriminant is a “classical” technique in pattern recognition , where it is used to find a linear combination of features which characterize or separate two or more classes of objects or events. The resulting combination may be used as a linear classifier or, more commonly, for dimensionality reduction before it can be classified. In computerized face recognition, each face is represented by a large number of pixel values. Linear discriminant analysis is primarily used here to reduce the number of features to a more manageable number before classification. Each of the new dimensions is a linear combination of pixel values, which form a template. The linear combinations obtained using Fisher's linear discriminant are called Fisher faces, while those obtained using the related principal component analysis are called eigenfaces. Linear Discriminant Analysis easily handles the case where the within-class frequencies are unequal and their performance has been examined on randomly generated test data.

2.4
PCA approach for face recognition

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 factor analysis. The purpose of PCA is to reduce the large dimensionality of the data space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables), which are needed to describe the data economically. This is the case when there is a strong correlation between observed variables. 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 are suitable, such as signal processing, image processing, system and control theory, communications, etc. Face recognition has many applicable areas. Moreover, it can be categorized into face identification, face classification, or sex determination. The most useful applications contain crowd surveillance, video content indexing, personal identification (eg. driver’s licence), mug shots matching, entrance security, etc. The main idea of using PCA for face recognition is to express the large 1-D vector of pixels constructed from 2-D facial image into the compact principal components of the feature space. This can be called eigenspace projection. Eigenspace is calculated by identifying the eigenvectors of the covariance matrix derived from a set of facial images(vectors).

Mathematics of PCA :

A 2-D facial image can be represented as 1-D vector by concatenating each row (or column) into a long thin vector. Let’s suppose we have M vectors of size N (= rows of image * columns of image) representing a set of sampled images. p_j's represent the pixel values.

The images are mean centered by subtracting the mean image from each image vector. Let m represent the mean image.

$$m = \frac{1}{M}\sum_{i=1}^{M} x_i$$

And let w_i be defined as mean centered image

$$w_i = x_i\text{-m}$$

Our goal is to find a set of $e_i's$ which have the largest possible projection onto each of the $w_i's$. We wish to find a set of M orthonormal vectors e_i for which the quantity

$$h_i = \frac{1}{M}(e_i^T w_n)^2$$

is maximized with the orthonormality constraint

$$e_l^T e_k = S_{lk}$$

It has been shown that $e_i's$ and $h_i's$ are given by the eigen vectors and eigen values of the covariance matrix

$$C = WW^T$$

Where W is a matrix composed of the column vectors w_i placed isde by side. The size of C is N*N which could be enormous. A common theorem in algebra states that the vectors e_i and scalars h_i can be obtained by solving for the eigenvalues and eigenvectors of M*M matrix $W^T * W$. Let d_i and u_i be the eigenvectors and eigenvalues of WW^T , respectively.

$$WW^T d_i = u_i d_i$$

By multiplying left to both sides by W

$$WW^T(Wd_i) = u_i(Wd_i)$$

which means that the first M-1 eigenvectors e_i and eigenvalues h_i of WW^T are given by Wd_i and u_i respectively . Wd_i needs to be normalized in order to be equal to e_i. Since we only sum up a finite number of image vectors, M , the rank of Covariance Matrix can not exceed M-1 (The -1 come from the subtraction of the mean vector m).

The eigenvectors corresponding to nonzero eigenvalues of the covariance matrix produce an orthonormal basis for the subspace within which most image data can be represented with a small amount of error. The eigenvectors are sorted from high to low according to their corresponding eigenvalues. The eigenvector associated with the largest eigenvalue is one that reflects the greatest variance in the image. That is, the smallest eigenvalue is associated with the eigenvector that finds the least variance. They decrease in exponential fashion, meaning that the roughly 90% of the total variance is contained in the first 5% to 10% of the dimensions.

A facial image can be projected onto M' (<< M) dimensions by computing

$$\Omega = [v_1 v_2 \dots \dots \dots .. v_M]$$ ^ T

where $v_i = e_i^T w_i. v_i$ is the i^{th} coordinate of the facial image in the new space, which came to be the principle component. The vectors e_i are also images , so called , eigenfaces, or eigenfaces in our case. They can be viewed as images and indeed look like faces . So, Ω describes the contribution of each eigenface in representing the facial image by treating the eigenfaces as a basis set for facial images. The simplest method for determining which face class provides the best description of an input facial image is to find the face class k that minimizes the Euclidean distance

$$\mathcal{E}_k = ||(\Omega - \Omega_k)\|$$

where Ω_k is a vector describing the k^{th} face class. If $\mathcal{E}_k$ is less than some predefined threshold θ_ε, a face is classified as belonging to class k.

By means of PCA one can transform each original image of the training set into a corresponding eigenface. original image. If one uses all the eigenfaces extracted from original images, one can reconstruct the original images from the eigenfaces exactly. But one can also use only a part of the eigenfaces. Then the reconstructed image is an approximation of the original image. However, losses due to omitting some of the eigenfaces can be minimized. This happens by choosing only the most important features (eigenfaces). Omission of eigenfaces is necessary due to scarcity of computational resources. Thus the purpose of PCA is to reduce the large dimensionality of the face space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables), which are needed to describe the data economically. This is the case when there is a strong correlation between observed variables.

To generate a set of eigenfaces, a large set of digitized images of human faces, taken under the same lighting conditions, are normalized to line up the eyes and mouths. They are then all resample at the same pixel resolution (say m*n), and then treated as m-n-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 belong to the same vector space as face images, they can be viewed as if they were m×n pixel face images: hence the name eigenfaces. Viewed in this way, the principal eigenface looks like a bland androgynous average human face. Some subsequent eigenfaces can be seen to correspond to generalized features such as left-right and top-bottom asymmetry. Other eigenfaces are hard to categorize, and look rather strange. When properly weighted, eigenfaces can be summed together to create an approximate gray-scale rendering of a human face. Remarkably few eigenvector terms are needed to give a fair likeness of most people's faces, so eigenfaces provide a means of applying data compression to faces for identification purposes.

2.5

Eigen values and Eigen vectors

Large matrices can be costly, in terms of computational time, to use. Large matrices may have to be iterated hundreds or thousands of times for a calculation. Additionally, the behavior of matrices would be hard to explore without important mathematical tools. One mathematical tool, which has applications not only for Linear Algebra but for differential equations, calculus, and many other areas, is the concept of eigenvalues and eigenvectors. The words eigenvalue and eigenvector are derived from the German word "eigen" which means "proper" or "characteristic." An eigenvalue of a square matrix is a scalar that is usually represented by the Greek letter λ and an eigenvector is a non-zero vector denoted by the small letter x. For a given square matrix, A, all eigenvalues and eigenvectors satisfy the equation

$$Ax = hx$$

In other words, 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. This integer value is the corresponding eigenvalue of the eigenvector.

Eigenvectors possess following properties :

- They can be determined only for square matrices
- There are n eigenvectors (and corresponding eigenvalues) in a n*n matrix.
- All eigenvectors are perpendicular, i.e. at right angle with each other.

Since each eigenvector is associated with an eigenvalue, we often refer to an *x* and λ that correspond to one another as an eigenpair. An eigenspace is a space consisting of all eigenvectors which have the same eigenvalue. These eigenvectors are derived from the covariance matrix of the probability distribution of the high-dimensional vector space of possible faces of human beings and hence eigenfaces are a set of eigenvectors.

STEPS FOR RECOGNITION USING PCA :

The step by step instructions along with the formulas for the recognition of faces using Principal Component Analysis (PCA) are as follows:

STEP 1: Prepare the Data

The first step is to obtain a set S with M face images. Each image is transformed into a vector of size N and placed into the set.

$$S = \{ \Gamma_1, \Gamma_2, \Gamma_3, \text{.........} ,\Gamma_M \}$$

STEP 2: Obtain the Mean

After obtaining the set, the mean image Ψ has to be obtained as,

$$\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n$$

STEP 3: Subtract the Mean from Original Image

The difference between the input image and the mean image has to be calculated and the result is stored in Φ.

$$\varphi_i = \Gamma_i - \Psi$$

STEP 4: Calculate the Covariance Matrix

The covariance matrix **C** is calculated in the following manner

$$C=\frac{1}{M}\sum_{n=1}^{M}\varphi_n\,\varphi_n^T$$

STEP 5: Calculate the Eigenvectors and Eigenvalues of the Covariance Matrix and Select the Principal Components

In this step, the eigenvectors (eigenfaces) u_i and the corresponding eigenvalues λ_i should be calculated. From M eigenvectors, u, only M' should be chosen, which have the highest eigenvalues. The higher the eigenvalue, the more characteristic features of a facedoes the particular eigenvector describe. Eigenfaces with low eigenvalues can be omitted, as they explain only a small part of the characteristic features of the faces. After M' eigenfaces are determined, the “training” phase of the algorithm is finished. Once the training set has been prepared the next phase is the classification of new input faces. The recognition procedure consists of two major steps:

STEP 6: Transform the New Face

The new face is transformed into its eigenface components and the resulting weights form the weight vectors.

$$\omega_k = u_k^T(\Gamma - \Psi)$$

where ω = weight, μ = eigenvector,   = new input image, Ψ = mean face

The weight vector $^{\mathbf{T}}$ is given by,

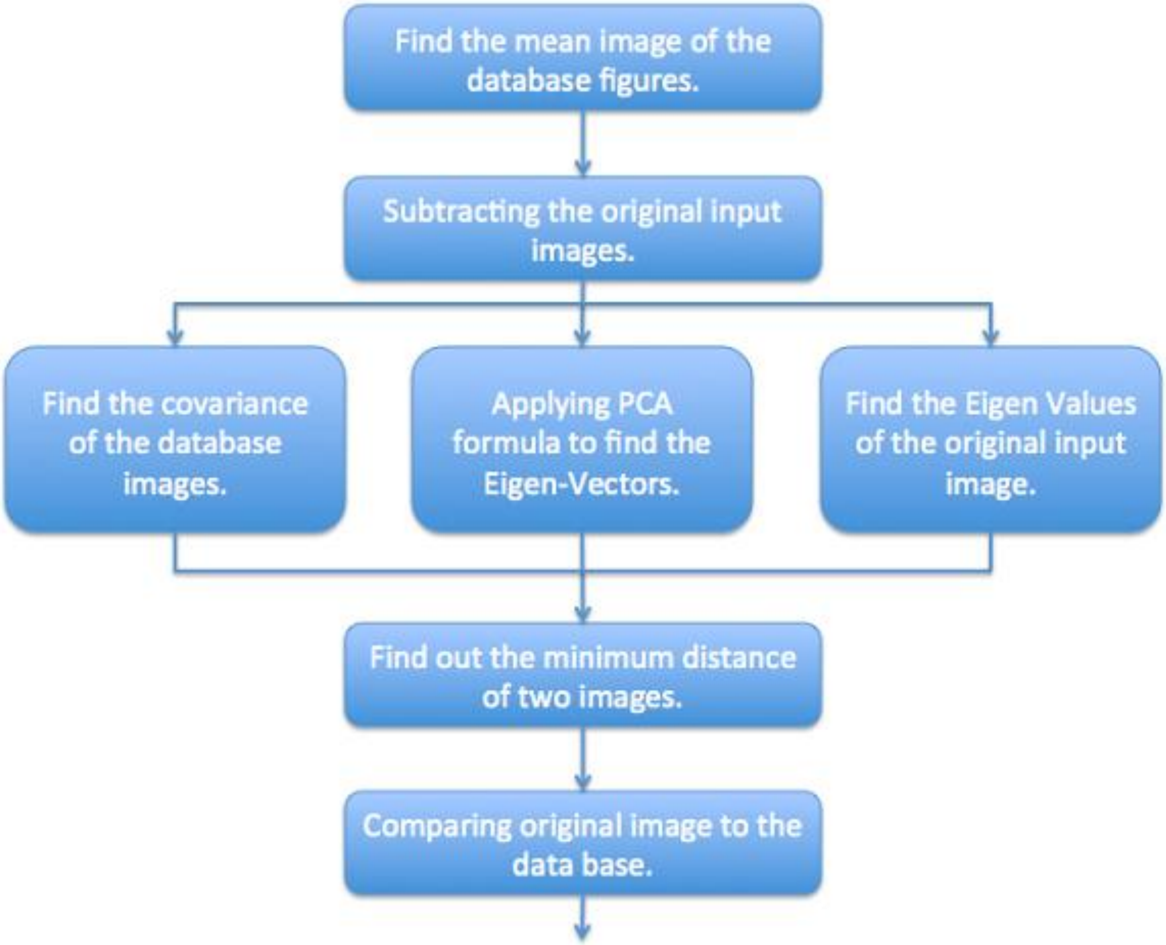
$$\Omega^T = [\omega_1, \omega_2, \;\ldots\ldots, \omega_M]$$

Chapter-3

Problem Solution And Work Done

3.1

Block Diagram



Find the mean image of the database figures.
Subtracting the original input images.
Find the covariance of the database images.
Applying PCA formula to find the Eigen-Vectors.
Find the Eigen Values of the original input image.
Find out the minimum distance of two images.
Comparing original image to the data base.

Fig.3) Block Diagram of face recognition technique

3.2

Algorithm (Flow Diagram)

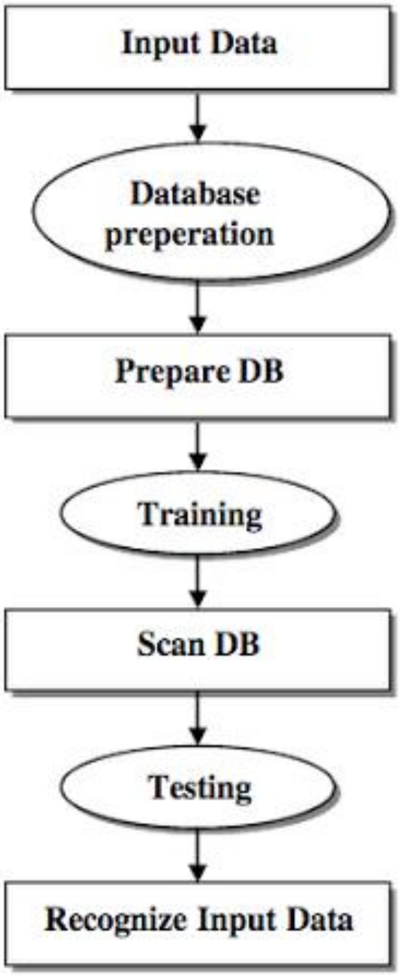


Fig.4) Algorithm of face recognition process

3.3

MATLAB CODES:

Test image :

I. Add image :

```
function []=addimage
while (1==1)
    choice=menu('Add an Image',...
                  'Choose From File',...
                  'Capture Now',...
          'Exit');
    if (choice ==1)
      ChooseFile=imgetfile
      capcha=imread(ChooseFile);
      capcha=imcrop(capcha,[180,40,480,680]);
      imshow(capcha);
      saveimage(capcha);
    end
    if (choice == 2)
        capturenow;
    end
    if (choice == 3)
        clc;
        close all;
        return;
    end
end
end
```

II. Capture now :

```
function []=capturenow
try close figure 1;end;
vid=videoinput('winvideo',1,'YUY2_640x480');
while (1==1)
    choice=menu('Capture Now',...
                  'Intialize Camera',...
                  'Capture Image',...
                  'Save Image',...
                  'Exit');
    if (choice ==1)
        inicamera(vid);
    end
    if (choice == 2)
        capcha=capture(vid);
    end
    if (choice == 3)
        saveimage(capcha);
    end
    if (choice == 4)
        delete(vid);
        clc;
        close all;
        return;
    end
end
end
```

III. Initialize camera :

```
function [] = inicamera(vid)
preview(vid);
end
```

IV. Capture :

```
function [capcha]= capture(vid)
    capcha=getsnapshot(vid);
    capcha=ycbcr2rgb(capcha);
    capcha=imcrop(capcha,[180,20,480,580]);
    imshow(capcha);
end
```

V. Save image :

```
function [ ] = saveimage( capcha )
imwrite(capcha,'InputImage.jpg');

end
```

VI. Recognition :

```
function OutputName = Recognition(m, A, Eigenfaces)
ProjectedImages = [];
Train_Number = size(Eigenfaces,2);
for i = 1 : Train_Number
    temp = Eigenfaces'*A(:,i); % Projection of centered images into facespace
    ProjectedImages = [ProjectedImages temp];
end
InputImage = imread('InputImage.jpg');
temp = InputImage(:,:,1);

[irow icol] = size(temp);
InImage = reshape(temp',irow*icol,1);
Difference = double(InImage)-m; % Centered test image
ProjectedTestImage = Eigenfaces'*Difference;
Euc_dist = [];
for i = 1 : Train_Number
    q = ProjectedImages(:,i);
    temp = ( norm( ProjectedTestImage - q ) )^2;
    Euc_dist = [Euc_dist temp];
end

[Euc_dist_min , Recognized_index] = min(Euc_dist);
OutputName = (Recognized_index);



end
```

Train image (database) :

I. Add image :

```
function []=addimage
while (1==1)
  choice=menu('Add an Image',...
         'Choose From File',...
         'Capture Now',...
     'Exit');
  if (choice ==1)
   ChooseFile=imgetfile
   capcha=imread(ChooseFile);
   capcha=imcrop(capcha,[180,40,280,380]);
   imshow(capcha);
   saveimage(capcha);
  end
  if (choice == 2)
    capturenow;
  end
  if (choice == 3)
    clc;
    close all;
    return;
  end
end
end
```

II. Capture now :

```
function []=capturenow
try close figure 1;end;
vid=videoinput('winvideo',1,'YUY2_640x480');
while (1==1)
    choice=menu('Capture Now',...
                  'Intialize Camera',...
                  'Capture Image',...
                  'Save Image',...
          'Exit');
    if (choice ==1)
        inicamera(vid);
    end
    if (choice == 2)
        capcha=capture(vid);
    end
    if (choice == 3)
        saveimage(capcha);
    end
    if (choice == 4)
        delete(vid);
        clc;
        close all;
        return;
    end
end
end
```

III. Initialize camera :

```
function [] = inicamera(vid)
preview(vid);
end
```

IV. Capture image :

```
function [capcha]= capture(vid)
    capcha=getsnapshot(vid);
    capcha=ycbcr2rgb(capcha);
    capcha=imcrop(capcha,[180,20,280,380]);
    imshow(capcha);
end
```

V. Save image :

```
function [ ] = saveimage(capcha)
SavePath = uigetdir('C:\Users\SHIVA PC\Desktop\face recognition\face rec pca\TrainDatabase',
'Select Student Folder' );
file_ext='.jpg';
folder_content = dir ([SavePath,'\*',file_ext]);
nface = size (folder_content,1);
str=int2str(nface+1);
str=strcat(str,'.jpg');
saveas=strcat(SavePath,'\',str);
imwrite(capcha,saveas);
disp('Image Sucessfully Saved As ');
disp(str);
end
```

Recognition process :

Create Database :

```
function []=CreateDatabase
cd TrainDatabase;
while (1==1)
    choice=menu('Create Database',...
                  ' Add  an  Image',...
                  ' Add  a  Folder',...
                  ' Exit');
    if (choice ==1)
        addimage;
    end
    if (choice == 2)
        addfolder;
    end
    if (choice == 3)
        cd ..;
        clc;
        close all;
        return;
    end
end
end
```

Delete Database :

```
function [ ] = DeleteDatabase(  )
disp('Please dont delete in between');
cd TrainDatabase
 while (1==1)
   choice=menu('Delete DataBase',...
            'Delete an Image',...
            'Delete a Folder',...
            'Exit');
   if (choice ==1)
      ChooseFile=imgetfile;
      delete(ChooseFile);
   end
   if (choice == 2)
      delfolder=uigetdir('E:\ufd\TrainDatabase','Delete Folder');
      rmdir(delfolder,'s');
   end
   if (choice == 3)
      cd ..
      clc;
      close all;
      return;
   end
end
```

Eigenfaces :

```
function [m, A, Eigenfaces] = EigenfaceCore(T)
disp('Creating Eigen Faces');
m = mean(T,2); % Train_Number = size(T,2);
A = [];
for i = 1 : Train_Number
    temp = double(T(:,i)) - m;
    A = [A temp];
end

L = A'*A; % L is covariance matrix C=A*A'.
[V D] = eig(L);
L_eig_vec = [];
for i = 1 : size(V,2)
    if( D(i,i)>1 )
        L_eig_vec = [L_eig_vec V(:,i)];
    end
end

Eigenfaces = A * L_eig_vec;  % A: centered image vectors
save train
end
```

Train Database :

```
function T = TrainDatabase(TrainDatabasePath)
no_folder=size(dir([TrainDatabasePath,'\*']),1)-size(dir([TrainDatabasePath,'\*m']),1)-2;
T = [];
disp('Loading Faces');
for i = 1 : no_folder
   stk = int2str(i);
   disp(stk);
   stk = strcat('\s',stk,'\*jpg');
   folder_content = dir ([TrainDatabasePath,stk]);
   nface = size (folder_content,1);
   disp(nface);
for j = 1 :  nface


   str = int2str(j);
   str = strcat('\',str,'.jpg');
   str = strcat('\s',int2str(i),str);
   str = strcat(TrainDatabasePath,str);
   img = imread(str);
   img = rgb2gray(img);


   [irow icol] = size(img);
   temp = reshape(img',irow*icol,1);
   T = [T temp];
end
end


end
```

Train Eigen faces :

```
function [m, A, Eigenfaces]=Trainit()

clear all
clc
close all
TrainDatabasePath = uigetdir('C:\Users\SHIVA PC\Desktop\face recognition\face rec pca\TrainDatabase', 'Select training database path' );
T = TrainDatabase(TrainDatabasePath);
[m, A, Eigenfaces] = EigenfaceCore(T);
end
```

Face Recognition :

```
function [ OutputName ] = FaceRec(m, A, Eigenfaces)
cd TestImage;
while (1==1)
    choice=menu('Face Recognition',...
                'Input Image From File',...
                'Capture Now',...
                'Recognition',...
                'Exit');
    if (choice ==1)
       try cd TestImage;close all; end;
         ChooseFile = imgetfile;
         capcha = imread(ChooseFile);
         capcha = imcrop(capcha,[180,20,480,680]);
```

```
     imshow(capcha);
     saveimage(capcha);
 end
 if (choice == 2)
    try cd TestImage;close all; end;
    capturenow;


 end
 if (choice == 3)
   OutputName=Recognition(m, A, Eigenfaces);
   n=((OutputName+1)/2);


   im=imread('InputImage.jpg');
   cd ..;
   img=strcat('TrainDatabase\s',int2str(n),'\1.jpg');
   SelectedImage=imread(img);
   subplot(121);
   imshow(im)
 title('Test Image');
 subplot(122),imshow(SelectedImage);
 title('Trained Image');
   disp('Image No');
   disp(int2str(n));


 end


if (choice == 4)
   clc;
    close all;
    return;
    end
end
```

Main Program :

```
close all
clear all
clc
while (1==1)
    choice=menu('Face Recognition System',...
                'Create Database of Faces',...
                'Delete DataBase',...
                'Train System',...
                'Face Recognition',...
                'Exit');
    if (choice ==1)
        CreateDatabase;
    end

    if (choice == 2)
        DeleteDatabase;
    end

    if (choice ==3)
        [m, A, Eigenfaces]=Trainit;
    end
    if (choice == 4)
        if exist('train.mat');
            load train;
        end
        FaceRec(m, A, Eigenfaces);
    end

    if (choice == 5)
        clear all;
        clc;
        close all;
        return;
    end

end
```

Conclusion And Future Scope

The face recognition strategy PCA (Principal Component Analysis) was implemented. PCA gave better results for varying poses. The results are pretty good for the test samples that we have considered.

Face recognition is quickly emerging as a viable authentication method today. With the adoption of standards and community awareness, this technology will become more and more acceptable. Current implementations of this technology are visible in airports, ATM's, border and security control checkpoints, crime investigation, entrance control in buildings, access control for computers in general, and other day-to-day affairs. Hence in this contemporary world, face recognition is playing an important role and can be concluded as an upcoming technology.

Further, we can improve the system by integrating with other biometric methods like speaker recognition, finger print, eye iris, and retina and voice recognition.

Scope and Features :

- The basic benefit in PCA is to reduce the dimension of the data.
- Smaller representation of database because we only store the training images in the form of their projections on the reduced basis.
- Noise is reduced because we choose the maximum variation basis and hence features like background with small variations are automatically ignored.
- PCA also benefits entrance control in buildings, access control for computers in general, for automatic teller machines in particular, day-to-day affairs like withdrawing money from bank account, dealing with the passport verification, and identifying the faces in a given database.

REFERENCES

[1] M.A. Turk and A.P. Pentland, “Face Recognition Using Eigenfaces”, IEEE Conf. on Computer Vision and Pattern Recognition, pp. 586-591, 1991.

[2] K. I. Diamantaras and S. Y. Kung, “Principal Component Neural Networks: Theory and Applica-tions”, John Wiley & Sons,Inc., 1996.

[3] Alex Pentland, Baback Moghaddam, and Thad Starner, “View-Based and Modular Eigenspaces for Face Recognition”, IEEE Conf. on Computer Vision and Pattern Recognition, MIT Media Laboratory Tech. Report No. 245 1994

[4] “Face recognition: Features versus templates”, R. Brunelli and T. Poggio, *IEEE Trans.* Pattern Analysis and Machine Intelligence, 15(10): 1042-1052, 1993

[5] “Human and machine recognition of faces: A survey”, R. Chellappa, C. L. Wilson, and S. Sirohey, Proc. of IEEE, volume 83, pages 705-740, 1995

[6] “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection”, P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, IEEE Trans. Pattern Analysis and Machine Intelligence.

[7] “Illumination cones for recognition under variable lighting: Faces”, A. S. Georghiades, D. J. Kriegman, and P. N. Belhumeur, Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pages 52-59, 1998

[8] “Human face segmentation and identification”, S. A. Sirohey, Technical Report CAR-TR-695, Center for Automation Research, University of Maryland, College Park, MD, 1993

[9] “Automatic recognition and analysis of human faces and facial expressions: A survey”, A. Samal and P. A. Iyengar, Pattern Recognition, 25(1): 65-77, 1992

Links :

- http://www.ijeat.org/attachments/File/v1i5/E0419051512.pdf
- http://scholar.fju.edu.tw/%E8%AA%B2%E7%A8%8B%E5%A4%A7%E7%B6%B1/upload/058029/content/991/D-7604-07699-E.pdf
- http://www.cc.gatech.edu/~kihwan23/thesis_face.html
- http://sun.aei.polsl.pl/~mkawulok/stud/fr/lect/07.pdf