# EFFICIENT COMBINED TECHNIQUES FOR PERSON RECOGNITION WITH COVERED FACE AND EYES

MD. MAHBUBUL ALAM

MASTER OF SCIENCE

NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

# EFFICIENT COMBINED TECHNIQUES FOR PERSON RECOGNITION WITH COVERED FACE AND EYES

MD. MAHBUBUL ALAM

Thesis submitted in partial fulfilment of the requirements for the award of the degree of Master of Science in Information and Communication Engineering

Department of Information and Communication Engineering
NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

DECEMBER 2020

# SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion this thesis is adequate in terms of scope and quality for the award of the degree of Master of Science in Information and Communication Engineering.

DR. MD. ASHIKUR RAHMAN KHAN
PROFESSOR & CHAIRMAN
DEPARTMENT OF INFORMATION AND COMMUNICATION ENGINEERING
NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

# CO-SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion this thesis is adequate in terms of scope and quality for the award of the degree of Master of Science in Information and Communication Engineering.

ZAYED- US- SALEHIN
ASSISTANT PROFESSOR
DEPARTMENT OF INFORMATION AND COMMUNICATION ENGINEERING
NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

# Certificate of Thesis Approval

This thesis titled **Efficient Combined Techniques for Person Recognition with Covered Face and Eyes** submitted by **Md. Mahbubul Alam, Roll No. ASH1711MSc134M,** Session: 2016-2017 has been accepted as satisfactory in partial fulfilment of the requirement for the degree of Master of Science in Information and Communication Engineering.

## BOARD OF EXAMINERS

| | | |
|---|---|---|
| 1. | ______________________________<br>Dr. Md. Ashikur Rahman Khan<br>Professor<br>Department of Information and Communication Engineering<br>Noakhali Science and Technology University<br>Noakhali-3814 | Chairman |
| 2. | ______________________________<br>Zayed- Us- Salehin<br>Assistant Professor<br>Department of Information and Communication Engineering<br>Noakhali Science and Technology University<br>Noakhali-3814 | Member |
| 3. | ______________________________<br>Sultana Jahan Soheli<br>Assistant Professor<br>Department of Information and Communication Engineering<br>Noakhali Science and Technology University<br>Noakhali-3814 | Member |
| 4. | ______________________________<br>Dr. Mohammad Shamsul Arefin<br>Professor<br>Department of Computer Science and Engineering<br>Chittagong University of Engineering & Technology<br>Chittagong, Bangladesh | Member<br>(External) |

# STUDENT’S DECLARATION

This thesis report is submitted to the Dept. of Information and Communication Engineering (ICE), Noakhali Science and Technology University (NSTU) in partial fulfillment of the requirements for having the M.Sc. degree in ICE. This is also needed to certify that the thesis work is under the M.Sc. course of ICE, NSTU. So, I hereby declare that this thesis report has not been submitted elsewhere for the requirement of any kind of degree, diploma or publication.

MD. MAHBUBUL ALAM

ID: ASH1711MSc134M

SESSION: 2016-2017

# ACKNOWLEDGEMENTS

I am very grateful and would like to express my gratitude to my supervisor Professor Dr. Md. Ashikur Rahman Khan for his setting up, invaluable guidance, continuous encouragement and constant support in making this research possible. He has always encouraged me with his outstanding and efficient professional conduct. I am grateful for all the support that I have received from him.

My sincere thanks to all my co-supervisor and teachers of the Department of Information and Communication Engineering (ICE) in NSTU for their valuable suggestion and academic help during the course of work.

I acknowledge my sincere indebtedness and gratitude to my parents for their love, dream and sacrifice throughout my life.

# ABSTRACT

Face is the most common parameter which defines any person uniquely. Face recognition is a process of identifying or verifying the identity of an individual using their face. Principal component analysis is algorithm for face recognition. In facial recognition there are some complexities that degraded the efficiency of face recognition using Principal Component Analysis for covered face and covered eyes. Iris recognition is a process of detecting a person uniquely by reading the eyes. Iris recognition can remove the covered face complexities but when someone wears sunglasses which covered eyes then iris recognition will not work properly. Face recognition is not efficient for covered face like faces with mask, long beard and naqab. Iris recognition is not efficient for covered eyes like eyes with sunglasses. To overcome all these complexities which include covered face or eyes, our proposed efficient models are Combined PCA-Segmentation, Combined PCA-Daugman and Combined PCA-SVM Face Recognition. In this work, Principal Component Analysis process will take place combined with Segmentation, Daugman and Support Vector Machine. Segmentation will help for creating upper and lower face evaluation separately, Support Vector Machine will help to detect the covered faces to improve the recognition accuracy. To calculated the accuracy of the systems there are four image datasets used called ORL, Yale, Real and CASIA which is iris image dataset. This work has used also real image dataset. For all dataset each model has provided better accuracy than solo PCA. The motto of this work is to remove the complexities of beard, mustache, hijab and sunglasses at facial recognition to increase the overall accuracy for improving the face recognition.

# TABLE OF CONTENTS

| Content | Page |
|---|---|

# LIST OF TABLES

# LIST OF FIGURES

# CHAPTER 1

# INTRODUCTION

## 1.1 INTRODUCTION

The purpose of this chapter is to provide an introduction of the study. This chapter discusses the background of the study along with the problem statement. Moreover, this chapter presents the objectives and scope of the study. In addition, it discusses the organization of the report.

## 1.2 BACKGROUND

Face Recognition is the process to recognize a person uniquely by evaluating face. It is a biometric identification process by which one can uniquely identify and verifying individuals by analyzing the contours of faces [1]. Facial recognition is widely used in various security and academic aspects. It is used in automatic attendance system as well as in national security like identifying terrorist, identifying invaders for home security. There have various statistical and neural network approaches for recognizing faces in computer system. Statistical approaches [2] include- Principal Component Analysis, Discrete Cosine Transform, Linear Discriminant Analysis, Locality Preserving Projections, Gabor Wavelet, Independent Component Analysis, Kernel PCA. There also have some machine learning approaches for facial recognition like

Neural networks with Gabor filters, Neural networks and Hidden Markov Models, Fuzzy neural networks.

Principal Component Analysis (PCA) is an algorithm for face recognition process [3,4]. PCA method uses eigen faces and eigen vectors to differentiate different faces from one another [5,6]. For covered face recognition PCA algorithm is not efficient to detect person accurately. PCA provides very less accuracy for covered faces such as faces with mask and naqab. When a person wears sunglass or somehow a person with covered eyes, Iris recognition will not work properly [7,8]. It is a great limitation of Iris recognition method to work with covered or closed eyes.

We have applied combined PCA and Segmentation method to overcome complexities to detect a person. This is one of the efficient techniques that recognizes a person even after the person wears sunglass, mask or increased beard-mustache. By using Segmentation process an image can be splitted into several separate images [9,10,11]. After segmentation the PCA algorithm will be applied on the segmented images. The segmentation process will be work on upper region of face and lower region of face. For the implementation of combined PCA and Segmentation MATLAB software is used because it is the very famous and reliable platform to work. This work proposes PCA-Segmentation combined approach which overcomes a number of difficulties of the covered face and eyes.

Unimodal biometric system is the system which uses only one indivitual feature to identify a person. There are some individual characteristics like face, iris and fingerprint etc. Face recognition is an unimodal biometric system that utilizes face to detect a person. Facial recognition method normally used in various field like security and attendance. Principal Component Analysis (PCA) is an algorithm for face recognition process that uses eigen faces and eigen vectors to indentify person uniquely.

Iris recognition [12] is another unimodal biometric system which utilizes iris of a person to detect properly. There are some common iris recognition algorithm [13] such as Daugman approach, Wildes approach, Li Ma approach, Tisse approach, LBP IN ADDTION TO LVQ method and Masek Method. Daugman algorithm is efficient and popular one for iris recognition method. In Unimodal system, there are some problems like noisy data, non universality and intra-class variation. Due to these problems unimodal system generally do not cross the expected level of performance. Multimodal biometric system which combines mone than one individual features to detect a person more accurately. Multimodal biometric system can recover the lack of accuracy for unimodal biometric system.

In this work, we combined face recognition and iris recognition method to detect a person and to increase accuracy of indentification using multimodal biometric system. For face recognition we used Principal Component Analysis with ORL, YALE and Real face dataset and for iris recognition we used Daugman algorithm with CASIA dataset. Therefore, we propose an efficient multimodal biometric technique called combined PCA-Daugman method. To recognize a person uniquely various biometric approach are used such as face, iris, fingerprint and so on. Face recognition approach is used in many organizations, universities and airports for counting attendance and criminal detection also. Principal Component Analysis (PCA) is an algorithm which is broadly used for face and object detection. It calculates eigen values and eigen vectors using a covariance matrix to create a set of eigen faces that differentiate person from person. But there are some barrier situations such as a person wears sunglasses or mask then PCA method does not have satisfactory performance to detect a person accurately. To recognize a face with mask may iris recognition will be efficient method but when system have to detect a person who wears sunglasses then iris recognition method is not efficient enough.

To remove all these limitations and to recognize a person even after with sunglasses and mask, we combined an image classifier and principal component analysis. Image classifier is the method which classify images based on the features and various parameters to label a class. Support Vector Machine (SVM) [14] is well known image classifier which is supervised machine learning method. It is able to categorize when a sets of labeled training data is provided for each category. First of all, we have used three training dataset which are for normal face, face with mask and face with sunglasses. Image classifier called support vector machine have used to classify a test image and label the test image as normal face, masked face or face with sunglasses. For three different classes there are three different training image datasets to detect a person accurately using PCA method. For the combination of PCA and SVM, MATLAB software and real face database are used which contains real face image with sunglasses or mask.

## 1.3 PROBLEM STATEMENT

Principal Component Analysis (PCA) is an algorithm for face recognition process that uses eigen faces and eigen vectors to indentify person uniquely. In facial recognition there are some complexities like covered face that degraded the efficiency of PCA algorithm these are related to long beard, mustache, hijab, naqab and mask. Iris recognition is another unimodal biometric system which utilizes iris of a person to detect properly. In Iris recognition there are some complexities like covered eyes that degraded the efficiency of it these are related to sunglasses and closed eyes. When a person wears sunglass or a person wear mask then PCA method does not provide satisfactory performance to detect the person accurately. To recognize a face with mask may iris recognition will be efficient method but when system have to detect a person who wears sunglasses then iris recognition method will not be efficient because sunglasses hide the eyes, iris is obscured by sunglasses and it can not take clear iris image as input. To overcome all these complexities in this work proposed methods are combined PCA-Segmentation method, PCA-Daugman method and PCA-SVM method.

## 1.4 OBJECTIVES

The main objective of this study is to enhance the performance of face and iris recognition by increasing the accuracy for covered face and covered eyes. There are some others objectives of this work to obtain ultimate goal such as-

(i) To remove long beard and mustache complexities

(ii) To remove hijab, scarf and naqab complexities

(iii) To remove sunglass complexity

(iv) To remove mask complexity

(v) To recognize a person efficiently and to increase the accuracy of detection.

This work also will be helpful for the application of automated attendance system, criminal investigation, immigration control and for overall security system.

## 1.5 SCOPE OF THE STUDY

This study will focus on efficient face and iris recognition for covered face or covered eyes using combined PCA-Segmentation, PCA-Daugman and PCA-SVM. This work will let face and iris recognition to detect a person properly even after the person wears mask, hijab, naqab or sunglasses and the person with long beard, mustache also. In addition, combined methods are applied on ORL, Yale, CASIA and Real face databases. Moreover, the proposed system will enhance the accuracy and performance of recognition system. In this work, all methods have provided better accuracy than traditional ones for various face database. In real face database the images are collected from Bangladesh. Main limitation of this work is that when a person wears mask and sunglasses both then the methods will not provide better accuracy because of lacking the features analysis.

## 1.6 ORGANIZATION OF THE REPORT

The rest of this dissertation is comprised of four chapters: the literature review, algorithms and basic terminologies, research methodology, results and discussion, and conclusions. Chapter 2 reviews past research conducted with regards to recognize person using their face and eyes features. Several previous works are also discussed about solo PCA and Daugman algorithm. Chapter 3 describes the details of used databases, working procedure and methodology. It describes the details of different algorithms and basic terminologies which are used in this work like principal component analysis, segmentation, Euclidean distance, daugman's algorithm and support vector machine. This chapter also deals with the procedure to carry out the experiments, measurements and computation of the process output. Eventually, Chapter 3 delivers the details of model development through combination of different methods. Chapter 4 presents the results obtained in favor of combined models as well as this chapter provides the results and discussion of proposed models over different standard database and also real face database. Chapter 5 reports the conclusions and recommendations for future work.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 INTRODUCTION

The purpose of this chapter is to provide a review of past research efforts related with face and iris recognition. This chapter discusses the relevant face and iris recognition researches that had been done before by researchers. Moreover, this chapter presents the current techniques used by PCA and Daugman method for face and iris recognition with the limitations of it to recognize covered face and covered eyes. In addition, it discusses the previous researches that had been done before using the traditional techniques. Literature review of this study is divided into three sections as Face recognition, Iris recognition and SVM because the study works with all these techniques.

## 2.2 FACE RECOGNITION

There are a lots of work based on face recognition using various methods like PCA. Although these approaches are not efficient enough for covered face and eyes.

In 2017, Singh et al. introduced a deep learning framework that disguised face identification by detecting 14 facial key-points and the two annotated facial key-points datasets since the training of deep learning architectures relies on large annotated datasets [15]. In 2018, Priya et al. proposed a new way of using a deep neural network for face recognition. Their approach provided only the extracted facial features instead of providing raw pixel values as input. It provided the accuracy of 97.05% and decreased the complexity on Yale faces dataset [16].

Gondole et al. presented in 2018 a novel approach to face recognition which considers both shape and texture information to represent the face [17]. The face area is first divided into

small regions from which Local Binary Pattern (LBP) histograms are extracted and concatenated into a single, spatially enhanced feature histogram efficiently representing the face image.
In 2017, Ayi, M. et al. proposed a model for face detection and tracking using MATLAB software interfacing with Arduino board [18]. They developed a real-time application like security system that is necessary in several platforms and also implemented the real-time face detection and tracking using hardware devices like Webcam and Arduino board with Microcontroller as input and output devices respectively.

In 2015, Parvaneh and Hadi combined the principal compo- nent analysis (PCA), eigenface and support vector machines (SVM) using N-fold method for face recognition [19]. In 2009, Faruqe and Hasan proposed a method of face recognition using Principle Component Analysis (PCA) which is used to play a key role in feature extractor and the SVMs which are used to tackle the face recognition problem [20]. They illustrated the potential of SVMs by using the Cambridge ORL Face database.

Therefore, we proposed a combined PCA and Segmentation technique using ORL, YALE and Real image dataset for face recognition which resolved the problems of covered face and eyes.

## 2.3 IRIS RECOGNITION

There are a lots of work based on iris recognition. Some previous works with Principal Component Analysis and Daugman method are mentioned here.
In 2017, Ammour et al. proposed multimodal biometric system using face and iris. They applied score level matching fusion and experimented on ORL, CASIA database [21]. In 2019, Mansoura et al. proposed to combine several biometric modalities like face and iris that improved performance of recognition. They used Fast Fourier Transform (FFT) and singular value decomposition (SVD) approaches to extract features [22].

Rabiul et al. in 2015 proposed multimodal biometric system in which ratio score fusion technique used to combine face and iris. Discrete Hidden Markov Model (DHMM) used for evaluation result and applied on ORL and CASIA database [23]. Singh proposed in 2019 a multimodal system that used FKP and Iris. They used PCA for features extraction and the matching is calculated using Neuro fuzzy neural network (NFNN) [24].

In 2009 Jie Lin et al. proposed a method for person recognition by the combining of iris and face. In this method, face recognition based on combined iris and face features for representing persons and then acts the modified PUM on these features for recognition [25]. They previously applied the modified PUM method for face recognition. In 2017, S. D. Jamdar and Y. Golhar combined the features of face, ear and iris to multimodal recognition system. They introduced ear as the feature for multimodal biometric system and developed multimodal system by combining Ear, iris and Face recognition [26].

Y. Bouzouina and L. Hamami in 2017 developed a biometric bimodal system for the identification of the person by combining the modality of the face and the iris [27]. They also introduced face recognition by using DCT and PCA transform. They used Snake method for the improvement of performance of the iris recognition by the precision of the segmentation. In 2005, T. Ko discussed some scenarios of multimodal biometric systems using fingerprint, face and iris recognition to improve recognition system accuracy and also discussed that the way to increase the image quality of fingerprint, face and iris for efficient multimodal biometric systems [28]. Azom et al. proposed a hybridized fusion strategy in 2015 that combines face and iris based on feature and score level fusion using a decision level fusion rule using the ORL face and CASIA iris datasets [29].

In 2016, V. V. Dakre and P. G. Gawande used a combination of face and iris biometric traits. They used Viola Jones algorithms for detection of face and circular Hough transform to segmented iris part in the image [30]. they used Modular Principal Component analysis (MPCA) algorithm and 2D Gabor filter for feature extraction for face and iris respectively.

In this work, we proposed a combined PCA and Daugman method for both face and iris multimodal recognition using the ORL and YALE, Real face dataset and CASIA image dataset to identify a person efficiently and to remove the limitations of unimodal system.

## 2.4 Image Classification

Some previous works with Principal Component Analysis and Support Vector Machine method are mentioned here.

In 2008, Zhang et al. proposed an efficient face detection method using principal components analysis (PCA) and support vector machine (SVM) [31]. They used SVM classifier to detect face feature in the test image. They also used PCA to reduce the dimension of sample data. Their proposed method had a effective and more accurate detection rate. Li and Chen in 2012 is presented a novel technique for face recognition that combined PCA and SVM [32]. They used PCA to extract feature, wavelet transform for image processing. They also used support vector machine (SVM) as classifiers in recognition stage.

In 2010, Gumus et al. presented an evaluation technique for face recognition using PCA, wavelets and SVM. They used Principal Component Analysis (PCA) as feature extracting techniques and Support Vector Machines (SVMs) for classification step [33]. They examined the technique using ORL face database and obtained the classification accuracy 98.1%. In 2013, Luo introduced a hybrid method of principal component analysis and local binary pattern (LBP). The support vector machine (SVM) is used for facial expression recognition [34]. They proved that this method more effective for expression recognition and got higher recognition rate than the traditional recognition methods.

In 2018, Li proposed a methodology to solve the problem of full illumination variation [35]. For the illumination normalization, Gabor wavelet and principal component analysis methods are applied for feature extraction. And for face classification a Support Vector Machine classifier is used. Ragab et al. used principal component analysis in 1999 to an Egyptian Face Database which consisted of 1100 face images of the Egyptian [36].

Rui et al. in 2017 used principal component analysis (PCA) for dealing with the text data and they proposed a method called combined principal component analysis and K - average algorithm for verification of effectiveness [37]. In 2010 Song et al. applied principal components analysis (PCA) to feature selection. The proposed the method for feature selection from a numerical analysis viewpoint. They also proved the potentiality of PCA for feature selection [38]. In this work, we proposed a combined PCA and Support Vector Machine method for face recognition using the Real face dataset to identify a person efficiently and to remove the limitations of covered face and eyes.

## 2.5 SUMMARY

The analysis of literature review had broadened the scope of face and iris recognition issues. The information and findings collected form this chapter is used as a guidance to develop the proposed systems. By reviewing the face and iris recognition system case study, we can conclude that PCA, Daugman and SVM play an important role in the recognition system although these solo traditional techniques are not efficient for covered face and eyes. The accuracy for covered face detection is not good enough using the methods which were used in these previous works. This chapter has also demonstrated the necessities of an efficient technique which is able to detect a person efficiently even after the person wears mask or sunglasses.

In this work, the contribution is that the present work is designed to overcome covered face and eyes complexities using our proposed models called Combined PCA-Segmentation, Combined PCA-Daugman and Combined PCA-SVM. These models efficiently able to detect a person when he or she wears sunglasses, naqab or mask, even when the person have long beard or mustache. This work is experimented over different image datasets include real image dataset. We used segmentation method along with PCA to improve the accuracy of detection for covered face and eyes. Daugman method is used to increase the detection accuracy for covered face. Also, we used SVM to classify images into masked face and normal face which improves the performance of detection. We calculated the performance of the models for various cases and compared these accuracies with each other to justify the efficiency and better performance of the models.

# CHAPTER 3

# METHODOLOGY

## 3.1 INTRODUCTION

This chapter presents an outline of research methods that were followed in the study. It provides information on the datasets, algorithms and basic terminologies used in the study. This chapter describes the research methods that were chosen for the purpose of this study and the reasons for this choice. The requirements of this work and the procedures that were followed to carry out this study are included.

## 3.2 NECESSARY TERMINOLOGIES

İn this work, face recognition algorithm called Principle Component Analysis (PCA) is used with the Segmentation technique , Daugman model and Support Vector Machine to detect a person uniquely and efficiently who have covered face or eyes.

### 3.2.1 Principal Component Analysis

Principle Component Analysis algorithm (PCA) is a statistical technique which commonly used for patterns recognition in data of high dimension. It is important in face recognition to reduce dimension vector [39].

The steps of PCA algorithm are:

(a) ***Mean vector***

If there are N samples then the mean vector is :

$$\bar{S} = \frac{(S_1+S_2+\cdots+S_N)}{N} \tag{3.1}$$

(b) ***Mean adjusted matrix***

The mean adjusted vector can be calculated for every image vector (i) as:

$$\overline{S_i} = (S_i - \bar{S}) \tag{3.2}$$

The mean adjusted matrix $S_{mean}$ consists of every mean adjusted vector:

$$S_{mean} = [\overline{S_1} \ldots \overline{S_N}] \tag{3.3}$$

(c) ***Covariance matrix***

$$\begin{matrix} C_{(0,0)} & C_{(0,1)} & C_{(0,2)} \\ C_{(1,0)} & C_{(1,1)} & C_{(1,2)} \\ C_{(2,0)} & C_{(2,1)} & C_{(2,2)} \end{matrix} \tag{3.4}$$

$$C_{i,j} = (S_i - \bar{S}).(S_j - \bar{S}) \tag{3.5}$$

Where,

$\bar{S}$= The mean vector computed in step 1

$S_i$= The i$^{th}$ image vector

$S_i$= The j$^{th}$ image vector

Covariance measurement of each sample is represented by the covariance matrix.

(d) ***Eigen vectors and Eigen values***

To compute Eigen vectors and Eigen values of the covariance matrix, Equation for the Eigen values λ of the covariance matrix:

$$\det(\lambda I - C) = 0 \tag{3.6}$$

Where,

det = The determinant of the matrix

$\lambda$ = The eigen values associated with the matrix

$I$ = The identity matrix

$C$ = The covariance matrix

For a given Eigen value $\lambda_k$, the Eigen vector $EV_k$can be computed as follows:

$$(\lambda_k I - C) * EV_k \ = 0 \tag{3.7}$$

Where,

$\lambda_k$ = One of the highest eigen values kept

$I$ = The identity matrix

$C$ = The Covariance matrix

$EV_k$ =The eigen vector which will be calculated

(e) ***Compute the basis vectors***

All Eigen vectors are in a matrix:

$$EV = [V_1 \dots V_n] \tag{3.8}$$

In step (b), computed mean adjusted matrix should be multified by the Eigen vector matrix to compute the basis vectors:

$$S_B = S_{mean} * EV \tag{3.9}$$

Where-

$S_B$ = Basis vectors dimensions

$S_{mean}$ = Mean adjusted matrix

$EV$ = Matrix of eigen vector

(f) ***Represent image by basis vectors***

Following formula is of combination of the basis vectors:

$$[n\ numbers] = (S_{sample} - \bar{S})^T * S_B \tag{3.10}$$

Where-

$S_{sample}$ = The sample we want to represent using basis vectors

$\bar{S}$ = The mean adjusted vector computed in step 2

$S_B$ = The basis vectors dimensions

### 3.2.2 Daugman's Algorithm

Daugman's algorithm is very common and popular segmentation method for iris recognition.Using Daugman's algorithm, iris recognize process works efficiently. Applying an integro-differential operator the Daugman's algorithm is used to find the iris and pupil contour [40,41]-

$$\max\,(r, x_0, y_0)\left|G_\sigma\,(r)\ *\ \frac{\partial}{\partial r}\oint_{r,x_0,y_0}\frac{I(x,y)}{2\pi r}\,ds\right| \tag{3.11}$$

where,

$x_0, y_0$ is center

and $r_0$ is the radius of of coarse circle

$G_\sigma\,(r)$ is Gaussian smoothing function

$\Delta r$ is radius range

$I(x, y)$ is original iris image

The Daugman's algorithm flowchart is shown in Figure 3.1 Gaussian filter decreases the noises and makes easier to obtain characteristics. It initializes the center and radius of iris and calculates the circle gradient. Gradient is directional change in the intensity or color in image.

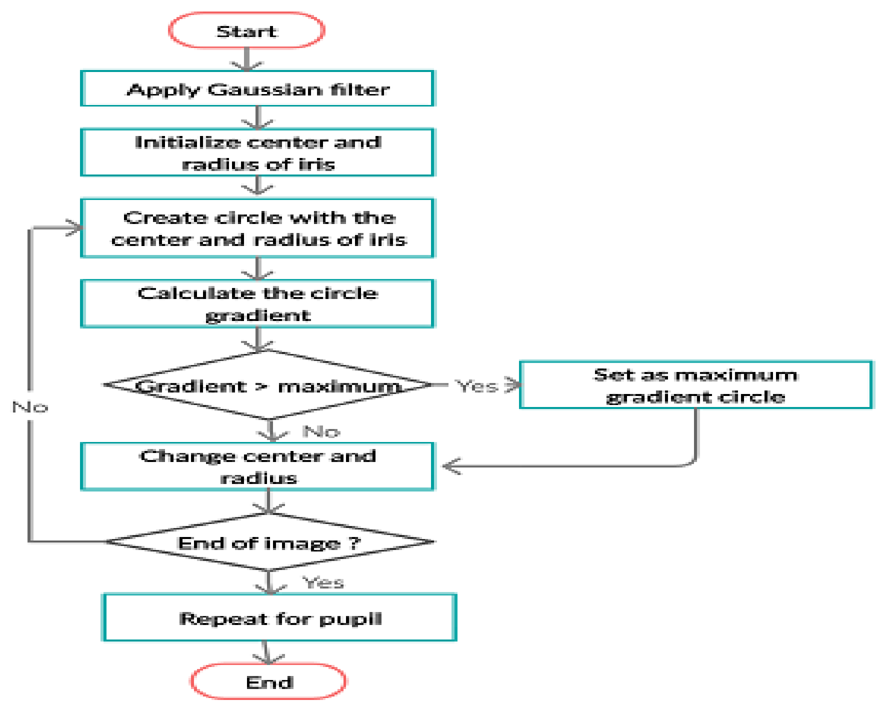



**Figure 3.1:** Flowchart of Daugman's algorithm

### 3.2.3 Support Vector Machine

SVM is a linear learning machine based on the principle of optimal separation of classes and was initially designed to solve classification problems. To find a linear separating hyperplane that separates classes of interest SVM is used. In a multidimensional space, hyperplane is a plane and is also called a decision surface or an optimal separating hyperplane or an optimal margin hyperplane. The linear separating hyperplane satisfies two conditions and is placed between classes. First, all data vectors that belong to the same class are placed on the same side of the hyperplane. Second, the distance between the closest data vectors in both the classes is maximized. In other words, the optimum hyperplane is the one that leaves the maximum margin between the two classes, where margin is defined as the sum of the distances of the hyperplane from the closest data vectors of the two classes. The data vectors forming the boundary of classes are located on supporting hyperplanes for each class. Thus, these data are called the Support Vectors and are the most significant ones for SVM. Many a times, a linear separating hyperplane is not able to classify input data without error. The data are transformed to a higher dimensional space using a non-linear transformation function that spreads the data apart such that a linear separating hyperplane may be found in that space.

### 3.2.4 Segmentation

Image segmentation refers to the process of partitioning an image into various segments. Image segmentation has several techniques- Thresholding Method, Edge Based Segmentation Method, Region Based Segmentation Method, Clustering Based Segmentation Method, Watershed Based Methods [42]. Region Based Segmentation Method uses splitting techniques for segmenting an image into various regions. Cropping is also a useful way to segment an image into multiple images.

### 3.2.5 Euclidean Distance

Euclidean distance is the distance between two points in Euclidean space. The Euclidean distance for one dimension can be calculated by measuring the absolute value of the difference between two points. If x and y are two points the Euclidean distance is |x-y|. The absolute value is used to consider non- negative value for distance measurement. To compute the Euclidean distance, d for two dimensions where two points are (x1, y1) and (x2, y2) the formula is-

$$d = \sqrt{(x1 - y1)^2 + (x2 - y2)^2} \tag{3.12}$$

## 3.3 DATA SET

There we have used the four different image databases, among these three databases are face image database -

(i) ORL database

(ii) YALE database

(iii) Real face database

And another one is for eyes image database called - CASIA-IrisV1 database

ORL face database [44, 45] is most applied face database in which 400 images of 40 persons and there are 10 different images per person. All the images have various expressions. Figure 3.2 represents the sample of ORL face dataset.

**Figure 3.2:** Sample of ORL face data set

The Yale Face Database [46, 47] is commonly used database which has total 165 images of different 15 person and images are in GIF format. There are different 11 images per person with different expression like normal, sad, surprised and sleepy [48]. Figure 3.3 represents the sample of Yale face dataset.

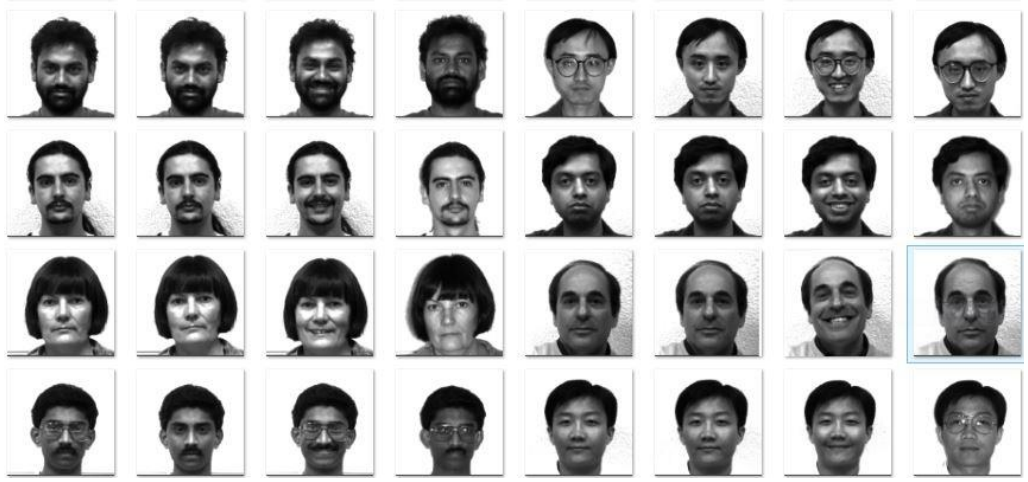


**Figure 3.3:** Sample of Yale face data set

In this work, very special thing is that there real face data set are used for experiment which is developed by us. Real face data set contains 300 images of 60 different persons with 5 different images each. Figure 3.4 represents the sample of Real face dataset.

**Figure 3.4:** Sample of Real face data set

The faces of this data set is real and collected from our close juniors, seniors and friends. In this data set there are both male and female with different ages. In this data set the images have various expressions such as close and open eyes, with sun glass, hijab, beard and mustache. CASIA Database Version 1.0 (CASIA-IrisV1) is very well known image database of 108 eyes with 756 iris images. Figure 3.5 represents the sample of iris image CASIA database Version 1.0.

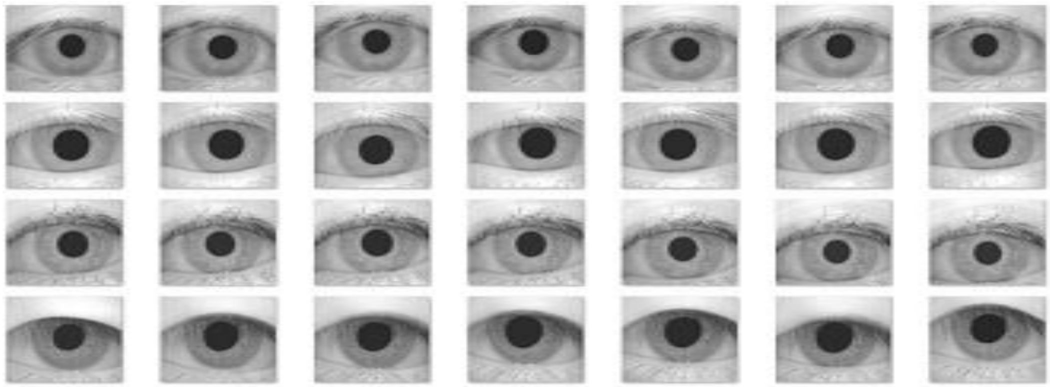


**Figure 3.5:** Sample of Iris Image CASIA Database Version 1.0

Using CASIA close-up iris camera for every eye, There are 7 captured images in 2 sessions. 3 and 4 samples are captured in the first and second session respectively. The format of images is BMP and resolution is 320×280 pixel [49]. These four different data sets which are used in this work make proper opportunity to compare the efficiency of this technique with the eifficiency of the traditional PCA algorithm.

## 3.4 MODELS

In this work, three methods are applied over ORL, Yale, CASIA and Real face dataset to obtain the expected outcomes. These methods are Combined PCA-Segmentation, PCA-Daugman and PCA-SVM which are discussed in the following subsections. All the methods are implemented using MATLAB software and experimented with different dataset to evaluate their performance.

### 3.4.1 Combination of PCA and Segmentation

There are so many algorithms for face recognition which are already very popular, PCA is one of them. In this work, we combined principal component analysis (PCA) with the segmentation method for the training of image datasets. We used three different image dataset to experiment the accuracy of our proposed technique called combined PCA- Segmentation approach for partially covered face recognition. Implementation of this model has been shown at Appendix A-1 to A-4. First of all the selected train dataset images will be stored in a folder. All the train images will be segmented using imcrop for specific location and will be stored in two different train folder as upper face and as lower face. So there are three different train dataset for normal face, upper face and lower face.

In Figure 3.6 a normal face image is segmented into upper and lower face images. These segmented images will also be trained.There are some parts of the working procedure of combined PCA-Segmentation technique such as Pre-processing part, Image training part, Image testing part and Final (Result) part.

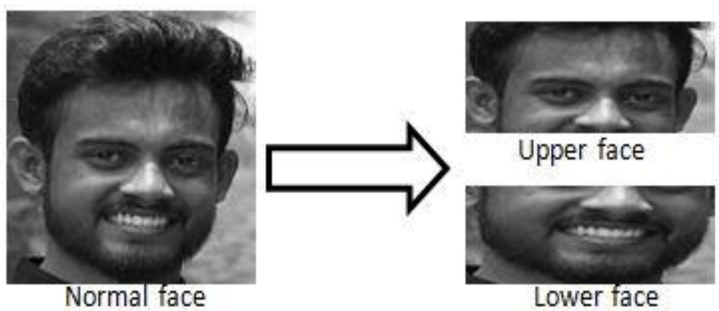



**Figure 3.6:** Segmentation of upper and lower face

In these parts many operations executed like loading, conversion, segmentation, reshaping and resizing along with PCA algorithm. The whole work is done using the MATLAB software. The process of this work of combined PCA-Segmentation is depicted step by step by flowchart in Figure 3.7

***(a) Pre-processing part***

In pre-processing part, the image dataset is being ready to load for the training.

(i) Load image data set: Loading the image data set is the first step to start the work for the purpose of training. Image data will be loaded for perform operations to train data.

(ii) Convert RGB to gray scale images: In this step the RGB images will be converted into gray scale images. Gray scale images will reduce the operation complexities.
(iii) Segment upper region of images: In this step loaded images will be segmented in upper region. After segmenting of upper region of images, the upper segmented images are stored in a new folder for training.
(iv) Segment lower region of images: In this step loaded images will be segmented in lower region. After segmenting of lower region of images, the lower segmented images are stored in another folder for training.
(v) Reshape images: There reshape operation held on the segmented upper and lower region images to bring back the previous original shape.

*(b)* ***Image training part***

In the data training part, the total training process took place on the three folders which contains original full-face images, segmented upper region images and segmented lower region images.

(i) Calculate mean image vector: Calculate the image vector and find the mean image vector by the number of images.
(ii) Subtract mean image vector from 1-D image vector: After calculating mean image vector, subtract the mean image vector from 1-D image vector.
(iii) Calculate a covariance matrix: Calculation of a convari- ance matrix will help to calculate eigen values and eigen vector.
(iv) Calculate eigen values and eigen vectors: Calculate eigen values using the covariance and identity matrix and calcute eigen vectors using the result of eigen values.
(v) Create a set of eigen faces: Create a set of eigen faces using the eigen values and eigen vectors.

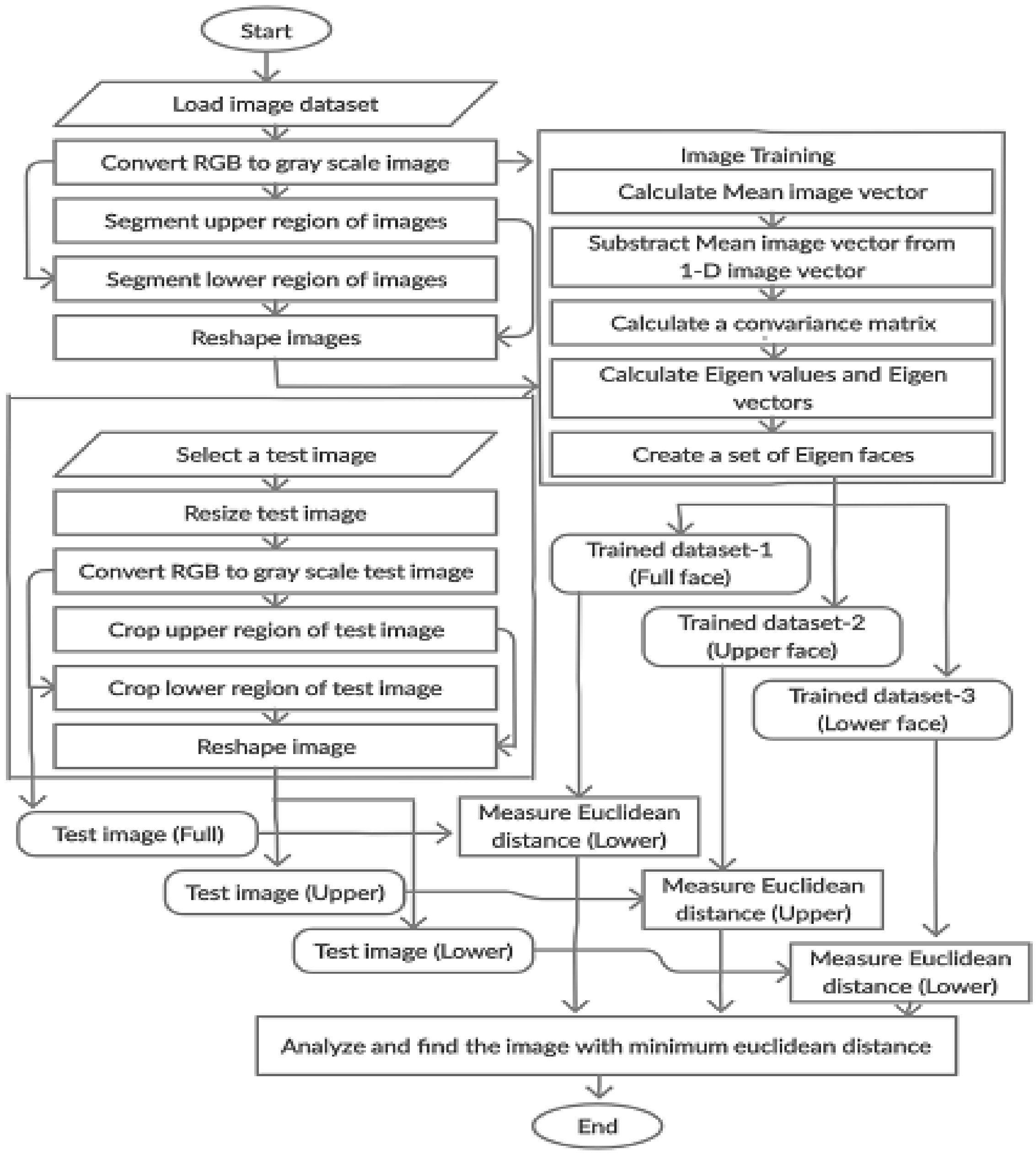



**Figure 3.7:** Flowchart of Combined PCA-Segmentation Technique

***(c) Image testing part***

In the image testing part, the test image will be ready for testing with the training data set.

(i) Select a test image: Selection of a test image is the starting step of image testing part.

(ii) Resize test image: Resizing the test image will make the process easier to compare with training images. It will also ensure consistency of the test process.

(iii) Convert RGB to gray test images: In this step the RGB test image will be converted into gray scale image.
(iv) Crop upper region of test image: After cropping the upper region of test image, the upper cropped test image is stored for further steps.
(v) Crop lower region of test image: After cropping the lower region of test image, the lower cropped test image is also stored for further steps.
(vi) Reshape image: Reshape the cropped upper and lower test images to bring back the previous original shape.

***(d) Final part***

In the final part, the three test images such as full-face test image, cropped upper test image and cropped lower test image will be compared with the three training data set such as original full-face images data set, segmented upper region images data set and segmented lower region images data set respectively. Then calculate the minimum Euclidean distances for the three different testing iteration. After all these steps analyze the three Euclidean distances and names of images and find the best matching image with least Euclidean distance.

### 3.4.2 Combination of PCA and Daugman Technique

The working procedure of combined PCA-Daugman's method are included face recognition and iris recognition. In these parts many operations executed like acquisition, processing, extraction, matching, normalization and combination. Implementation of this model has been shown at Appendix A-5 to A-7.

To overcome all these complexities which include covered face or eyes and to detect emotions of any person, our efficient model is Combined Face-Iris recognition. The motto of this work is to remove the complexities of beard, mustache, hijab and sunglass at facial recognition and emotions detection to increase the accuracy for improving the overall efficiency. Figure 3.8 depicts the flow diagram of combined face and iris recognition.

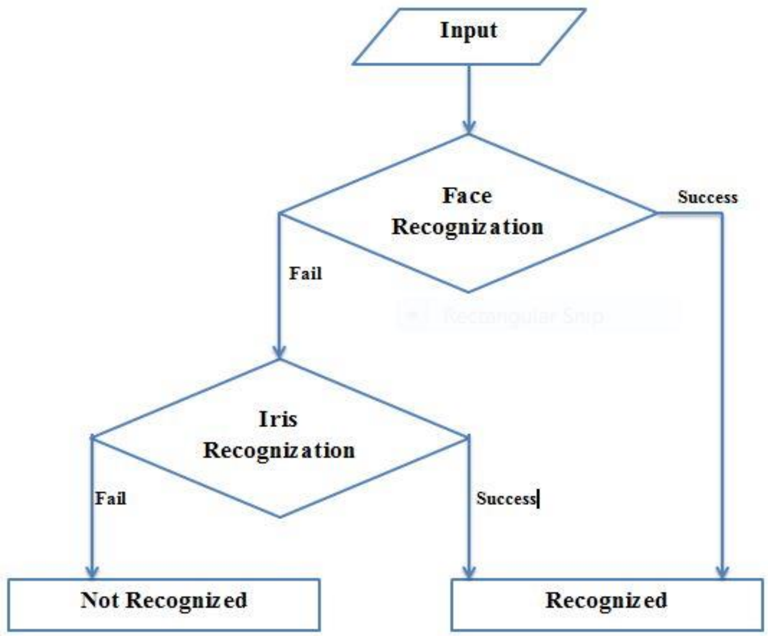



**Figure 3.8:** Flow diagram of Combined Face-Iris Recognition

The process of this work of combined PCA-Daugman method is depicted step by step by flowchart in Figure 3.9.

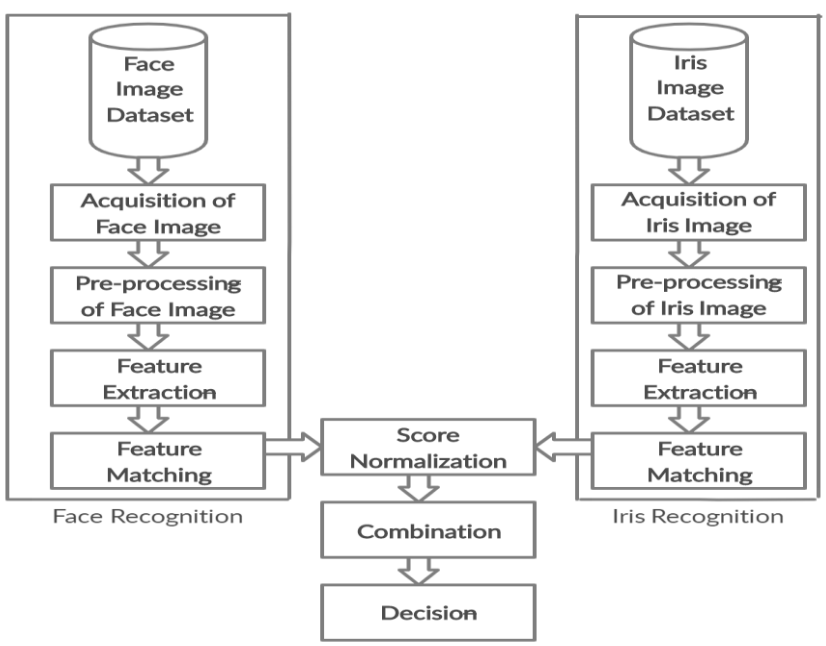



**Figure 3.9:** Face and Iris recognition using Combined PCA-Daugman Method

### *(a) Face Recognition*

Principal Component Analysis (PCA) method for face recognition is used to extract features called 'Eigenface' from face images. Principal Component Analysis has feature vector dimensionality reduction capability by features extraction. Principal Component Analysis is used to the image databases of ORL, YALE and Real face database. The components used in this module are –

(i) Acquisition of Face Image: From database face images are loaded in this step.

(ii) Pre-processing of Face Images: There are some operations in pre-processing stage like scaling, contrast and brightness being executed to make face database images compatible.

(iii) Feature Extraction: Feature extraction is used for creating eigenfaces. Principal Component Analysis commonly used for feature extraction. The steps of feature extraction is described below: Calculate the image vector and find the mean image vector by the number of images. After calculating mean image vector, subtract the mean image vector from 1-D image vector. Calculation of covariance matrix will help to calculate eigen values and eigen vector. Calculate eigen values using the covariance and identity matrix and calculate eigen vectors using the result of eigen values. Create a set of eigen faces using the eigen values and eigen vectors. Calculate the feature Vector for Training image set and select an image for testing.

(iv) Feature Matching: Using the euclidean distance between training and testing feature vectors, the matching score is calculated. Matching scores is used for Combined PCA-Daugman approach of face and iris recognition.

### *(b) Iris Recognition*

Daugman method for iris recognition is used for segmentation process on eye images. Daugman's method is used to the eye image database of CASIA. The components used in iris reccognition are -

(i) Acquisition of Iris Image: From CASIA database iris images are loaded in this phase.

(ii) Pre-processing of Iris Images: Iris localization and normalization are used for iris image pre-processing. Normalized iris is used for the feature extraction.

(iii) Feature Extraction: From normalized iris pattern features are encoded using 1D Log-Gabor Wavelets. Features are stored as masks and templates.

(iv) Feature Matching: Hamming distance is used for the calculation of matching score. Matching scores used for Combined PCA-Daugman approach of face and iris recognition.

***(c) Face and Iris Recognition***

Using the combined PCA-Daugman method, face and iris recognition works together efficiently. In this phase, the previous scores are used. The components of this module are –

(i) Score normalization: Feature extraction in face recognition using PCA and euclidean distance is used for calculation of matching score. Daugman's method is used for iris recognition feature extraction and hamming distance is used to find matching score. In this stage, both matching score is used to normalize the score using min max normalization technique with common domain and range.

(ii) Combination: Score level fusion technique for face and iris recognition is used.

(iii) Decision: Combined score of face and iris final decision have taken place to identify a person.

### 3.4.3 Combination of PCA and SVM

Principal Component Analysis and Support Vector Machine are working together to define covered face like face with mask and then identify the person efficiently.

In Figure 3.10 depicts the combination of principal component analysis and support vector machine. Support Vector Machine is applied on selected test image to find out whether the image has covered face or not and Principal Component Analysis (PCA) method is applied to the image for face recognition is used to extract features called 'Eigenface' from face images belongs to the related path based on the result of support vector machine. It increases the accuracy and efficiency of face recognition for covered face images.

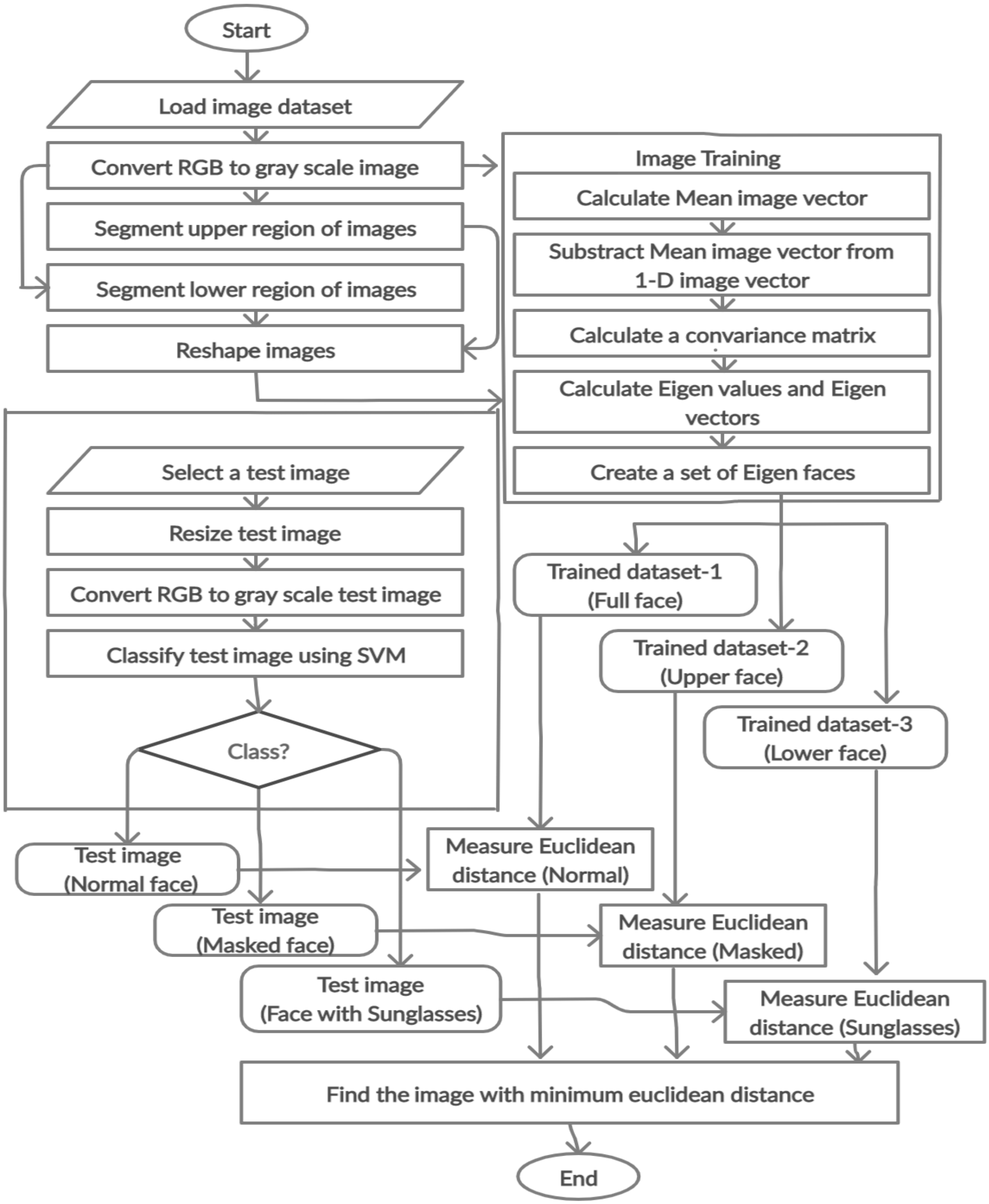



**Figure 3.10:** Combination of PCA-SVM Method

## 3.5 ACCURACY CALCULATION

Accuracy is calculated in this work for three different techniques over different image datasets. Accuracy represents the efficiency and performance of the techniques for covered face and eyes. To calculate the accuracy, there are some parameters needed such as- the total number of epochs for test image execution along with true positive, true negative, false positive and false negative cases for face recognition. For the accuracy calculation in percentage the following formula is used-

$$\frac{TNT-(FP+FN)}{TNT} \times 100 \tag{3.13}$$

Where,

TNT = Total Number of Test images

TNT = TP + TN + FP + FN

TP = Number of True Positive cases

TN = Number of True Negative cases

FP = Number of False Positive cases

FN = Number of False Negative cases

This formula is used to calculate the accuracy of different techniques for different image datasets which include full face, face with mask and face with sunglasses images. The accuracy of proposed combined techniques are being used to compare with the accuracy of the previous solo techniques in the result section for covered face and eyes.

## 3.6 REQUIREMENTS

In this work, there are some hardwires required such as computer for overall work, calculator for calculations and camera / webcam for image collection. Most important software for this work is MATLAB software. It is used for the implementations and execution of the methods. Notepad++ is used for note something and Microsoft Excel is used to produce graphs for the calculated percentages from the gathered data or results. For this work MATLAB language is used for programming. In this work, the image datasets which are used ORL, Yale, CASIA and Real Face dataset. Principal Component Analysis, Daugman's Algorithm and Support Vector Machine are the required algorithms for this work. We have combined PCA and Segmentation, PCA and Daugman's algorithm as well as PCA and SVM technique. To evaluate performance of these techniques we used different datasets which comprises with full face, covered face and eyes.

## 3.7 SUMMARY

This methodology chapter have discussued about the algorithms and basic terminologies which are used in this work such as Principal Component Analysis, Daugman's Algorithm, Support Vector Machine, Segmentation and Euclidean distance. In this chapter also have included used datasets such as ORL, Yale, CASIA and Real dataset. Most importantly the methods called Combined PCA-Segmentation, PCA-Daugman and PCA-SVM are discussed in this this along with the requirements of this work.

# CHAPTER 4

# RESULTS AND DISCUSSION

## 4.1 INTRODUCTION

In this chapter the results of the study are presented and discussed with reference to the aim of the study, which was to overcome the covered face and eyes recognition complexities and to calculate the performance of proposed methods. There recognition performance was evaluated and compared for the three methods called combined PCA-Segmentation, PCA-Daugman and PCA-SVM over different dataset such as ORL, Yale, CASIA and Real face dataset. These aspects were described in the previous chapter that presented the methodology used in the study.

## 4.2 PRINCIPAL COMPONENT ANALYSIS ON IMAGES

Principle Component Analysis algorithm (PCA) is applied for 10 trained images which are 128*128 pixels.


**Figure 4.1:** Selected 10 images for training using PCA

Figure 4.1 shows the selected 10 images for training using PCA. As the size of images are 128*128 pixel so the matix size for single image is 16384 by 1, for the 10 images the matrix size is 16384 by 10, Where 16384 is the number of total pixels and 10 is the number of total images. İmage matrix is shown in Table 4.1

**Table 4.1:** Image matrix for 10 images with 128*128 pixels

| Images / Pixels | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 3 | 13 | 0 | 18 | 16 | 0 | 0 | 0 | 0 | 0 |
| **2** | 0 | 9 | 0 | 11 | 18 | 0 | 0 | 0 | 0 | 0 |
| **3** | 0 | 24 | 1 | 0 | 17 | 0 | 0 | 0 | 0 | 0 |
| **.** | . | . | . | . | . | . | . | . | . | . |
| **.** | . | . | . | . | . | . | . | . | . | . |
| **.** | . | . | . | . | . | . | . | . | . | . |
| **16382** | 14 | 3 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 0 |
| **16383** | 13 | 1 | 0 | 0 | 6 | 22 | 0 | 0 | 0 | 0 |
| **16384** | 5 | 0 | 0 | 0 | 0 | 95 | 0 | 0 | 0 | 0 |

The covariance matrix for 10 images is the matrix of 10 by 10. Table 4.2 shows the covariance square matrix for the selected 10 images.

**Table 4.2:** Covariance matrix for 10 images

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 6512887 | 4344949 | 4021541 | 4248009 | 3041285 | 600164 | 693882 | 534009 | 640730 | 833365 |
| **2** | 4344949 | 22406353 | 6129745 | 7833805 | 12600991 | 792291 | 1293350 | 824645 | 1267719 | 1877845 |
| **3** | 4021541 | 6129745 | 7744897 | 5210614 | 4706597 | 473748 | 677989 | 620096 | 778856 | 621921 |
| **4** | 4248009 | 7833805 | 5210614 | 21632152 | 11500617 | 1807171 | 471568 | 958492 | 392460 | 992993 |
| **5** | 3041285 | 12600991 | 4706597 | 11500617 | 17142180 | 791038 | 522287 | 557834 | 464237 | 950277 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **6** | 600164 | 792291 | 473748 | 1807171 | 791038 | 1831389 | 270443 | 518274 | 239169 | 497557 |
| **7** | 693882 | 1293350 | 677989 | 471568 | 522287 | 270443 | 1325027 | 376538 | 846419 | 441275 |
| **8** | 534009 | 824645 | 620096 | 958492 | 557834 | 518274 | 376538 | 1328571 | 278436 | 333181 |
| **9** | 640730 | 1267719 | 778856 | 392460 | 464237 | 239169 | 846419 | 278436 | 1232072 | 353537 |
| **10** | 833365 | 1877845 | 621921 | 992993 | 950277 | 497557 | 441275 | 333181 | 353537 | 1928043 |

[V,D] = eig(A) returns diagonal matrix D of eigenvalues and matrix V whose columns are the corresponding right eigenvectors, so that A*V = V*D where A is the covariance matrix. Table 4.3 shows the Eigen vector matrix for the selected 10 images.

**Table 4.3:** Eigen vector matrix for 10 images

| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0.012335 | -0.00857 | -0.06038 | -0.00585 | -0.36118 | 0.642263 | 0.211165 | 0.610071 | -0.00578 | 0.190615 |
| **2** | -0.00342 | -0.00172 | -0.05199 | 0.021947 | -0.12772 | -0.06348 | -0.46017 | -0.02705 | 0.660342 | 0.57264 |
| **3** | -0.02821 | 0.030688 | 0.061397 | 0.097403 | 0.173136 | -0.64299 | 0.375642 | 0.578692 | 0.039174 | 0.253123 |
| **4** | 0.002242 | -0.01162 | 0.037962 | -0.07382 | -0.07144 | -0.08176 | -0.38276 | 0.006113 | -0.74333 | 0.531129 |
| **5** | 0.005469 | 0.003667 | -0.00309 | -0.00422 | 0.093735 | 0.16056 | 0.655064 | -0.50596 | 0.010273 | 0.529294 |
| **6** | -0.02953 | 0.466399 | -0.4538 | 0.556642 | 0.45638 | 0.191757 | -0.10708 | 0.057279 | -0.06065 | 0.048328 |
| **7** | -0.69609 | 0.090138 | -0.06076 | -0.5553 | 0.401601 | 0.136785 | -0.04204 | 0.101147 | 0.043845 | 0.037043 |
| **8** | 0.085229 | -0.85027 | -0.33484 | 0.095054 | 0.36242 | 0.092078 | -0.04227 | 0.075038 | -0.00989 | 0.03593 |
| **9** | 0.710845 | 0.213967 | -0.07184 | -0.53936 | 0.361738 | 0.081219 | -0.03062 | 0.105055 | 0.046755 | 0.035167 |
| **10** | 0.032142 | -0.06669 | 0.813343 | 0.257631 | 0.421432 | 0.258873 | -0.10473 | 0.077353 | 0.043052 | 0.056502 |

**Table 4.4:** Diagonal matrix for 10 images

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 421574.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **2** | 0 | 955798 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | 0 | 0 | 1356851 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 0 | 0 | 0 | 1510799 | 0 | 0 | 0 | 0 | 0 | 0 |
| **5** | 0 | 0 | 0 | 0 | 2492591 | 0 | 0 | 0 | 0 | 0 |
| **6** | 0 | 0 | 0 | 0 | 0 | 3097299 | 0 | 0 | 0 | 0 |
| **7** | 0 | 0 | 0 | 0 | 0 | 0 | 4877087 | 0 | 0 | 0 |
| **8** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8108288 | 0 | 0 |
| **9** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 14322624 | 0 |
| **10** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 45940658 |

Table 4.4 shows the Diagonal matrix which is calculated from eigen function for the selected 10 images. Eigenfaces for 10 images is the matrix of 16384 by 10. Eigenfaces calculated from the Eigen vector matrix. Table 4.5 shows the Eigenface matrix for the selected 10 images.

**Table 4.5:** Eigenfaces for 10 images

| Images / Pixels | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0.120358 | -0.19866 | -0.22322 | -1.12855 | -2.53001 | 2.198757 | -1.75746 | -6.50678 | -4.64845 | 26.04519 |
| **2** | 0.092294 | -0.07736 | -0.10602 | -0.69045 | -0.24807 | 1.419352 | 3.439202 | -9.28352 | -2.04862 | 20.52347 |
| **3** | -0.01739 | 0.051644 | -1.23901 | 0.552437 | -1.29858 | 0.562909 | 0.467578 | -8.67182 | 16.06202 | 22.99448 |
| **.** | . | . | . | . | . | . | . | . | . | . |
| **.** | . | . | . | . | . | . | . | . | . | . |
| **.** | . | . | . | . | . | . | . | . | . | . |
| **16382** | 0.206173 | -0.09584 | -1.02604 | -0.04984 | -4.68973 | 10.08571 | 6.816304 | 4.412151 | 1.98226 | 8.620879 |
| **16383** | -0.45985 | 10.16962 | -10.8391 | 12.16668 | 5.779761 | 13.46795 | 3.85951 | 6.128245 | -0.68748 | 7.289612 |
| **16384** | -2.74341 | 44.26503 | -43.4129 | 52.85175 | 41.5502 | 21.42823 | -9.11715 | 8.491886 | -5.79064 | 5.544231 |

Training vectors are calculated from the Eigenfaces. After calculating all training vector the training vector matrix is 10 by 10. Table 4.6 shows the training vector matrix for the selected 10 images.

**Table 4.6:** Training vector matrix for 10 images

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0.012335 | -0.00342 | -0.02821 | 0.002242 | 0.005469 | -0.02953 | -0.69609 | 0.085229 | 0.710845 | 0.032142 |
| **2** | -0.00857 | -0.00172 | 0.030688 | -0.01162 | 0.003667 | 0.466399 | 0.090138 | -0.85027 | 0.213967 | -0.06669 |
| **3** | -0.06038 | -0.05199 | 0.061397 | 0.037962 | -0.00309 | -0.4538 | -0.06076 | -0.33484 | -0.07184 | 0.813343 |
| **4** | -0.00585 | 0.021947 | 0.097403 | -0.07382 | -0.00422 | 0.556642 | -0.5553 | 0.095054 | -0.53936 | 0.257631 |
| **5** | -0.36118 | -0.12772 | 0.173136 | -0.07144 | 0.093735 | 0.45638 | 0.401601 | 0.36242 | 0.361738 | 0.421432 |
| **6** | 0.642263 | -0.06348 | -0.64299 | -0.08176 | 0.16056 | 0.191757 | 0.136785 | 0.092078 | 0.081219 | 0.258873 |
| **7** | 0.211165 | -0.46017 | 0.375642 | -0.38276 | 0.655064 | -0.10708 | -0.04204 | -0.04227 | -0.03062 | -0.10473 |
| **8** | 0.610071 | -0.02705 | 0.578692 | 0.006113 | -0.50596 | 0.057279 | 0.101147 | 0.075038 | 0.105055 | 0.077353 |
| **9** | -0.00578 | 0.660342 | 0.039174 | -0.74333 | 0.010273 | -0.06065 | 0.043845 | -0.00989 | 0.046755 | 0.043052 |
| **10** | 0.190615 | 0.57264 | 0.253123 | 0.531129 | 0.529294 | 0.048328 | 0.037043 | 0.03593 | 0.035167 | 0.056502 |

Now, selected testing image has the testing vector matrix of 10 by 1. Table 4.7 shows the testing vector matrix for test images.

**Table 4.7:** Testing vector matrix for test image

| | 1 |
|---|---|
| **1** | 0.012335 |
| **2** | -0.00857 |
| **3** | -0.06038 |
| **4** | -0.00585 |
| **5** | -0.36118 |
| **6** | 0.642263 |
| **7** | 0.211165 |
| **8** | 0.610071 |
| **9** | -0.00578 |
| **10** | 0.190615 |

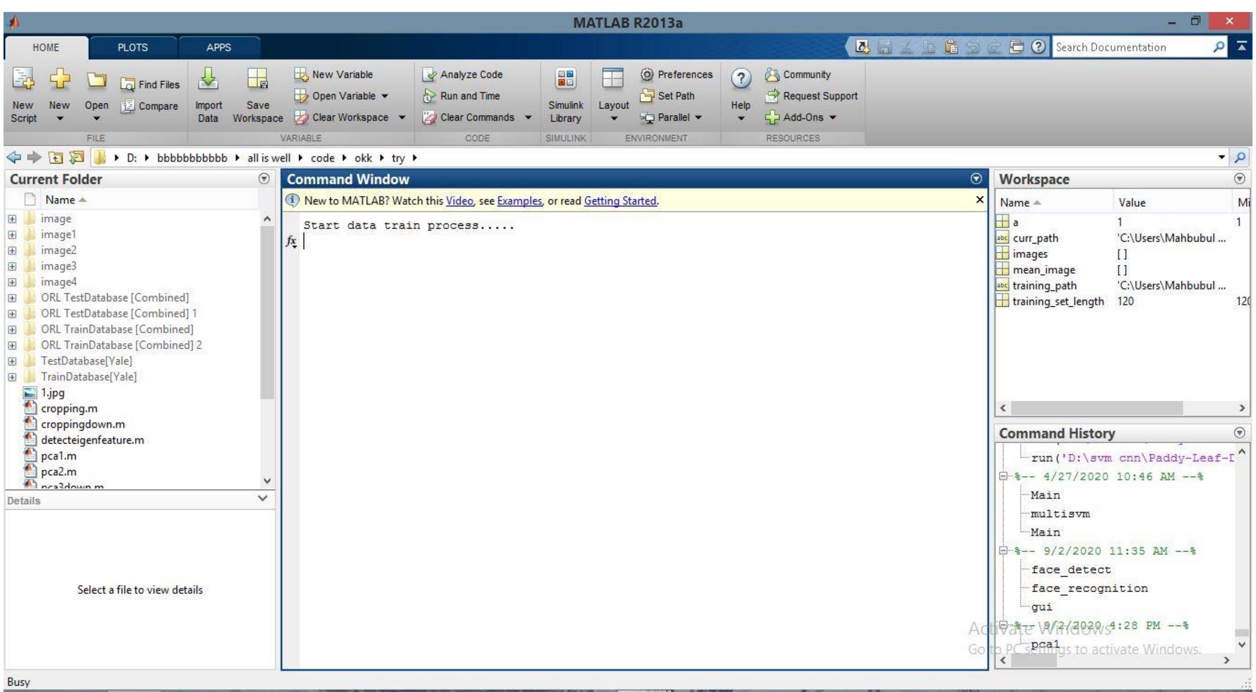
From these training and testing vector matrix the calculating eucledian distances are:

| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|---|---|---|---|
| **0.767224** | 1.239927 | 1.11913 | 0.960521 | 0.80316 | 1.235389 | 1.096893 | 1.199368 | 1.219357 | 1.152709 |

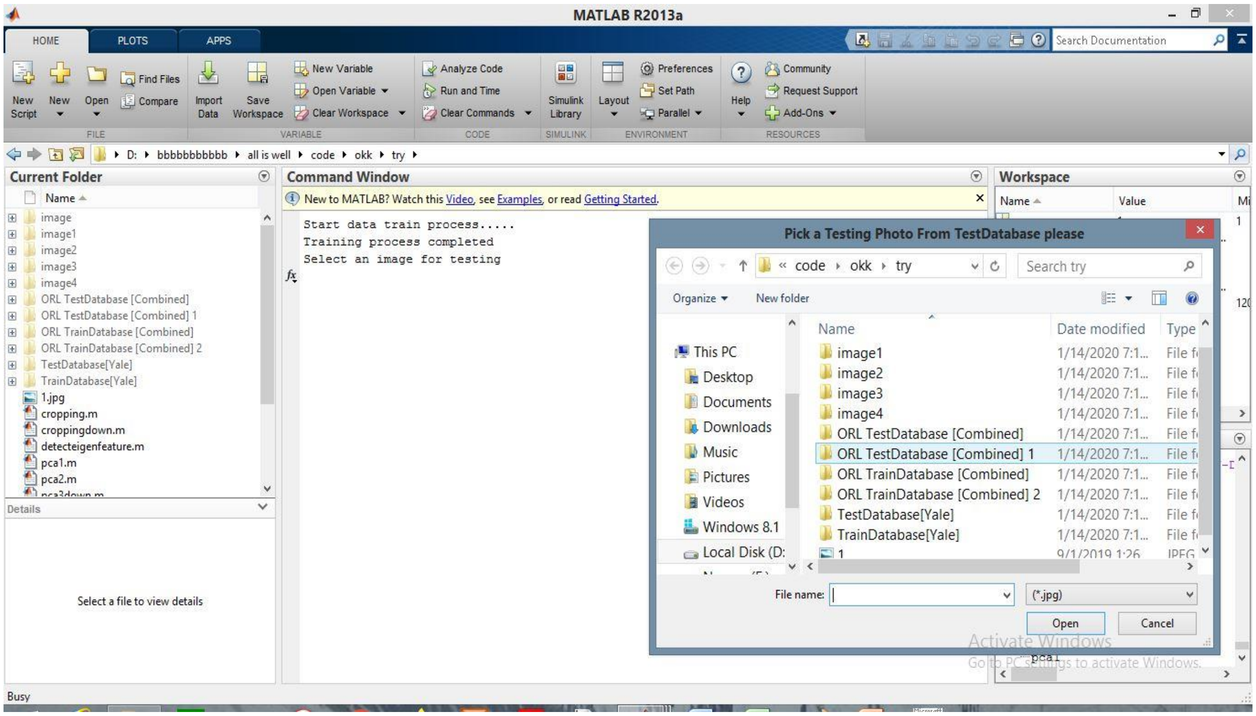
From all these eucledian distances the minimum eucledian distance is **0.767224** which indicates the image number 1. So the recognized image id is 1 from the training images.

## 4.3 COMBINATION OF PCA AND SEGMENTATION

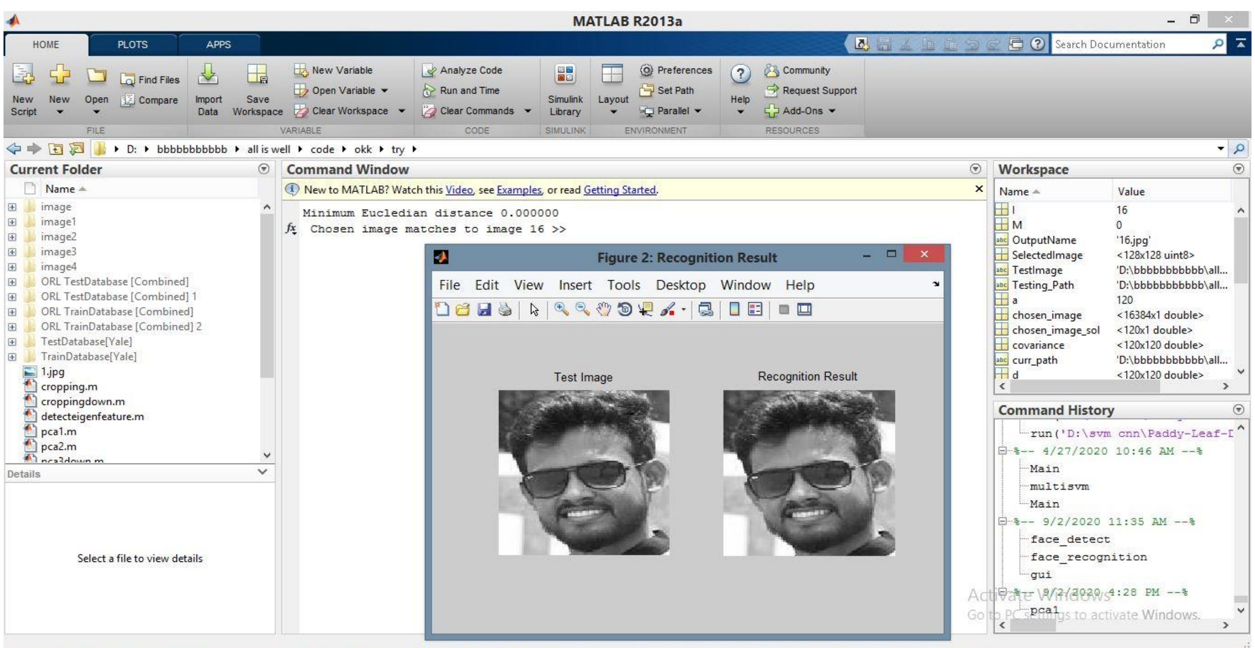
Using the combined PCA-Segmentation technique, the accuracy of face recognition for different datasets are different but obviously better than the accuracy by using the PCA algorithm. The accuracy is calculated by several epochs for random test images. In Figure 4.1 shown the output interface for face recognition. Figure 4.2 (a) shown that the train process is ongoing for the selected dataset, Figure 4.2 (b) shown that after completion of data training commant window gives a message to select any test image and Figure 4.2 (c) shown the recognized image number with the minimun euclidean distance for chosen image.




(a)




(b)




(c)

**Figure 4.2:** Some snapshots of output interface

### 4.3.1 Combined PCA-Segmentation on ORL Dataset

The performance of face recognition for ORL dataset is calculated on random choice test images and the recognition performance of PCA and Combined PCA-Segmentation approach are shown in Table 4.8.

**Table 4.8:** Face recognition performance for ORL database

| Methods | TP | TN | FP | FN | Accuracy (%) |
|---|---|---|---|---|---|
| Using PCA | 130 | 8 | 40 | 12 | 72.63 |
| Using Combined PCA-Segmentation method | 171 | 19 | 0 | 1 | 99.47 |

From Table 4.8 it is very clear that face recognition accuracy using PCA algorithm on ORL dataset is less than the Combined PCA-Segmentation method. Figure 4.3 depicts the graphical representation of the performance accuracy for ORL dataset using PCA method and Combined PCA-Segmentation method.

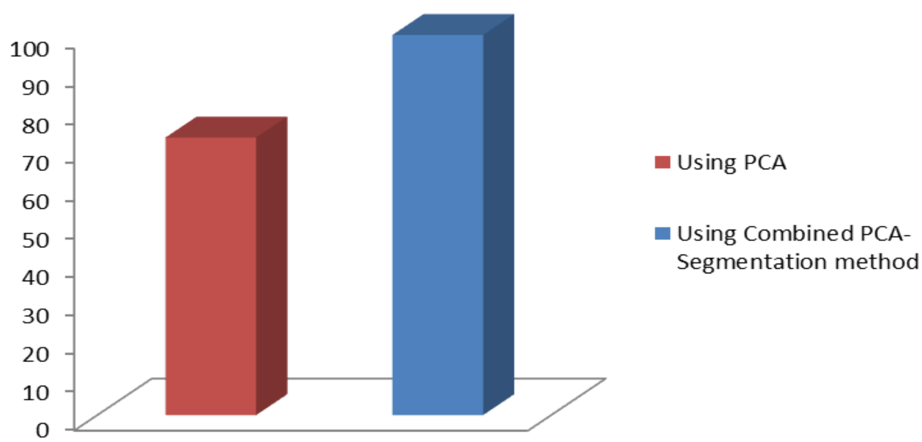



**Figure 4.3:** Graphical representation of performance for ORL database

The accuracy for ORL dataset using Combined PCA-Segmentation method is much better than solo PCA method. The reason behind it that combined method used three different train dataset included segmented dataset called upper face and lower face dataset and calculate minimum Euclidean distance for each dataset. That is why combined PCA and Segmentation method provided better accuracy for ORL dataset. Figure 4.4 depicts number of epochs performed in this method versus minimum Euclidean distance for each test image. As 190 epoch were executed over ORL dataset, minimum Euclidean distance were graphed for each 190 epoch.

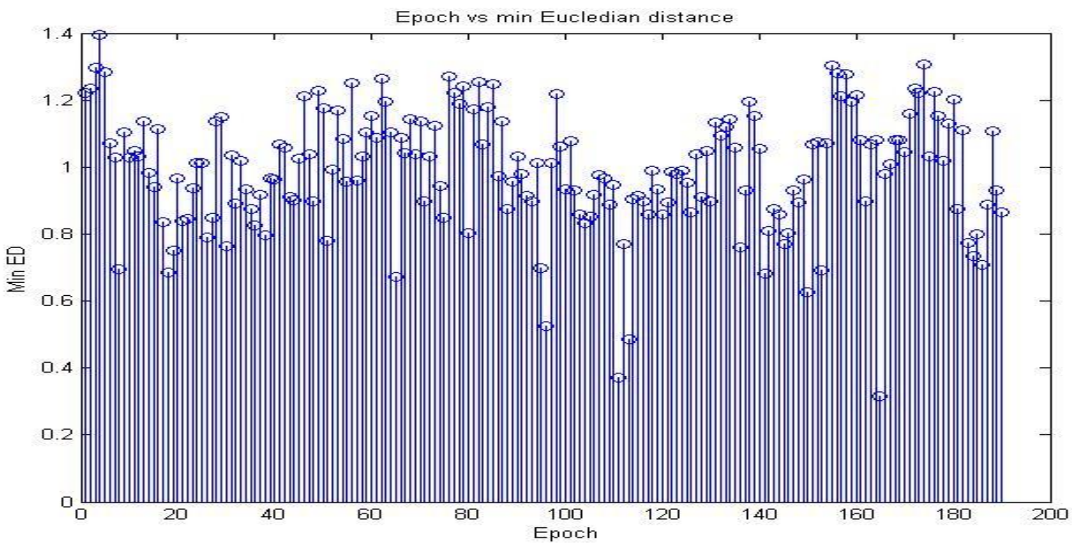



**Figure 4.4:** Epoch vs min eucledian distance for orl dataset

### 4.3.2 Combined PCA-Segmentation on Yale Dataset

We have calculated the performance for Yale dataset using the PCA and Combined PCA-Segmentation approach. Face recognition performance of PCA and Combined PCA-Segmentation approach are shown in Table 4.9.

**Table 4.9:** Face recognition performance for Yale database

| Methods | TP | TN | FP | FN | Accuracy (%) |
|---|---|---|---|---|---|
| Using PCA | 32 | 7 | 4 | 2 | 86.67 |
| Using Combined PCA-Segmentation method | 35 | 9 | 1 | 0 | 97.78 |

From Table 4.9 the finding is that face recognition accuracy on Yale dataset is far better for Combined PCA-Segmentation method than the PCA algorithm. Figure 4.5 depicts the graphical representation of the performance accuracy for Yale dataset using PCA method and Combined PCA-Segmentation method.

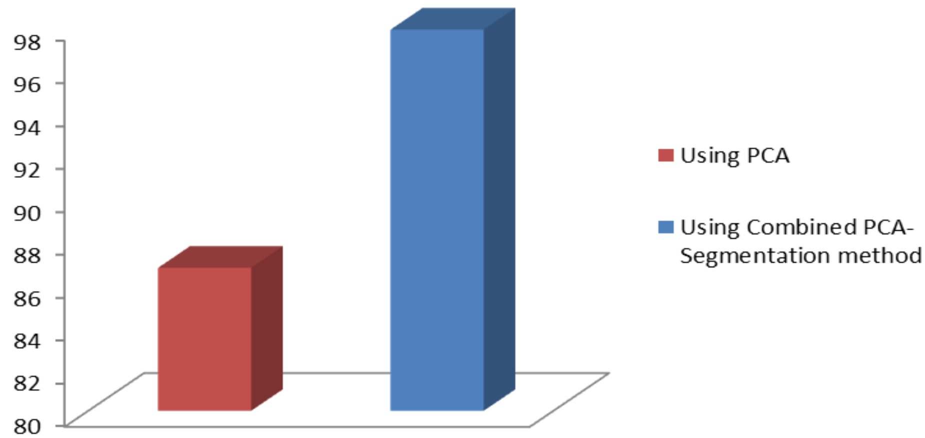



**Figure 4.5:** Graphical representation of performance for Yale database

The accuracy for Yale dataset using Combined PCA-Segmentation method is better than solo PCA method because of combined method used three different train dataset included segmented dataset. Figure 4.6 depicts number of epochs performed in this method versus minimum Euclidean distance for each test image. As 45 epochs were executed over yale dataset, minimum Euclidean distance were graphed for each 45 epochs.

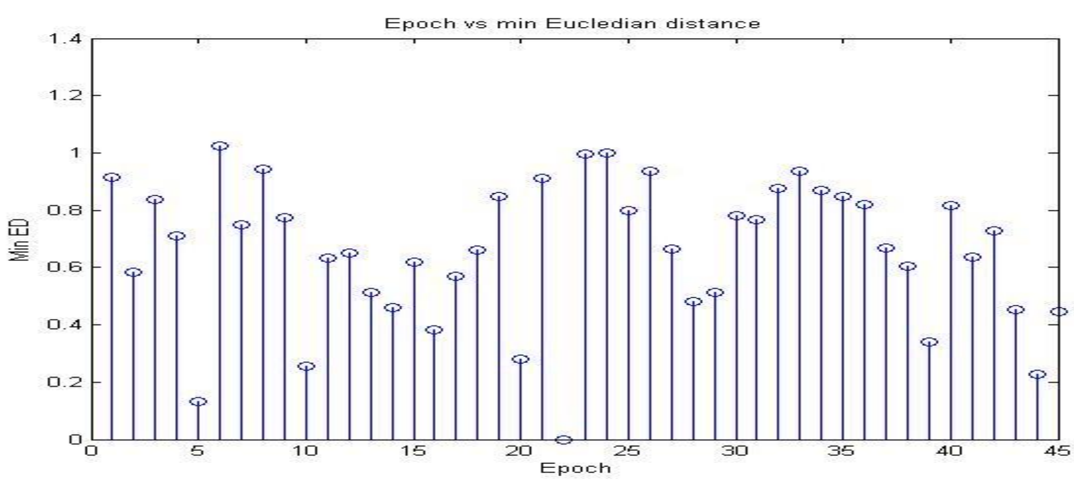



**Figure 4.6:** Epoch vs min eucledian distance for yale dataset

### 4.3.3 Combined PCA-Segmentation on Real Face Dataset

The real face dataset consists of real images to check the performance of this approach. We choice random images from dataset to find the least Euclidean distance image from trained dataset. The face recognition performance of PCA and Combined PCA-Segmentation approach are shown in Table 4.10.

**Table 4.10:** Face recognition performance for Real Face database

| Methods | TP | TN | FP | FN | Accuracy (%) |
|---|---|---|---|---|---|
| Using PCA | 29 | 3 | 8 | 6 | 69.57 |
| Using Combined PCA-Segmentation method | 36 | 8 | 1 | 1 | 95.65 |

From Table 4.10, face recognition accuracy Using PCA algorithm on Real face dataset clearly is less than the Combined PCA-Segmentation method. Figure 4.7 depicts the graphical representation of the performance accuracy for Yale dataset using PCA method and Combined PCA-Segmentation method.

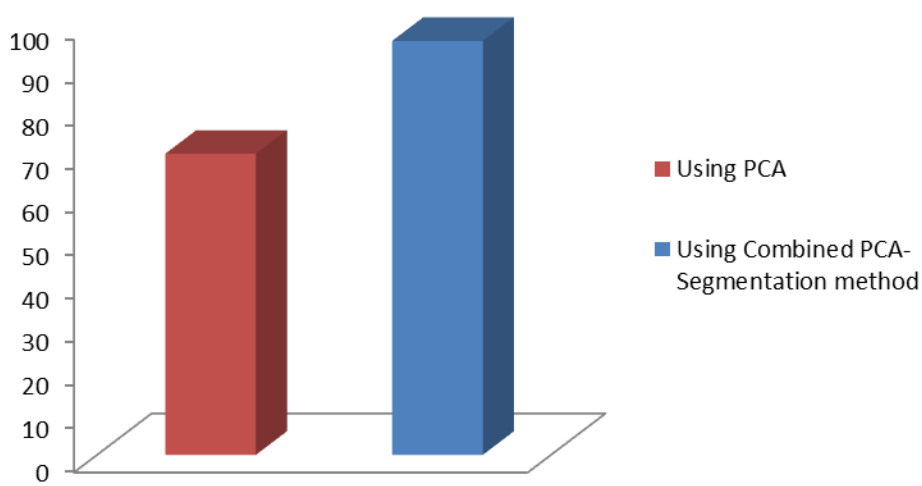



**Figure 4.7:** Graphical representation of performance for Real Face database

The accuracy for Real face dataset using Combined PCA-Segmentation method is better than solo PCA method because of combined method used three different train dataset included segmented dataset.

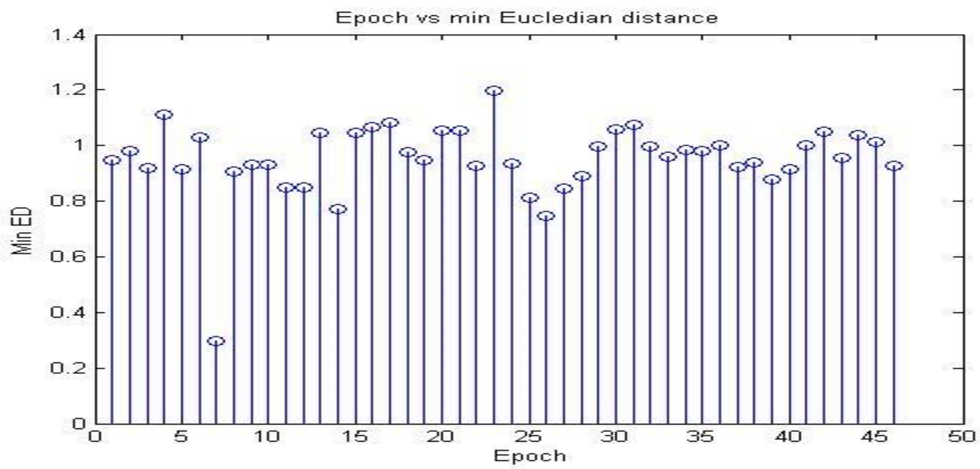



**Figure 4.8:** Epoch vs min eucledian distance for real face dataset

Figure 4.8 depicts number of epochs performed in this method versus minimum Euclidean distance for each test image. As 46 epochs were executed over real face dataset, minimum Euclidean distance were graphed for each 46 epochs. This subsection shows that the face recognition performance using the combined PCA Segmentation approach is better than the traditional PCA method.

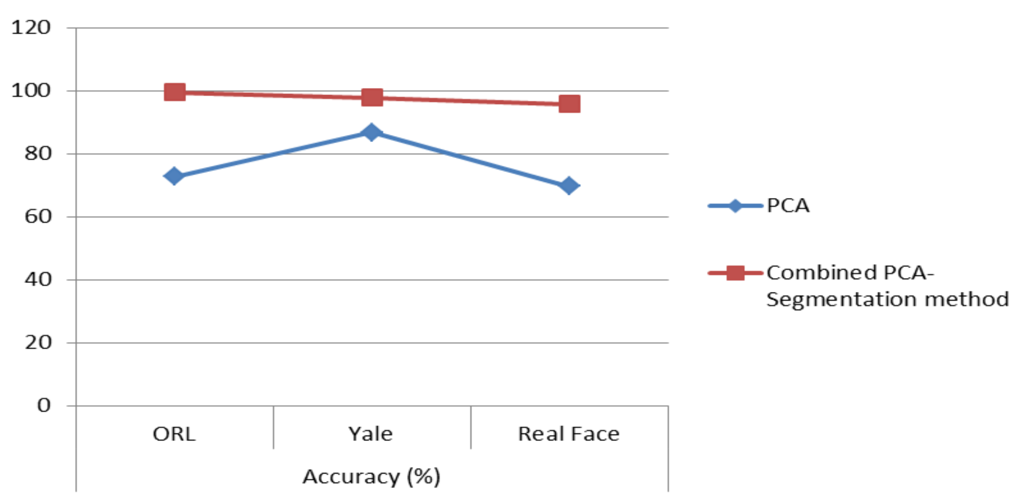



**Figure 4.9:** Comparison of performance for various database

Figure 4.9 showed the comparison of performance accuracy between solo PCA and PCA-Segmentation method for three datasets. It also showed that performance accuracy varies from dataset to dataset because each dataset has different standard and parameters to capture images. Actually, performance accuracy is higher than traditional one because of two reason, one is combination of PCA and Segmentation method and another is three different image datasets included segmented datasets.

We experiment the combined PCA-Segmentation approach on three different image datasets like ORL, Yale and Real face dataset which contains real images along with PCA algorithm. Accuracy of this method is very satisfactory over PCA algorithm for covered face and eyes situations. It is also good for the sun glass and scarf user. From the result it is clearly shown that this research methodology works well. So, the Combined PCA-Segmentation method is efficient for removing the complexities of covered eyes and faces recognition.

## 4.4 COMBINATION OF PCA AND DAUGMAN

Using the combined PCA-Daugman technique, the accuracy of multimodal face and iris recognition system for different datasets are different but obviously better than the accuracy of unimodal recognition using the PCA or Daugman algorithm separately. The accuracy is calculated by 45 epochs for random test images.

### 4.4.1 Combined PCA-Daugman on ORL and CASIA Dataset

The performance of face recognition for ORL database, iris recognition for CASIA database and both database for combined face and iris recognition is calculated on random choice test images. The recognition performance for ORL and CASIA database are shown in Table 4.11.

**Table 4.11:** Face and Iris recognition performance for ORL and CASIA database

| **Methods** | **Accuracy (%)** |
|---|---|
| PCA (ORL Database) | 72.63 |
| Daugman method (CASIA Database) | 88.89 |
| Combined PCA-Daugman method (ORL-CASIA Database) | 97.78 |

From Table 4.11 it is very clear that face and iris recognition accuracy Using Combined PCA-Daugman method is more efficient for ORL and CASIA database. Figure 4.10 represents the performance accuracy graphically.

The accuracy for ORL and CASIA dataset using Combined PCA-Daugman method is better than solo PCA or solo Daugman method because of combined method used two different train datasets like face dataset and iris dataset included related template and mask.

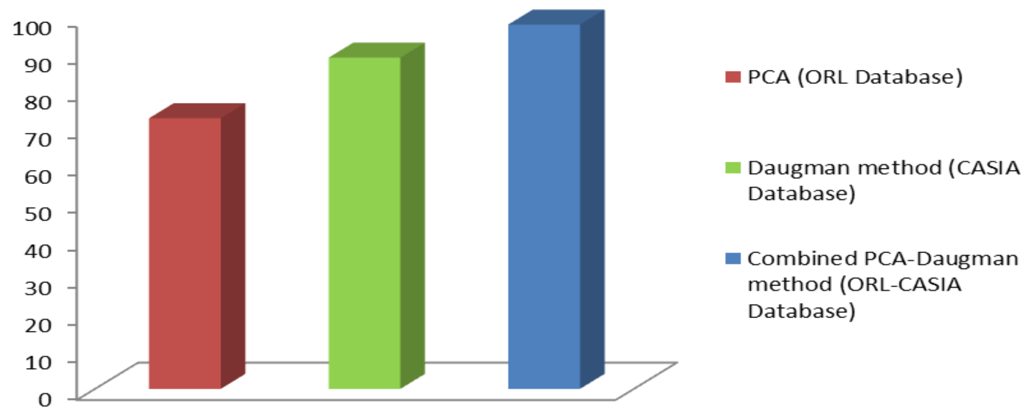



**Figure 4.10:** Graphical representation of performance for ORL and CASIA database

### 4.4.2 Combined PCA-Daugman on Yale and CASIA Dataset

We have calculated the performance for Yale dataset using the PCA, Daugman and Combined PCA-Daugman approach. The recognition performance for Yale and CASIA database are shown in Table 4.12.

**Table 4.12:** Face and Iris recognition performance for Yale and CASIA database

| Methods | Accuracy (%) |
|---|---|
| PCA (Yale Database) | 86.67 |
| Daugman method (CASIA Database) | 88.89 |
| Combined PCA-Daugman method (Yale-CASIA Database) | 99.99 |

Table 4.12 showed that the performance on Yale and CASIA database is far better for Combined PCA-Daugman method than the unimodal PCA and Daugman algorithm. Figure 4.11 depicts the graphical representation of the performance accuracy for Yale dataset using PCA method, CASIA dataset using Daugman method and Combined PCA-Daugman method.

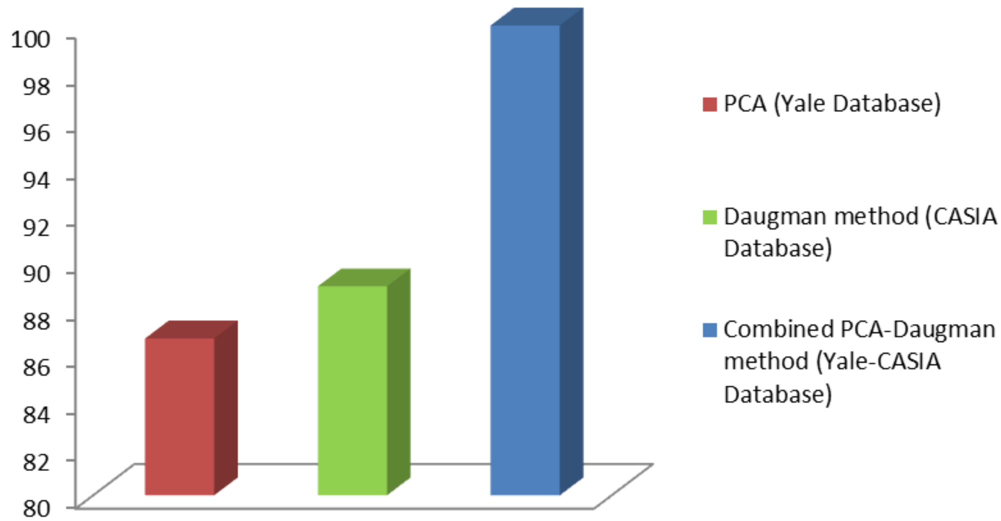



**Figure 4.11:** Graphical representation of performance for Yale and CASIA database

### 4.4.3 Combined PCA-Daugman on Real Face and CASIA Dataset

The real face database consists of real images to check the performance of this approach. The recognition performance of PCA, Daugman and Combined PCA-Daugman approach for Real face Database and CASIA Database are shown in Table 4.13.

**Table 4.13:** Face and Iris recognition performance for Real Face and CASIA database

| Methods | Accuracy (%) |
|---|---|
| PCA (Real Face Database) | 69.57 |
| Daugman method (CASIA Database) | 88.89 |
| Combined PCA-Daugman method (Real Face-CASIA Database) | 95.56 |

Table 4.13 showed that recognition accuracy Using PCA algorithm and Daugman method on Real face and CASIA database clearly is less than the Combined PCA-Daugman method. Figure 4.12 depicts the graphical representation of the performance accuracy for Real dataset using PCA method, CASIA dataset using Daugman method and Combined PCA-Daugman method.

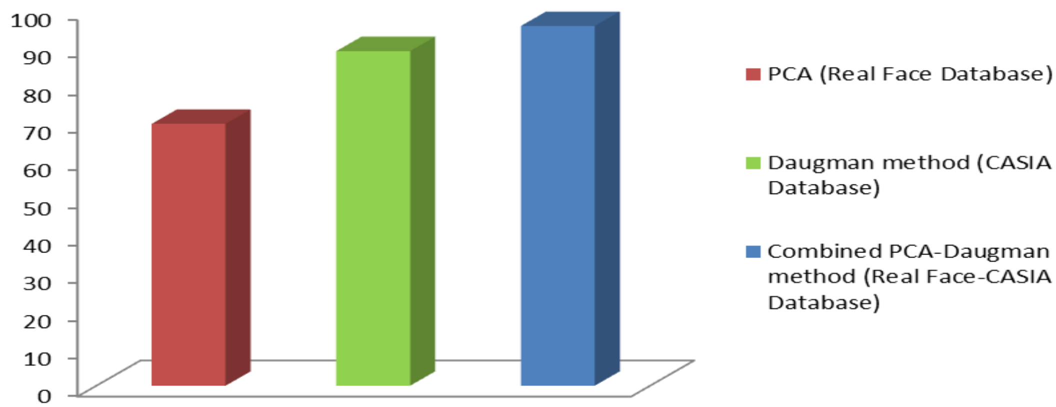



**Figure 4.12:** Graphical Representation of performance for Real Face and CASIA database

The performance of combined PCA-Daugman method is better than the solo PCA and solo Daugman method. We have experimented the combined PCA-Daugman method on three different image database like ORL, Yale, Real face and CASIA database. Figure 4.13 depicts the comparison of performance accuracy between solo PCA, solo Daugman and Combined PCA-Daugman method for three different datasets.

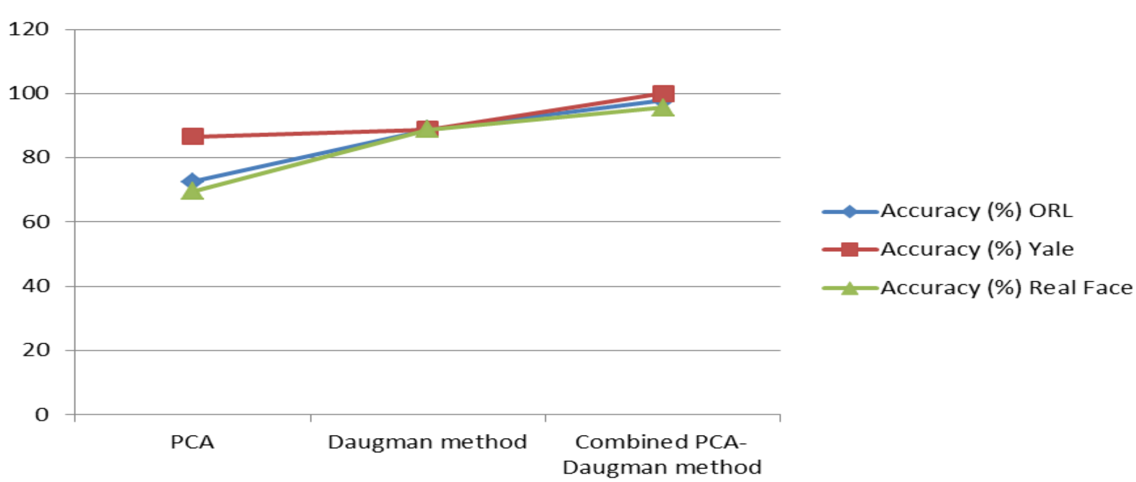



**Figure 4.13:** Comparison of performance for various database

The reason behind the variation in accuracy for different datasets is that different database images have different characteristics, position, lighting, quality, resolution and focus. Accuracy of this method is very satisfactory over face recognition and iris recognition separately to identify a person. From the result it is clearly shown that this research methodology works well. An efficient multimodal recognition system is Combined PCA-Daugman method.

## 4.5 COMBINATION OF PCA AND SVM

Using the combined PCA-SVM technique, the accuracy of face recognition for real face dataset better than the accuracy by using the PCA algorithm. The accuracy is calculated by several epochs for random test images using the real face train dataset.

The performance of face recognition for Real Face dataset is calculated on random choice test images and the recognition performance of PCA and Combined PCA-SVM approach are shown in Table 4.14. The real face dataset consists of real images with mask face and face with sunglasses to check the performance of this approach. We chose random images from dataset to find the least Euclidean distance image from trained dataset.

**Table 4.14:** Face recognition performance for Real Face database

| **Methods** | **TP** | **TN** | **FP** | **FN** | **Accuracy (%)** |
|---|---|---|---|---|---|
| Using PCA | 29 | 3 | 8 | 6 | 69.57 |
| Using Combined PCA-SVM | 36 | 9 | 1 | 0 | 97.83 |

Table 4.14 showed that face recognition accuracy using PCA algorithm on Real face dataset clearly is less than the Combined PCA-SVM method. Figure 4.14 depicts the graphical representation of the performance accuracy for Real dataset using PCA method and Combined Principal Component Analysis – Support Vector Machine.

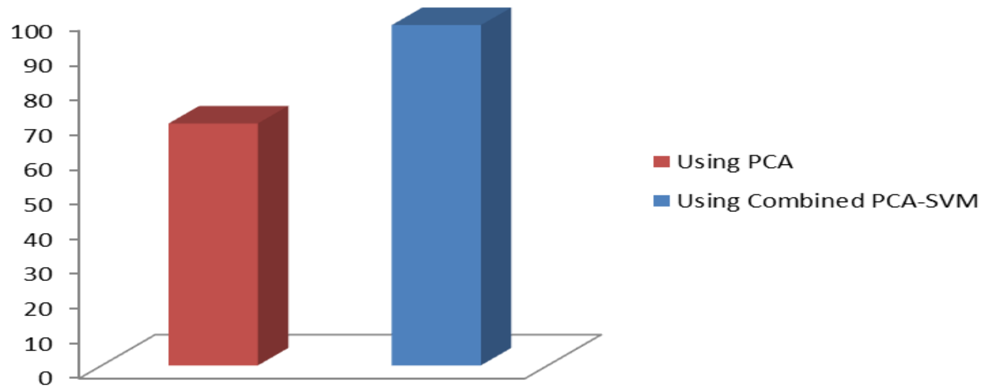



**Figure 4.14:** Graphical Representation of Face recognition performance for Real Face database

Face recognition performance using the combined PCA and SVM approach is better than the PCA method. We experiment the combined PCA-SVM approach on Real face dataset which contains real images along with PCA algorithm. SVM is used to detect faces whether with mask or not. There some parameters such as contrast, energy, homogeneity, entropy, variance and smoothness were used to detect whether masked face or not. Accuracy of this method is very satisfactory over PCA algorithm for covered face and eyes situations. It also provides better performance for the images with sunglasses and scarf. From the result it is clearly shown that this research methodology works well. So, the Combined PCA-SVM method is efficient for removing the complexities of covered face recognition. SVM method works to detect mask and sunglasses along with PCA is used to recognize faces accurately. When it is predictable that face is with mask or not then face recognition process will be more efficient for covered face recognition.

## 4.6 COMPASION AMONG MODELS

In this work, we have used three different models called combined PCA-Segmentation, PCA-Daugman and PCA-SVM. These models provide a good performance to recognize a person even when the person wears sun glasses, mask, hijab and scarf. Models are also efficient to detect a person who have long beard and mustache. Figure 4.15 shows the performance comparison of three models. Although all of these models are very efficient for covered face recognition, the best performance is found from combined PCA-Segmentation model and other two models have provided very similar performance but less than the combined PCA-Segmentation method. It provides better performance because it uses three different trained datasets concurrently and calculate the minimum Euclidean distance to detect the proper person. It also works on specific target region of an image to increase the performance of the model.

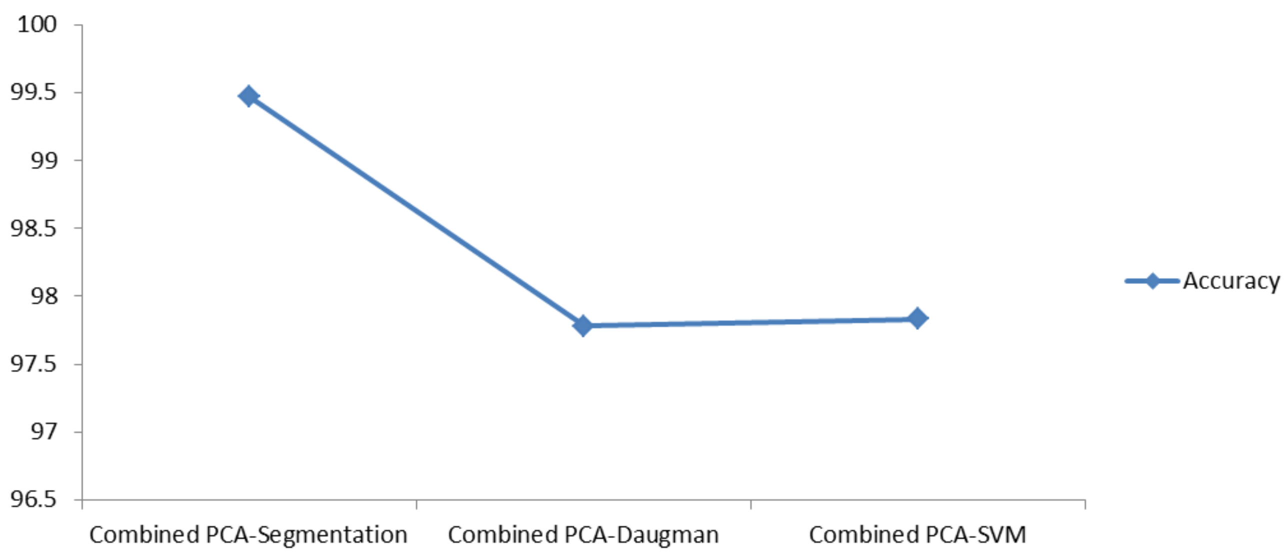



**Figure 4.15:** Performance comparison of models

## 4.7 SUMMARY

This chapter presented the results and findings of the study with regard to the efficient person recognition with covered face and covered eyes using Combined PCA-Segmentation, PCA-Daugman and PCA-SVM comparing to the traditional methods. The proposed methods executed over different dataset to compare the performances. This chapter have proved the efficiency of the proposed methods for person recognition with covered face and eyes.

# CHAPTER 5

# CONCLUSION AND FUTURE WORK

The motto of our work was to develop and compare various combined system which will provide better recognition result than PCA method for covered face and eyes recognition. Principal Component Analysis (PCA) algorithm for face recognition process uses eigen faces and eigen vectors. For covered face recognition PCA algorithm is not efficient because PCA provides very less accuracy for covered faces. We have used three different models called combined PCA-Segmentation, PCA-Daugman and PCA-SVM.

This work combined principal component analysis (PCA) with the Segmentation method, Daugman method and Support Vector Machine for the training of image datasets. This work is done using the MATLAB software. Segmentation method is used for cropping the images in specific region. Daugman method is used to find the iris and pupil contour. SVM is used to classify images into different classes.

In this work, there were four different image datasets to use called ORL, Yale, Real and CASIA dataset. Among these ORL, Yale and Real are face image dataset although CASIA is iris image dataset. ORL face database has 400 images of 40 persons and there are 10 different images per person. The Yale Face Database has total 165 images of different 15 person and the images have various expressions. Real face data set has 300 images of 60 different persons with 5 different images each. Face images consists of face with mask and sunglasses. CASIA Database has 108 eyes with 756 iris images.

Accuracy calculation needed some parameters such as- the total number of epochs for test image execution along with true positive, true negative, false positive and false negative cases for face recognition. Combined PCA-Segmentation, PCA-Daugman and PCA-SVM models provide better performance for covered face recognition than the solo PCA method. Introduction of this report have discussed background, problem statement, objectives and scope of the study. The main objective of this study is to recognize person efficiently with covered face and eyes. Literature Review have discussed previous works which are related to this work and motivation of this study. Methodology consists of models, datasets and accuracy of the study. Models and related algorithms are mentioned at methodology section. Result and Discussion section have discussed the results of the models for different datasets and it showed the performance of the models.

The models provide a solution to recognize a person even when the person wears sun glasses, mask, hijab and scarf. Models are also able to detect a person who have long beard and mustache. In future, we wish to use more extended and realistic complex database and apply more efficient and dynamic algorithms to increase the accuracy and to reduce the complexities. We also wish to use advance artificial intelligence and machine learning approaches to make the recognition system more adaptable and reliable for the near future.

# REFERENCES

[1] “What is Facial Recognition? - Definition from Techopedia,” Techopedia.com. [Online]. Available: https://www.techopedia.com/definition/32071/facial-recognition. [Accessed: 03-Mar-2020].

[2] Ion Marques, Manuel Grana, "Face Recognition Algorithms", June 16, 2010.

[3] Xingfu Zhang and Xiangmin Ren, "Two Dimensional Principal Component Analysis based Independent Component Analysis for face recognition," 2011 International Conference on Multimedia Technology, Hangzhou, 2011, pp. 934-936.

[4] J. Shermina, "Illumination invariant face recognition using Discrete Cosine Transform and Principal Component Analysis," 2011 International Conference on Emerging Trends in Electrical and Computer Technology, Nagercoil, 2011, pp. 826-830.

[5] Xiangfei Nie and Zefu Tan, "Face recognition using Bidirectional Principal Component Analysis and wavelet transform," 2011 International Conference on Electric Information and Control Engineering, Wuhan, 2011, pp. 4203-4206.

[6] W. Y. Min, E. Romanova, Y. Lisovec and A. M. San, "Application of Statistical Data Processing for Solving the Problem of Face Recognition by Using Principal Components Analysis Method," 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Saint Petersburg and Moscow, Russia, 2019, pp. 2208-2212.

[7] S. Lim, K. Lee, O. Byeon, T. Kim. “Efficient iris recognition through improvement of feature vector and classifier”,2001

[8] R. Wildes, J. Asmuth, G. Green, S. Hsu, R. Kolczynski, J. Matey, S. McBride,“A system for automated iris recognition”,1994

[9] Song, Y., & Yan, H. (2017). Image Segmentation Techniques Overview. 2017 Asia Modelling Symposium (AMS).

[10] Taneja, A., Ranjan, P., & Ujjlayan, A. (2015). A performance study of image segmentation techniques. 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions).

[11] Scheleyer, G., Cubillos, C., Lefranc, G., Osorio-Comparan, R., & Millan, G. (2016). A new colour image segmentation. 2016 6th International Conference on Computers Communications and Control (ICCCC).

[12] M. Sharkas, "A neural network based approach for iris recognition based on both eyes," 2016 SAI Computing Conference (SAI), London, 2016, pp. 253-258.

[13] Chandranayaka I R, “Various Iris Recognition Algorithms for Biometric Identification: A Review”, International Journal of Exploring Emerging Trends in Engineering (IJEETE) Vol. 03, Issue 04, JUL-AUG, 2016 Pg. 286 – 290

[14] R. Senthilkumar and R. K. Gnanamurthy, "Performance improvement in classification rate of appearance based statistical face recognition methods using SVM classifier," 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, 2017, pp. 1-7.

[15] Singh, Patil, Reddy, & Omkar (2017). Disguised Face Identification (DFI) with Facial KeyPoints Using Spatial Fusion Convolutional Network. 2017 IEEE International Conference on Computer Vision Workshops (ICCVW).

[16] Dr. Priya Gupta, Nidhi Saxena, Meetika Sharma, Jagriti Tripathi. Deep Neural Network for Human Face Recognition. I.J. Engineering and Manufacturing, 2018, 1, 63-71

[17] Devendra Gondole, Prof. P. A. Salunkhe (2018). Face Recognition Based On Local Binary Pattern.

[18] Ayi, M., Ganti, A. K., Adimulam, M., & Karthik, B. (2017). Interfacing of MATLAB with Arduino for face detection and tracking algorithm using serial communication. 2017 International Conference on Inventive Computing and Informatics (ICICI).

[19] Parvaneh and Hadi (2015). Face recognition using Eigen faces, PCA and support vector machines. European Journal of Applied Engineering and Scientific Research.

[20] Faruqe and Hasan (2009). Face recognition using PCA and SVM. 3rd International Conference on Anti-Counterfeiting, Security, and Identification in Communication.

[21] B. Ammour, T. Bouden and S. Amira-Biad, "Multimodal biometric identification system based on the face and iris," 2017 5th International Conference on Electrical Engineering - Boumerdes (ICEE-B), Boumerdes, 2017, pp. 1-6.

[22] L. Mansoura, A. Noureddine, O. Assas and A. Yassine, "Biometric recognition by multimodal face and iris using FFT and SVD methods With Adaptive Score Normalization,"

2019 4th World Conference on Complex Systems (WCCS), Ouarzazate, Morocco, 2019, pp. 1-5.

[23] Md. Rabiul Islam, Md. Abdus Sobhan, Rizoan Toufiq. Face and Iris based Multimodal Person Identification using Likelihood Ratio Score Fusion Approach. Journal of Computer Technology & Applications. 2015; 6(3): 60–67p

[24] Singh, Sukhdev and Kant, Chander, FKP and Iris Based Multimodal Biometric System Using PCA with NFNN (February 24, 2019). Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur - India, February 26-28, 2019.

[25] Jie Lin, Jian-Ping Li, Hui Lin and Ji Ming, "Robust person identification with face and iris by modified PUM method," 2009 International Conference on Apperceiving Computing and Intelligence Analysis, Chengdu, 2009, pp. 321-324.

[26] S. D. Jamdar and Y. Golhar, ""Implementation of unimodal to multimodal biometrie feature level fusion of combining face iris and ear in multi-modal biometric system"," 2017 International Conference on Trends in Electronics and Informatics (ICEI), Tirunelveli, 2017, pp. 625-629.

[27] Y. Bouzouina and L. Hamami, "Multimodal biometric: Iris and face recognition based on feature selection of iris with GA and scores level fusion with SVM," 2017 2nd International Conference on Bio-engineering for Smart Technologies (BioSMART), Paris, 2017, pp. 1-7.

[28] T. Ko, "Multimodal biometric identification for large user population using fingerprint, face and iris recognition," 34th Applied Imagery and Pattern Recognition Workshop (AIPR'05), Washington, DC, 2005, pp. 6 pp.-223.

[29] V. Azom, A. Adewumi and J. Tapamo, "Face and Iris biometrics person identification using hybrid fusion at feature and score-level," 2015 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), Port Elizabeth, 2015, pp. 207-212.

[30] V. V. Dakre and P. G. Gawande, "An efficient technique of multimodal biometrics using fusion of face and iris features," 2016 Conference on Advances in Signal Processing (CASP), Pune, 2016, pp. 231-236.

[31] J. Zhang, X. Zhang and S. Ha, "A Novel Approach Using PCA and SVM for Face Detection," 2008 Fourth International Conference on Natural Computation, Jinan, 2008, pp. 29-33.

[32] X. Li and G. Chen, "Face Recognition Based on PCA and SVM," 2012 Symposium on Photonics and Optoelectronics, Shanghai, 2012, pp. 1-4.
[33] Gumus, E., Kilic, N., Sertbas, A., & Ucan, O. N. (2010). Evaluation of face recognition techniques using PCA, wavelets and SVM. Expert Systems with Applications, 37(9), 6404–6408.
[34] Luo, Y., Wu, C., & Zhang, Y. (2013). Facial expression recognition based on fusion feature of PCA and LBP with SVM. Optik - International Journal for Light and Electron Optics, 124(17), 2767–2770.
[35] Li, M., Yu, X., Ryu, K.H. et al. Face recognition technology development with Gabor, PCA and SVM methodology under illumination normalization condition. Cluster Comput **21,** 1117–1126 (2018).
[36] Ragab, M. E., Darwish, A. M., Abed, E. M., & Shaheen, S. I. (1999). Face Recognition Using Principal Component Analysis Applied to an Egyptian Face Database. Lecture Notes in Computer Science, 540–549. doi:10.1007/978-3-540-48765-4_58
[37] L. Rui, W. Jiamei, L. Bingze and Y. Changsen, "Digital information mining technology based on principal component analysis," 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE), Beijing, 2017, pp. 619-623.
[38] F. Song, Z. Guo and D. Mei, "Feature Selection Using Principal Component Analysis," 2010 International Conference on System Science, Engineering Design and Manufacturing Informatization, Yichang, 2010, pp. 27-30.
[39] A. Alkandari and S. J. Aljaber, "Principle Component Analysis algorithm (PCA) for image recognition," 2015 Second International Conference on Computing Technology and Information Management (ICCTIM), Johor, 2015, pp. 76-80.
[40] Milad Soltan, Saeid Toosi Zadeh and Hamid-Reza Pourreza, “Daugman’s Algorithm Enhancement for Iris Localization” Advanced Materials Research Vols. 403-408 (2012) pp 3959-3964.
[41] Prateek Verma, Maheedhar Dubey, Praveen Verma and Somak Basu, “Daughman’s Algorithmmethod for Iris Recognition-A Biometric Approach” International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 6, June 2012.
[42] Dilpreet Kaur and Yadwinder Kaur, “Various Image Segmentation Techniques: A Review”. International Journal of Computer Science and Mobile Computing, Vol.3 Issue.5, May- 2014, pg. 809-814, Johor, 2015, pp. 76-80.

[43] “Support Vector Machines: A Simple Explanation,” KDnuggets. [Online]. Available: https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html. [Accessed: 01-Sep-2020].

[44] P. Dinkova, P. Georgieva, A. Manolova and M. Milanova, "Face recognition based on subject dependent Hidden Markov Models," 2016 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Varna, 2016, pp. 1-5.

[45] M. Agarwal, H. Agrawal, N. Jain and M. Kumar, "Face Recognition Using Principle Component Analysis, Eigenface and Neural Network," 2010 International Conference on Signal Acquisition and Processing, Bangalore, 2010, pp. 310-314.

[46] K. Shailaja and B. Anuradha, "Effective face recognition using deep learning based linear discriminant classification," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Chennai, 2016, pp. 1-6.

[47] M. A. Abuzneid and A. Mahmood, "Enhanced Human Face Recognition Using LBPH Descriptor, Multi-KNN, and Back-Propagation Neural Network," in IEEE Access, vol. 6, pp. 20641-20651, 2018.

[48] “UCSD Computer Vision,” Yale Face Database. [Online]. Available: http://vision.ucsd.edu/content/yale-face-database. [Accessed: 07-Mar-2020].

[49] Biometrics Ideal Test. [Online]. Available: http://biometrics.idealtest.org/dbDetailForUser.do?id=1. [Accessed: 21-Apr-2020].

# APPENDICES

## A-1

```
clear all
close all
clc

%add the training path name where images are stored
training_path='D:\naeem\all is well\code\okk\try\image4\';

training_set_length=120;
images=[];
mean_image=[];
disp('Start data train process.....')

for a = 1:training_set_length
   curr_path=strcat(training_path,int2str(a),'.jpg');
   image=imread(curr_path);
   %imshow(image)
   %{
    if (a==7 | a==8)
    image=rgb2gray(image); %already done
   end

   I = imread('circuit.tif');
  Crop image, specifying crop rectangle.
```

```
I2 = imcrop(I,[75 68 130 112]);
   %}
   %if length(unique(image)) > 255
   if size(image,3)==3
     image = rgb2gray(image);
    end
  image = imresize(image, [128 128]);
  image=reshape(image,[],1);
   images = [images,image];
   if a==1
       mean_image=image/training_set_length;
   else
       try_img=image/training_set_length;
       mean_image=mean_image+try_img;
   end
end
%mean_image=mean_image/training_set_length;

for a=1:training_set_length

    images(:,a)=images(:,a)-mean_image;
end
images_transpose=transpose(images);
covariance=double(images_transpose)*double(images);

[v,d]=eig(covariance);
eigenfaces=[];
for a=1:training_set_length

    eigenfaces=[eigenfaces,double(images)*double(v(:,a))];
end
%Calculating All Training Vectors
sol=[];
for a=1:training_set_length
```

```
  sol(:,a)=linsolve(double(eigenfaces),double(images(:,a)));
end
disp('Training process completed')
%Testing Phase

disp('Select an image for testing')
%[filename,pathname]=uigetfile('*.jpg','Select the test image');
Testing_Path = 'D:\bbbbbbbbbbb\all is well\code\okk\try\image3';    %Set your directory for testing data
file



[filename, pathname] = uigetfile({'*.jpg'},'Pick a Testing Photo From TestDatabase please');
disp('Hold a second for computing')

disp('Visualize eigenfaces')
Visualize_Eigenface(eigenfaces,128,128);
disp('Recognizing..........')
TestImage = [pathname, filename];
im = imread(TestImage);
%{
if length(unique(im)) > 255
   im = rgb2gray(im);
end
%}
im = imresize(im, [128 128]);

%test= imread(fullfile(pathname,filename), 'jpg');
chosen_image =im;
%imshow(chosen_image);
%chosen_image = imread(TestImage);
%chosen_image=rgb2gray(chosen_image);
chosen_image=reshape(chosen_image,[],1);
chosen_image=chosen_image-mean_image;
chosen_image=double(chosen_image);
```

```
chosen_image_sol=linsolve(double(eigenfaces),chosen_image);   %Testing vector
eucledian_distance=[];
for a=1:training_set_length
    eucledian_distance(:,a)= sqrt(sum((chosen_image_sol - sol(:,a)) .^ 2));
end
[M,I] = min(eucledian_distance);

clc;
%Print the results
fprintf('Minimum Eucledian distance %f\n ',M);
fprintf('Chosen image matches to image %d ',I);

%To be worked
OutputName = strcat(int2str(I),'.jpg');
%imshow(img_g, 'InitialMagnification', 'fit')

 SelectedImage = strcat(training_path,'\',OutputName);
 SelectedImage = imread(SelectedImage);

figure('name','Recognition Result')
subplot(1,2,1);
imshow(im)
title('Test Image');

subplot(1,2,2);
imshow(SelectedImage);
title('Recognition Result');
```

**A-2**

```
clear all
close all
clc


training_path= 'C:\Users\Mahbubul Alam Naeem\Desktop\all is well\code\okk\try\ORL TestDatabase
[Combined]\';
dname_save= 'C:\Users\Mahbubul Alam Naeem\Desktop\all is well\code\okk\try\image3\';

for a = 1:138
   %curr_path=strcat(training_path,int2str(a),'.jpg');
   curr_path=strcat(training_path,int2str(a),'.jpg');
   image=imread(curr_path);

   %image=imread(dname_open);
   if size(image,3)==3
     image = rgb2gray(image);
   end
    %imshow(image)

   % image = imcrop(image,[5 30 128 50]);
   %image = imcrop(image,[15 75 95 50]);
   %image = imresize(image, [128 128]);
  %image=reshape(image,[],1);

  imshow(image)

%imcrop;

%set(gcf, 'PaperPositionMode', 'auto');
%h=gcf;
%saveas(h,[dname_save '\' a], 'jpg')
```

```
%imwrite(image);

%id = 5;
%folder = 'c:\result\'
newimagename = [dname_save num2str(a) '.jpg'];
imwrite(image,newimagename)
end
```

**A-3**

```
clear all
close all
clc

training_path= 'C:\Users\Mahbubul Alam Naeem\Desktop\all is well\code\okk\try\image\';
dname_save= 'C:\Users\Mahbubul Alam Naeem\Desktop\all is well\code\okk\try\image1\';

for a = 1:120
  %curr_path=strcat(training_path,int2str(a),'.jpg');
  curr_path=strcat(training_path,int2str(a),'.jpg');
  image=imread(curr_path);

  %image=imread(dname_open);
  if size(image,3)==3
    image = rgb2gray(image);
  end
   %imshow(image)

  % image = imcrop(image,[5 30 128 50]);
  image = imcrop(image,[5 30 128 60]);
  image = imresize(image, [128 128]);
 %image=reshape(image,[],1);
```

```
  imshow(image)

%imcrop;

%set(gcf, 'PaperPositionMode', 'auto');
%h=gcf;
%saveas(h,[dname_save '\' a], 'jpg')

%imwrite(image);

%id = 5;
%folder = 'c:\result\'
newimagename = [dname_save num2str(a) '.jpg'];
imwrite(image,newimagename)
end
```

**A-4**

```
function Visualize_Eigenface(Eigenfaces, imgrow, imgcol)
%-------------------Show the maximum nine pictures of Eigenfaces---------------
   Num_Eigenvalue = size(Eigenfaces,2);
   figure('Name','Eigenface')
   img = zeros(imgrow, imgcol);
   for i=1:min(Num_Eigenvalue,9)
      img(:) = Eigenfaces(:,i);
      subplot(3,3,i);
      imshow(img',[]);
   end
end
```

**A-5**

```
training_path='C:\Users\Mahbubul Alam Naeem\Desktop\all is well\code\okk\try\image2\';

  %Set your directory for training data file

training_set_length=120;
images=[];
mean_image=[];
disp('Start data train process.....')



for a = 1:training_set_length
   curr_path=strcat(training_path,int2str(a),'.jpg');
   image=imread(curr_path);
   %imshow(image)
   %{
    if (a==7 | a==8)
    image=rgb2gray(image); %already done
   end

   I = imread('circuit.tif');
Crop image, specifying crop rectangle.

I2 = imcrop(I,[75 68 130 112]);
   %}
   %if length(unique(image)) > 255
   if size(image,3)==3
     image = rgb2gray(image);
    end
  image = imresize(image, [128 128]);
  image=reshape(image,[],1);
```

```
  images = [images,image];
  if a==1
     mean_image=image/training_set_length;
  else
     try_img=image/training_set_length;
     mean_image=mean_image+try_img;
  end
end
%mean_image=mean_image/training_set_length;

for a=1:training_set_length

   images(:,a)=images(:,a)-mean_image;
end
images_transpose=transpose(images);
covariance=double(images_transpose)*double(images);

[v,d]=eig(covariance);
eigenfaces=[];
for a=1:training_set_length

   eigenfaces=[eigenfaces,double(images)*double(v(:,a))];
end
%Calculating All Training Vectors
sol=[];
for a=1:training_set_length
   sol(:,a)=linsolve(double(eigenfaces),double(images(:,a)));
end
disp('Training process completed')
%Testing Phase

disp('Select an image for testing')
%[filename,pathname]=uigetfile('*.jpg','Select the test image');
```

```
Testing_Path = 'C:\Users\Mahbubul Alam Naeem\Desktop\all is well\code\okk\try\image3';    %Set
your directory for testing data file



[filename, pathname] = uigetfile({'*.jpg'},'Pick a Testing Photo From TestDatabase please');
disp('Hold a second for computing')

TestImage = [pathname, filename];
im = imread(TestImage);

%image = imcrop(image,[15 75 95 50]);
im = imcrop(im,[15 75 95 50]);
im = imresize(im, [128 128]);

disp('Visualize eigenfaces')
Visualize_Eigenface(eigenfaces,128,128);
disp('Recognizing..........')

%{
if length(unique(im)) > 255
    im = rgb2gray(im);
end
%}
%im = imresize(im, [128 128]);

%test= imread(fullfile(pathname,filename), 'jpg');
chosen_image =im;
%imshow(chosen_image);
%chosen_image = imread(TestImage);
%chosen_image=rgb2gray(chosen_image);
chosen_image=reshape(chosen_image,[],1);
chosen_image=chosen_image-mean_image;
chosen_image=double(chosen_image);
chosen_image_sol=linsolve(double(eigenfaces),chosen_image);   %Testing vector
```

```
eucledian_distance=[];
for a=1:training_set_length
    eucledian_distance(:,a)= sqrt(sum((chosen_image_sol - sol(:,a)) .^ 2));
end
[M,I] = min(eucledian_distance);

clc;
%Print the results
fprintf('Minimum Eucledian distance %f\n ',M);
fprintf('Chosen image matches to image %d ',I);

OutputName = strcat(int2str(I),'.jpg');
%imshow(img_g, 'InitialMagnification', 'fit')

 SelectedImage = strcat(training_path,'\',OutputName);
 SelectedImage = imread(SelectedImage);

figure('name','Recognition Result')
subplot(1,2,1);
imshow(im)
title('Test Image');

subplot(1,2,2);
imshow(SelectedImage);
title('Recognition Result');
```

**A-6**

```
function hd = gethammingdistance(template1, mask1, template2, mask2, scales)

template1 = logical(template1);
mask1 = logical(mask1);
```

```
template2 = logical(template2);
mask2 = logical(mask2);

hd = NaN;

% shift template left and right, use the lowest Hamming distance
for shifts=-8:8

    template1s = shiftbits(template1, shifts,scales);
    mask1s = shiftbits(mask1, shifts,scales);



    mask = mask1s | mask2;

    nummaskbits = sum(sum(mask == 1));

    totalbits = (size(template1s,1)*size(template1s,2)) - nummaskbits;

    C = xor(template1s,template2);

    C = C & ~mask;
    bitsdiff = sum(sum(C==1));

    if totalbits == 0

        hd = NaN;

    else

        hd1 = bitsdiff / totalbits;


        if  hd1 < hd || isnan(hd)
```

```
        hd = hd1;

      end


    end

end
```

**A-7**

```
function [polar_array, polar_noise] = normaliseiris(image, x_iris, y_iris, r_iris,...
x_pupil, y_pupil, r_pupil,eyeimage_filename, radpixels, angulardiv)

global DIAGPATH

radiuspixels = radpixels + 2;
angledivisions = angulardiv-1;

r = 0:(radiuspixels-1);

theta = 0:2*pi/angledivisions:2*pi;

x_iris = double(x_iris);
y_iris = double(y_iris);
r_iris = double(r_iris);

x_pupil = double(x_pupil);
y_pupil = double(y_pupil);
r_pupil = double(r_pupil);
```

```
% calculate displacement of pupil center from the iris center
ox = x_pupil - x_iris;
oy = y_pupil - y_iris;

if ox <= 0
    sgn = -1;
elseif ox > 0
    sgn = 1;
end

if ox==0 && oy > 0

    sgn = 1;

end

r = double(r);
theta = double(theta);

a = ones(1,angledivisions+1)* (ox^2 + oy^2);

% need to do something for ox = 0
if ox == 0
    phi = pi/2;
else
    phi = atan(oy/ox);
end

b = sgn.*cos(pi - phi - theta);

% calculate radius around the iris as a function of the angle
r = (sqrt(a).*b) + ( sqrt( a.*(b.^2) - (a - (r_iris^2))));

r = r - r_pupil;
```

```
rmat = ones(1,radiuspixels)'*r;

rmat = rmat.* (ones(angledivisions+1,1)*[0:1/(radiuspixels-1):1])';
rmat = rmat + r_pupil;



% exclude values at the boundary of the pupil iris border, and the iris scelra border
% as these may not correspond to areas in the iris region and will introduce noise.
%
% ie don't take the outside rings as iris data.
rmat  = rmat(2:(radiuspixels-1), :);

% calculate cartesian location of each data point around the circular iris
% region
xcosmat = ones(radiuspixels-2,1)*cos(theta);
xsinmat = ones(radiuspixels-2,1)*sin(theta);

xo = rmat.*xcosmat;
yo = rmat.*xsinmat;

xo = x_pupil+xo;
yo = y_pupil-yo;

% extract intensity values into the normalised polar representation through
% interpolation
[x,y] = meshgrid(1:size(image,2),1:size(image,1));
polar_array = interp2(x,y,image,xo,yo);

% create noise array with location of NaNs in polar_array
polar_noise = zeros(size(polar_array));
coords = find(isnan(polar_array));
polar_noise(coords) = 1;
```

```
polar_array = double(polar_array)./255;

% start diagnostics, writing out eye image with rings overlayed

% get rid of outling points in order to write out the circular pattern
coords = find(xo > size(image,2));
xo(coords) = size(image,2);
coords = find(xo < 1);
xo(coords) = 1;

coords = find(yo > size(image,1));
yo(coords) = size(image,1);
coords = find(yo<1);
yo(coords) = 1;

xo = round(xo);
yo = round(yo);

xo = int32(xo);
yo = int32(yo);

ind1 = sub2ind(size(image),double(yo),double(xo));

image = uint8(image);

image(ind1) = 255;
%get pixel coords for circle around iris
[x,y] = circlecoords([x_iris,y_iris],r_iris,size(image));
ind2 = sub2ind(size(image),double(y),double(x));
%get pixel coords for circle around pupil
[xp,yp] = circlecoords([x_pupil,y_pupil],r_pupil,size(image));
ind1 = sub2ind(size(image),double(yp),double(xp));

image(ind2) = 255;
```

```
image(ind1) = 255;


% write out rings overlaying original iris image
w = cd;
cd(DIAGPATH);

imwrite(image,[eyeimage_filename,'-normal.jpg'],'jpg');

cd(w);

% end diagnostics

%replace NaNs before performing feature encoding
coords = find(isnan(polar_array));
polar_array2 = polar_array;
polar_array2(coords) = 0.5;
avg = sum(sum(polar_array2)) / (size(polar_array,1)*size(polar_array,2));
polar_array(coords) = avg;
```