KAUNAS UNIVERSITY OF TECHNOLOGY

INFORMATICS FACULTY

ASIF ALI SAHUKAR

INVESTIGATION OF SUBSPACE MIXTURE MODEL AND DISTANCE MEASURE FOR FACE RECOGNITION

Master's Degree Final Project

Supervisor
Assoc. prof. Armantas Ostreika

KAUNAS, 2016

KAUNAS UNIVERSITY OF TECHNOLOGY
INFORMATICS FACULTY

INVESTIGATION OF SUBSPACE MIXTURE MODEL AND DISTANCE MEASURE FOR FACE RECOGNITION

Master's Degree Final Project (621I10003)

M4016M21 Informatics

Supervisor
(signature) Assoc. prof. Armantas Ostreika
(date)

Reviewer
(signature) Assoc. prof. dr. Sigitas Drasutis
(date)

Project made by
(signature) Asif Ali Sahukar
(date)

KAUNAS, 2016

KAUNAS UN IVERSITY OF TECHNOLOGY
FACULTY OF INGORMATICS

Approved:

Head of Informatics Department

_______________ *(Signature, date)*

_______________ *(Name, Surname)*

Head of Study Programmes in the Field of Informatics

_______________ *(Signature, date)*

_______________ *(Name, Surname)*

MASTER STUDIES FINAL PROJECT TASK ASSIGNMENT
Study programme INFORMATICS – 621I0003

Approved by the Dean's Order No of month date, year y

Assigned to the student **ASIF ALI SAHUKAR**

Name, Surname

1. Title of the Project

INVESTIGATION OF SUBSPACE MIXTURE MODEL AND DISTANCE MEASURE FOR

2. Aim of the project

Investigation of subspace mixture model with different distance measure which is designed for efficient and robust face recognition task.

3. Tasks of the project

Summary, Introduction,
1.Overview of existing methodology on face recognition, 2. Methodology, 3. Experimental results and discussion, 4. Conclusions, References.

4. Specific Requirements

Conducting the final experimental project thesis according to KTU regulations and requirements

5. This task assignment is an integral part of the final project

6. Project submission deadline:

Task Assignment received _______________ (*Name, Surname of the Student*) _______________ (*Signature, date*)

Supervisor

_______________ (*Position, Name, Surname*) _______________ (*Signature, date*)

KAUNAS UNIVERSITY OF TECHNOLOGY

FACULTY OF INFORMATICS

(Faculty)

ASIF ALI SAHUKAR

(Student's name, surname)

M4016M21 Informatics (621I10003)

(Title of study programme, code)

"INVESTIGATION OF SUBSPACE MIXTURE MODEL AND DISTANCE MEASURE FOR FACE RECOGNITION"

DECLARATION OF ACADEMIC INTEGRITY

23 MAY 2016

Kaunas

I confirm that the final project of mine, **ASIF ALI SAHUKAR**, on the subject "**Investigation of subspace mixture model and distance measure for face recognition**" is written completely by myself; all the provided data and research results are correct and have been obtained honestly. None of the parts of this thesis have been plagiarized from any printed, Internet-based or otherwise recorded sources; all direct and indirect quotations from external resources are indicated in the list of references. No monetary funds (unless required by law) have been paid to anyone for any contribution to this thesis.

I fully and completely understand that any discovery of any facts of dishonesty inevitably results in me incurring a penalty under procedure effective at Kaunas University of Technology.

(name and surname filled in by hand) *(signature)*

Table of Contents

List of Figures

List of Tables

Asif Ali Sahukar. investigation of subspace mixture model and distance measure for face recognition. Master's Final Project / supervisor assoc. prof. Armantas Ostreika**.** Faculty of Informatics, Kaunas University of Technology.

Study area and field: Informatics.

Keywords*: Face recognition, Gaussian mixture model(GMM), Expectation-Maximization (EM) algorithm, Principal component analysis (PCA), Fisher linear discriminant analysis (LDA/FLDA), Feature Extraction, Distance measures, Feature Extraction, Maximum likelihood clusters, classification.*

SUMMARY

In this thesis work, we introduced subspace mixture model with different distance measure which is designed for efficient and robust face recognition task. The objective is to work on six different distance measures are used for classification purpose to obtain an average classification rate with standard databases. Face recognition is a challenging problem in image analysis that have been faced in computer vision fields and this field has tremendously grown because of the need for digital image recognition, authentication and also surveillance. The growing consumerism has provided the opportunity for more research developments related to this area. Hence the development of efficient face recognition algorithms is going on. This thesis presents a method for determining the significant features of a face within a maximum likelihood framework. We propose a FLD-Mixture Models and analyzed the effect of different distance metrics for Face Recognition System. In this method, first Expectation Maximization (EM) algorithm method is applied to learn mixture of Gaussian distributions to obtain best possible maximum likelihood clusters. Gaussian Mixture Models is used for clustering data in unsupervised context. Further, Fisher's Linear Discriminant Analysis (FLDA) is applied for K =4 mixtures to preserve useful discriminatory information in reduced feature space. All experiments are done on standard databases like ORL.

Asif Ali Sahukar. investigation of subspace mixture model and distance measure for face recognition. Master's Final Project / supervisor assoc. prof. Armantas Ostreika. Faculty of Informatics, Kaunas University of Technology.

Study area and field: Informatics.

Keywords*: Face recognition, Gaussian mixture model(GMM), Expectation-Maximization (EM) algorithm, Principal component analysis (PCA), Fisher linear discriminant analysis (LDA/FLDA), Feature Extraction, Distance measures, Feature Extraction, Maximum likelihood clusters, classification.*

SANTRAUKA

Darbe nagrinėjamas poerdvių maišos modelis (subspace mixture model) apjungiant jį su atstumų nustatymo algoritmu, kuris skirtas efektyviai ir patikimai atpažinti veido parametrus. Darbo tikslas yra apdoroti skirtingus veido parametrus - būdingus atstumus tarp veido taškų, išgauti vidutines klasifikavimo parametrų reikšmes, pasinaudojant prieinamų standartinių duomenų bazių duomenimis, bei pasiūlyti metodą, leidžiantį nustatyti reikšmingus veido bruožus maksimalaus tikėtinumo principu.

Darbe pasiūlyta naudoti FLD maišos modelius bei išanalizuotos pasirinktos atstumų įvertinimo metrikos bei jų įtaka veido atpažinimo skaičiavimams. Taikant pasiūlytą metodiką, pirmiausia pritaikomas Tikėtinumo maksimizavimo (Expectation Maximization - EM) algoritmas, siekiant "išmokinti" algoritmą ir gauti geriausius Gausinio pasiskirstymo parametrus. Gausinis maišos modelis yra panaudojamas automatiškai klasterizuoti (grupuoti) duomenis. Toliau, pritaikoma Fišerio linijinė diskriminantinė analizė (FLDA) siekiant išsaugoti naudingą informaciją sumažintoje atributinių reikšmių erdvėje. Visi eksperimentai atlikti panaudojant duomenis iš laisvai prieinamų standartinių duomenų bazių, tokių kaip ORL.

INTRODUCTION

Digital image processing has wide area of applications such as surveillance, biometric, medicine etc. In this biometric is an emerging field of bio-engineering, it is an automated method of identifying a person founded on physical characteristics. Many bio-metric systems are there such as signature, figure prints, face, voice, retina, hand geometry [1]. We have chosen face as bio-metric system and started this face recognition project. Here advantages are face is a natural and also it is a basic idea how humans easily differentiate from one person to other across their faces. Many performances and experiments on face recognition system has been done and it is improving parallel but still problem faced is in terms of producing accuracy.

It is extremely easy for humans to detect and recognize faces of different persons but this process has been more difficult for an automatic face recognition, since faces will be different with respect to their facial expressions, age, image quality, pose, and occlusion. [2] We can do frontal image face detection and recognition with various algorithms having high precision and efficiency. The problem comes when difference between the images of same face because of viewing direction and illuminations are higher than the image difference because of modification in face identity.

Automatic recognition [3] of different people have become an increasing attention from different community during last few years like machine learning, pattern recognition [4]. In this automatic face recognition is still a problem being solved. Face recognition includes man and machine interface and interaction and access control with virtual worlds contact less and also automatic search in image databases, gestures feature identification, experiments on face.

Face recognition system used to identify faces automatically from the different images and videos. It works in two ways, face verification that is authentication. This is one to many approaches, and it compares query face images with database, and the system needs to accept or reject the identity of input image. Face recognition that is identification, one to one approach, here input is an unknown face and the system gives back the preferred recognition from the database. In modern life security issues in pattern recognition, internet communication, computer vision and they need have authorized and control access.

The ultimate goal of automatic face recognition system not only to get the image structure but also to know what it represents. We can state the automatic face recognition as follows, given an image of a scene or a video, task is to identify or authentication of the persons in the scene or video by comparing with set of stored face databases. The solutions for this problem carried out by segmentation of face detection [5] from cluttered scenes, feature extraction from the faces, authentication or identification.

In this thesis work basically to get the hidden information we introduced latent variables that is with the Gaussian mixture model further this mixture models can be generalized by Expectation Maximization algorithm. To extract the information from these Gaussian mixtures we introduced Fisher linear discriminant to extract the hidden information. Finally, performance measured in terms of recognition accuracy with different distance measure techniques for different databases.

1.OVERVIEW OF EXISTING METHODOLOGY ON FACE RECOGNITION

Despite the fact that it is a great degree simple for individuals to identify and perceive faces in unconstrained environment, an automatic face recognition framework remains to a great degree troublesome on the grounds that the face pictures can shift impressively as far as outward appearance, crowded scene with Occlusions, age, picture quality, posture, and disguise [6]. Face detection and recognition can be done for the front face images with different algorithm with high accuracy. But the real problem occurs because of variation within the images of same face due to viewing angle, illumination is higher than image variations due to changes in face identities. Also, there are other parameters like image resolution, aperture, compression, lens deviations, exposure time and sensor response will make more variability within the different images of same subject that's is intra subject variability. Although many algorithms are developed to solve these problems, but they have to be explore more in order to get more efficiency. With respect to all above conditions computer program should learn that is train phase and later recognize test phase of new given faces. The main challenges [7] that are not able to solve from different algorithms are mentioned as below.

1. ***Large intra-subject variability*** - The presence of a same face image may fluctuate because of a few factors, for example, varieties appearance, lighting, stance, time furthermore because of occlusion. It neatly demonstrates the difficulties required in doing face recognition. Subsequently, the varieties between the pictures of the same face because of brightening that is illumination and viewing angle that is direction are bigger than the image varieties because of change in face identity.

2. ***Single sample per person*** - capability to work under single training sample per individual person requires huge consideration and attention to accomplish essential level of correctness. Numerous fruitful face recognition algorithms grew today highly relies on upon the train samples and even a few algorithms totally come up unsuccessful under this condition.

3.***Small sample size*** - The quantity of samples per individual person for training the hidden complex is generally much lesser than the dimensionality of the image space. Under such conditions, the system may not sum up well to identify unseen samples of the face.

4. ***Unconstrained conditions*** - Numerous algorithms work well under profoundly controlled situations. Their performance will drop under defied with unconstrained circumstances. Unconstrained conditions incorporate uncontrolled background, video sequence, varieties in face because of maturing, make-up, noise and many other.

5. ***Subject's non-cooperation*** - At long last, because of the subject's non collaboration, for example, the refusal to take a look at the camera and moving subject additionally enormously slow down the execution of the face identification scheme. Although there are many algorithms, efficient algorithm yet to be discover.

1.1 ALGORITHM CLASSIFICATION

Currently algorithms for face recognition are basically classified into two main categories, 1. Model based schemes in this 2dimensional models and 3dimensional models 2. Appearance based schemes.in this linear and non-linear approaches [8].

1.1.1 Model based schemes

The model-based face recognition scheme is focused for building a model of the human face, which can catch the facial varieties. Variations that is changes can happen essentially be in two parts: changes in shape and changes in texture pattern across the face. Both shape and texture can also vary because differences between individual and also due to changes in expression, lighting, viewpoint variations. The earlier information of human face is profoundly used to plan the model then there exists a strong concept known as model based approaches (statistical models of appearance), that will generate compact models of shape and texture variation to interpret images of faces. For instance, feature-based matching develops distance and corresponding location features from the placement of internal facial elements (e.g., eyes, etc.) [8].

The methodology depends on a vast and delegate set of training facial images each of which noted with set of highlighted feature points, then the position of these points are utilized to define characterize the state of the face and its variety over the training faces. The pattern of intensities is examined to learn different ways in which composition that is texture can differ. These two models together are equipped for synthesizing any of the training images and generalize them to interpret the new faces. The earlier learning of human face is used to outline the model. The model-based scheme usually contains three steps [8],1) Constructing the model; 2) Fit the model to the specified face image; 3) Consuming the constraints of the built-in model as the feature vector to compute the likeness among the query face and sample faces in the database to perform the recognition

One of the earliest face recognition algorithms based on automatic feature extraction was proposed by Kanade [9]. This system localizes the edges of eyes, nostrils, etc. in frontal perspective that is view images and compares the features against parameters of known faces. A recent feature-based system, based on elastic bunch graph matching, was developed by Wiskott et al. [10] as an extension to their original graph matching system [11]. Cootes et al built up a 2D morphable face model, through which the face variations are learned [12]. Numerous developed algorithms of this idea have been proposed. In our exploration, we focus on another alternative approaches known as subspace strategies or subspace methods or appearance based schemes. Is more straightforward and effective.

1.1.2 Appearance based schemes

Numerous ways to deal with object recognition and to computer graphics are directly based on images without use of midway three dimensional models. Many of these strategies depend on representation of images that include vector space structure and, in principle, requires dense correspondence [8]. This section gives a brief outline of subspace based methodologies that are efficiently created for face recognition. The goal of subspace based methodology is to extend the information of faces onto a dimensionally reduced space where the real recognition will be done. Appearance-based approaches represent an object as far as several object views. An image is reflected as a high-dimensional vector, i.e., a point in a high-dimensional vector space. Many view-based approaches use statistical methods to examine the spreading of the object image vectors in the vector space, and determine a proficient and effective representation (feature space) as per distinctive applications. Given a test image, the likeness among the stored examples and the test perspective is then completed in the feature space [8].

In appearance based scheme there are 2 approaches that is linear and non-linear, there are mainly three linear appearance based classifiers, principal component analysis, independent component analysis, fisher linear discriminant. Turk and Pentland in 1991, initially investigated the Principal Component Analysis (PCA) [13] for face recognition and utilized the PCA anticipated components as the features. The PCA is an unsupervised learning technique and thus does exclude label information of the data in order to work and utilizes label information with respect to classes of the data, Linear Discriminant Analysis (LDA) was proposed [14]. This method figures the basis vectors from the hidden information that ideally segregates among classes. This is not at all like the PCA technique, which scans for basis vectors that best depicts the data. The goal of LDA is to boost the between-class measure while minimizing the inside class measure. In any case, because of large dimensions, usage of the LDA technique turns into a difficult task, to determine this, the first n dimension of the information is anticipated onto l dimensional space utilizing PCA, where l _ n. This PCA+LDA representation is known as Fishers LDA (FLD) or Fisher face method [15].

The concept of Independent Component Analysis (ICA) also investigate for face recognition [16]. ICA will extract the information data contained in the higher-order connections among pixels too, this is a speculation of the PCA. ICA is different compare to PCA with fallowing aspects 1. Independent Component Analysis can also deal with data that's not a Gaussian 2. It minimizes higher order conditions, not at all like the PCA which minimizes the second order (moments) conditions of the information. 3. The vectors for through ICA need not be orthogonal in nature. The ICA was approaches with two fundamentally different architectures: Architecture-1 (ICA-I) and Architecture-2 (ICA-II). Where ICA-I reflect more local properties of the faces while the ICA-II reflect more global properties and thus closely captures the face information. As another option to the Principal Component Analysis, the Locality Preserving Projections (LPP) otherwise called Laplacian faces, was proposed which ideally protects the neighborhood structure of the information set [17]. The LPP shares more information representation properties of nonlinear procedures, for example, Laplacian Eigen maps or Locally Linear Embedding.

These algorithms are best in class subspace techniques proposed for face recognition. Numerous variations of these algorithms are contrived to overcome particular abnormality, for example, storage burden, computational complexity, Single Sample per Person (SSPP) issue and so forth. Particularly under latter circumstance, performance of numerous algorithms falls apart because of failure of training samples. Luckily, a few approaches to get around this trouble have been proposed [18]. Couple of effort were also made to make the Fisherface technique good under SSPP issue [19]. Many numerous forms of these algorithms have been proposed taking into account the standard Eigenface method last few years, for example, probabilistic Eigenface method, Support Vector Machines (SVM) strategy [20], and so on. Correlation techniques for enhancing the classification performance furthermore to reduce the computational complexity of the subspace based face recognition have also been presented [21].

A single approached model was developed to combine advantages of PCA, LDA and Bayesian subspace methods [22]. Using three-dimension subspace as axes a three-dimension space was made. A better accuracy was got through this subspace method. With the help of Gaussian mixture model [23] we extract a set of features instead of single feature to increase the performance of PCA and FLD methods. This method got good results under major difference in illumination and pose although they are computationally expensive. PCA and FLD methods are applied to whole image component wise, to utilize the local parts of whole image [24].This increased computational effort and also accuracy under different illumination and facial expressions.

The original PCA and FLD were put on discrete cosine transformation domain and we got result that DCT domain are same as obtained in spatial domain [25].Although many substitute methods were made to reduce the load on PCA and FLD. As, of DCT domain these algorithms also implemented on wavelet domain [26]. Here the advantage of transforming image from time domain to frequency domain to reduced computational efforts and to work on reduced set of coefficient.

The PCA and FLD methods are well known and most recognized but they are costly computationally. Because, of a two dimensional matrix has to convert into one dimensional matrix later this leads to large covariance matrix, with respect to this Eigen value solving task will be more time consuming one. To solve this a two dimensional PCA [2DPCA] [27] and two dimensional FLD [2DFLD] [9] which have a benefit that here no need to do vector covariance. Due to this benefit many merits were made 1there is a high accuracy improvement 2. Time for computation also reduced 3. Analyzing or computation of Fisherface in FLD became simpler. Major disadvantage of this method is it needs a big feature matrix to identification and representation. Then the drawbacks of these methods come across by two dimensional two PCA [2D2PCA] [28] and two dimensional two FLD [2D2FLD] [29]. Here recognition accuracy of these methods were high or same than the corresponding 2D versions, these reduced in storage requirement and time for computation when compared with the 2DPCA and 2DFLD methods.

More new methods namely DiaPCA [30] and DiaFLD [29] have been proposed. These methods check optimal projection vectors from diagonal face image so that the correlation between variations across rows and columns of images well-preserved [30]. DiaPCA and DiaFLD methods combines with 2DPCA and 2DFLD to get more accuracy and storage efficiency. Although compare to 2DPCA and 2DFLD efficiency of DiaPCA and DiaFLD was more.

Recently, a noteworthy improvement to subspace technique has been proposed by taking a tensor way to deal with image information [7]. Two vital aspects of this has the capacity to get over nearby minimums and been fast meeting that is convergence. These outcomes have taken the standard administered learning methodologies to Support Vector Machines, minimax probability machine, FLD and stretched out into multi-linear structure of this with enhanced results. As, far as ongoing application.

From above literature survey it’s well known that subspace method in new one from past many years and till now we are getting new and better results day by day in this. From past many years there was sharp increase in implementations and investigations for real time streaming problems. All these problems and privies works make research communities to more work and focus on present work on that.

1.2 CLASSIFICATION TECHNIQUES

Classification between the object is easy assignment for people however it has turned out to be complex issue for machines. The situation where highly powerful machines with lower resolution cameras with respect to this parallel increasing need of automatic face recognition or video analysis yields to classification algorithms. Classification consist of database that compares with recognized image. That is classification done in order to find the similarities between query and store image. Distance measure or similarity measures are easier to solve numerous pattern recognition problems like classification, clustering, and recovery or retrieval. Many retrieval systems fail due to inefficient classification techniques, with this we proposed here viability of various distance or similarity measure techniques. There are mainly two basic types of image classification, supervised and unsupervised.

1. Supervised classification: The procedure of utilizing tests of known training set (preparing sets) to classify pixels of unknown character [31]. That is image expert manages the selection of regions/clusters that signify patterns/features that the analyst can identify. Supervised classification relies on nature of training set. All supervised classification work as fallowing, learning step and classification. Characterizing the training set that is usually two or three training set will be selected, more the set of training better the result will be.

Extraction of features, finally classification of images. Once the selection of training set and extracting the feature finally classification will be done Development of a classification scheme by selecting representative areas using reference sources such as higher resolution imagery or field data. The software then describes the statistical designs of the illustrative spaces and categorizes the image. Example, distance to means algorithm, parallelepiped calculation, maximum likelihood algorithm.

2. Unsupervised classification: In unsupervised classification, unknown pixels are divided into number of classes [31]. That is statistical "clustering" algorithms used to select classes inherent to the data. From the situations like where less information to classifies in an area, only image should be characteristics are used as fallows, group of data divided in to set of classes by using clustering techniques. Then this cluster classes will be used for further solving the problem. For example, the software is used to explore the image and calculate clusters that signify collections of pixels with alike spectral characteristics. Those clusters are evaluated to determine what number of clusters is needed, then the pixels are assigned to the best-fit cluster.

Many distance measure techniques are there some of them are like Euclidean, modified squared Euclidean, Manhattan, Weighted Manhattan, Minkowski, Angle-based, Correlation Co-efficient, Mahalonobis, Mahalonobis between normed vector, Mean Square Error, Canberra, Weighted mean squared error, based distance measure techniques [32].

1.3 DATABASES USED IN THE WORK

In our project we conducted all experiments based on standard databases. Some existing standard databases, like ORL database images are taken between 1991-1994 developed by AT&T Laboratories Cambridge [33]. 40 images in PGM format in gray scale with the properties varying lighting condition, facial expression. The images are in 40 directories, with 10 different view per directory, we represent it by sX, and Y.pgm, here x is image among 40 directories and. pgm to represent 10 among one view of a particular image.

1.4 APPLICATIONS AND ADVANTAGES FOR FACE RECOGNITION

Face recognition is used for mainly two primary tasks [37].

1. Verification (one-to-one matching)*:* When given a face image of an unidentified individual along with a case of identity, finding out whether the individual is who he/she.

2. ***Identification (one-to-many matching):*** There will be an image of an unidentified person, establishing that person's identity by matching that image with a database images of well-known persons.

Many applications are there for face recognition, some of them are listed below [37].

1. Security access control in the places like constructions, airports, ATM's and border checkpoints, network security, email authentication on multimedia workstations.
2. Surveillance that is a huge number of CCTV's can be observed for finding out known criminals, drug offenders, etc. And authorities can be notified when one is located.
3. Verification general identity that is electoral registration, banking, electronic commerce, national IDs, passports, driver's licenses, employee IDs.
4. Criminal justice systems, forensics.

5. Examinations of image database that is searching image databases of licensed drivers, advantage recipients, missing children, immigrants and police bookings.

6. Application like smart card here there will be a database of facial images, the face-print can be saved in a smart card, bar code or magnetic stripe, here identification will be provided by comparing store image and live image.

7. Finding out the faces in a video and labeling it.

8. Witness face reconstruction.

With the fallowing above stated applications, the primary techniques of face recognition technology have also been altered and used for related applications like gender classification, expression recognition and facial feature recognition and tracking, fallowing this has been used in various fields. For example, expression recognition used in the field of medicine for intensive care while facial feature identification and recognition can be exploited for tracking a vehicle driver's eyes and thus monitoring his weakness, as well as for stress detection [37].

Face recognition is also being used in combining with other bio-metrics such as speech, iris, fingerprint, ear and posture recognition in order to enhance the recognition performance of these methods [37].

2. METHODOLOGY

An overview of face recognition can be depicted in this given below figure (Figure1), here we can make it into main two phases test phase and train phase.

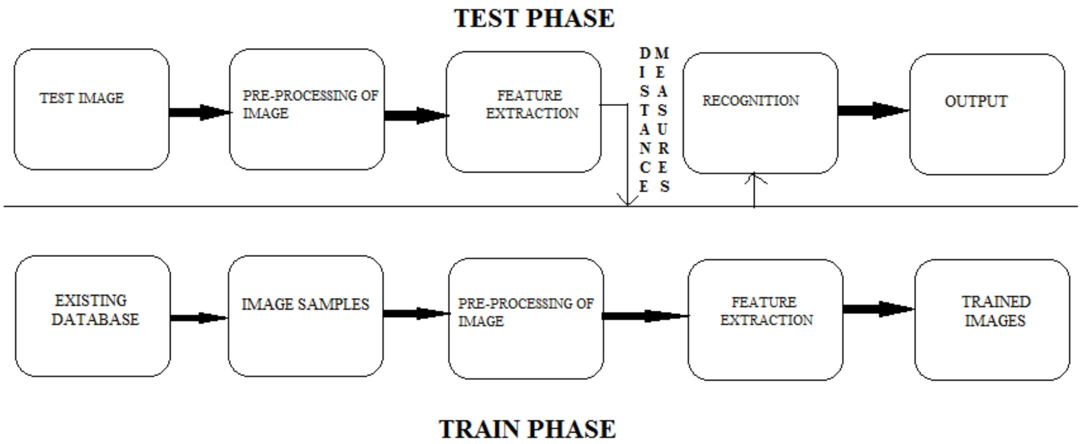

Figure 1 : overview of face recognition

In this thesis we used Gaussian mixtures and expectation maximization (EM)algorithm to explore more about Gaussian mixtures, this in together called mixture model method. We used discrete latent variables that is unobserved variables in mixture distribution to solve this fallowing stated issues, here goal is to give great approximation for multimodal distribution falls because of its essential unimodal property of Gaussian distribution furthermore extremely restricted in showing to enough amount of distributions. EM algorithm used for finding the greatest probability or maximum likelihood estimation in latent variables [19]. Latent variables further defined as allocated data points in mixture components.

A single and simple Gaussian distribution it's not able to include all main clumps of given dataset. Because of this we use Gaussian mixture distributions that is a combination of basic Gaussian distribution, this distribution will generate some random data points as shown in the below figure (Figure 2(b)). We can see that a linear combination of four Gaussians gives to complex densities and are further approximated using sufficient number of mixtures involving fine-tuning of their means and covariance. Gaussian Mixtures estimated by EM algorithm for four clusters are as shown in figure (Figure 2(b)).

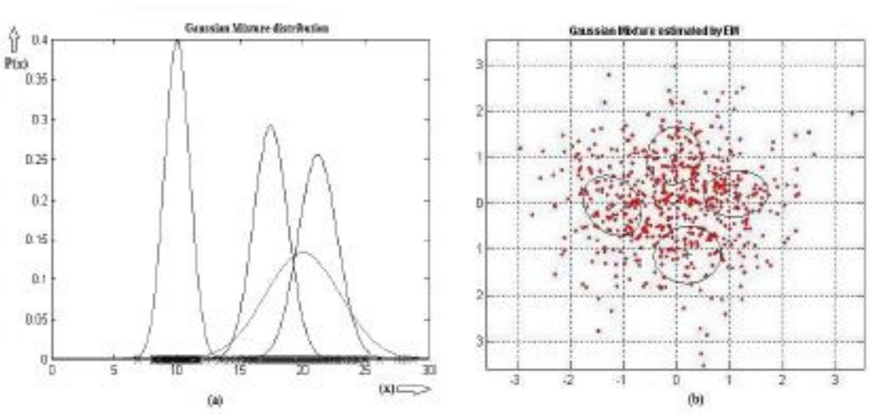

Figure 2 : showing of (a) Gaussian mixture distribution in one dimension showing 4 Gaussians.(b) Gaussian mixtures estimated by EM algorithm for 4 clusters [25].

While learning procedure of classification, we found centres and differences that is variances of the Gaussian component connected with mixing coefficients which are utilized as managing parameters. Gaussian mixtures used to break down complex likelihood distribution are planned as far as discrete latent variables and further characterized as the weighted sum of "M" Gaussian component as shown below:

$$p(\frac{x}{\lambda}) = \sum_{i=1}^{m} W_i * g(\frac{x}{\mu_i}, \Sigma i) \quad (1)$$

Here ‘x’ and ‘λ’ are dimensional data vectors that is denotes training samples and Eigen value, ‘Wi’ i=1,2….m, is mixture weights and function ‘g’ is Gaussian density components, ‘μ’ & ‘Σ’ are mean and co-variances separately.

2.1 EM ALGORITHM FOR GAUSSIAN MIXTURE MODEL

Expectation-maximization algorithm it assigns the data with some probability and estimates the maximum like hood data. It is a frequently iterative algorithm that start its process with some random estimates, then repeatedly proceeds to update the random estimates until convergence is defined. here it consists of E-step and M-step.

E-step (expectation step):

$$r_{ic}=\frac{\Pi c N(Xi;\mu c,\sum c)}{\sum c,\Pi c,N(Xi;\mu c,\sum c)} \tag{2}$$

Here μ_c is mean, $\sum_C$ is a covariance and Π is size of data.

For each datum X; this step computes the probability of each element of cluster 'c'.

It normalizes the sum to one, means it will collect more same type of data as a group called cluster.

If 'X' is very likelihood under the GMM, then it is considered as more weighted data.

M-step (maximization step):

It updates the parameters which like mean, covariance and size of data and also covariance reveals of how much two random variables can change together.

2.2 DENSITY MODELLING WITH GAUSSIAN MIXTURE MODEL

We considered the issue of modelling randomly produced information by mixture of three Gaussians in 2 dimensions with known priors (0.3, 0.5 and 0.2), centres (2, 3.5), (0, 0) and (0,2) and the variances (0.2, 0.5 and 1.0). Figure (Figure 3a and b) contain a scatter and surface plot (3-D perspective of density function) of the information individually. A Gaussian mixture model with three component trained using EM, 1-standard deviation hovers for the three parts of the blend model is appeared in figure (Figure 4).

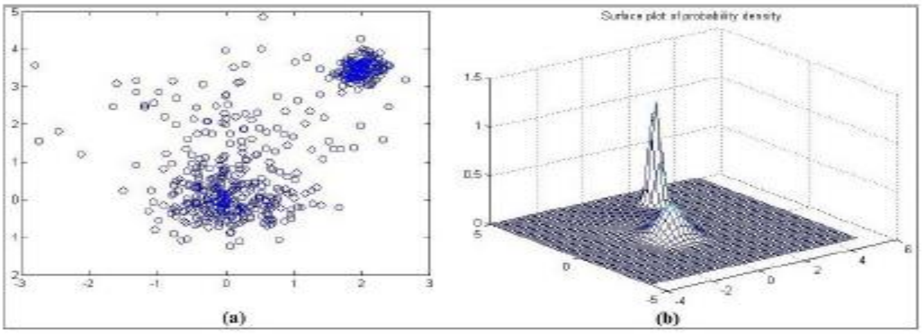

Figure 3 : illustration of (a) scatter plot of data (b surface plot) [25].

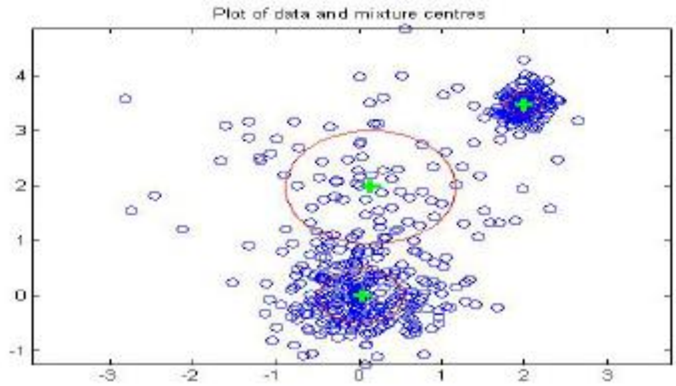

Figure 4: plot of data and mixture centres. [25]

It turns out to be important to derive basis images which are delicate to these high-order statistics, because in an object recognition issue a significant part of the data will be available in the high-order connections among pixel intensities. The objective of proposed model is to divide set of all classes into a few clusters utilizing Gaussian mixture model and to get transformation matrix for every cluster in decreased feature space to minimize the impact of higher dimension. In such manner, we research subspace-based algorithm on GMM, for example, PCA and FLD Mixture Models for effective recognition and recovery.

2.3 FEATURE EXTRACTION

Basically humans can able to identify the faces from early ages. We can recognize people faces even they are with beards, ageing, glasses etc. but for this process is more difficult for the computers. The main challenge in face recognition is to get the different information from the image that is feature extraction, this feature extraction is nothing but recognizing different information from an image or face. In future with the help of this feature we should able to recognize different faces with minimum error rates. And also this process should be efficient in memory usage and time consuming. Feature extraction can be expressed as dimensionality reducing after this feature extraction and selecting the feature from the extracted one.

Here dimensionality reduction is main task, and classifiers efficiency relies on as many samples for an image and complexity of classifiers. For feature extraction process there should as many as training samples for an image or class.as we keep feature as small as possible the classifier will work fast and use less memory and Features should be less redundant. Feature extraction is a process transforming the original image data to make a proper subspace from the original image. Then feature selection will select the best subset from the feature space. And it discards the non-relevant feature. We can sum up this process as first features will be extracted from the image and best subset of these features to be selected.

2.3.1 Principal component analysis

In face recognition system the main task is feature extraction. Although many approaches have been produced to solve this but most popular one is appearance based approaches. In appearance based approaches as we mentioned earlier of this thesis, it directly operates on the face images. PCA provides best tool for data analysis and pattern recognition, main algorithm in approach based method. Major task here is reducing dimensionality that is representing data in lower dimensionality space. That is Eigenfaces will be built based on PCA technique. A collection of a face image can be represented in form of small set of images. Finally, in more simplified way PCA used for data compression without losing information.

Principal component analysis is a procedure uses an orthogonal transformation to convert a set of M face images to a set of K uncorrelated variables called Eigenfaces. Here the number of Eigenfaces should always less than or equal to the number of original images, that Is K<M, then transformation is defined as the first Eigenface shows the best and more dominant features of training set of images. Each Eigenface shows most dominant features under constraints that is uncorrelated to preceding Eigenface. To reduce the calculation needed to find this Eigen faces the dimensionality of original training set is reduced before they calculated. Since Eigenfaces show the directions in the training set and each processing Eigen faces shows fewer directions and more noise, so for this only first few Eigenface are selected whereas rest if the last Eigenfaces are discarded.

These K Eigen face can safely represent the whole original training set because they depict the major features that are up the dataset. Therefore, each image in original training set can be represented in terms of these K Eigenfaces. Representing image this way reduces the number of values needed to recognize it from M to K. This makes the recognition process faster and more free of error caused by noise. PCA can be done by eigenvalue decomposition of a data covariance matrix. The result of PCA can be discussed in terms of Eigenface weights that is Eigenfaces proportions.

PCA, where Eigen faces that is principle components of a face in an image is extracted, encoded and made a compression with the database. Another now image anticipated onto face space basically by multiplying the difference between the image and average finally outcome result will be multiplied with individual Eigen vectors. The final outcome of this operation is the weighted sum of the each Eigenfaces representing the input face image and regarding the Eigenfaces as a primary set for face images.

Method:

1. Get some data let us take a set of data X1, X2, X3…. Xm is N*1

$$X=\frac{1}{M}\sum_{i=1}^{M} Xi \tag{3}$$

2. Make a subtraction of X values with its mean that is

$$\Phi_i=Xi-X_{mean}$$

3.Calculate the covariance matrix from a matrix with all mean that is

$$A= [\Phi_1,\Phi_2\ldots\ldots\Phi_m] \text{ x*y matrix}$$

$$C=\frac{1}{M}\sum_{i=1}^{M} \Phi_n\, \Phi_n= AA^T \tag{4}$$

4. For covariance matrix calculate the Eigen vectors and Eigen values,

an x*y matrix has X eigenvector and λ eigenvalues.

$Cx=\lambda x$ here x is set of Eigen vectors with its eigenvalue λ.

$$C - \lambda I = 0 \tag{5}$$

5. When eigenvectors found from covariance matrix, then we have to order those eigenvalues in high to low order. This gives components in order of significance. Choose eigenvalues with the highest order and it will form feature vector in a simple words choose component and form feature vector

6. Take the transpose of eigenvectors and multiple with original dataset hence gain the latest dataset.

2.3.2 The FLD mixture model

principal component analysis based on assumption of normality. Another feature extraction method that is fisher linear discriminant analysis proposed by R.A. Fisher it is a best known pattern recognition feature extraction technique. FLD based on class based projections. In practical FLD will project as by increase in class dimension of classes increases. Fallowing discussion given with Gaussian mixture model with FLD integration.

The fundamental task of FLD to preserve different class information, also by decreasing dimensionality. Keeping in mind the end goal to accomplish this, Fisher proposed maximizing the mean difference by normalizing the measure of inside class information scatter for better classification. Since PCA is an unsupervised procedure, the principle disadvantage is missing class label data. To find class label information and finding projections, Fisher Linear Discriminant Analysis considers the optimal projection w_{opt} which maximizes the ration of,

$$w_{opt} = \text{argmax}_W \frac{|W_T S_B W|}{|W_T S_W W|} \tag{6}$$

Because of within class and between class matrices are proportional to covariance matrices, here w-set of eigenvectors of sB and sW, which don't have any effect on solution with respect to function 'w' {i=1, 2...m} and here 'm' is largest Eigen value, sB is between class scatter matrices and sW is within class scatter matrices.

$$S_B = \sum_{i=1}^{c} Ni(\mu i - \mu)(\mu i - \mu)T \tag{7}$$

$$S_W = \sum_{i=1}^{c} \sum_{Xk=Xi} (Xk - \mu i)(Xk - \mu i)T \tag{8}$$

Ni =Number of training samples in class, C is number of distinct classes, μi is mean of sample belongs to class 'i', 'Xi' is the set of samples belongs to class 'i'.

In the beginning PCA is applied to within class to reduce the dimensionality reduction and then FLDA applied for between matrix to attain non-singular scatter. By applying PCA mixture model to group of means μi for each category with different Gaussian mixtures, we can able to achieve the FLD mixture model to use many transformation matrices in overall classes. Transformation matrix T_k and diagonal matric U_k and cluster mean C_k are acquired alongside as diagonal element with respect to the dth biggest Eigen value of covariance matrix. We can plan the scatter matrix for kth mixture can be formed with these set of outcomes as shown below

$$S_{BK} = T_K U_K T^T{}_K \tag{9}$$

$$s_{WK} = \sum_{l \in Lk} \frac{1}{nl} \sum_{x \in ci} (x - \mu i)(x - \mu i)T \tag{10}$$

By using these equations, we can able to compute transformation matrix W_k for kth mixture component with maximizing the fallowing function,

$$SJ_K(U) = \frac{|U^\wedge T S_B U|}{|U^\wedge T S_{W_K} U|} \tag{11}$$

Some basic steps are first we have to calculate between-class and within-class scatter then we have form a projection matrix by taking all eigenvectors of those matrices, finally a low subspace of lower dimensionality will be created by maximizing between-class scatter and by minimizing within-class scatter. Here FLD uses some set of images instead of a single image. With the help of multiple images system will be trained for each class then creating eigenvectors for Eigenfaces by preserving maximized difference between class and minimizing within class difference. By projecting single image of single class and considering their average then classes are projected into this space. Then input images projected into fisher space and compared to the average class projections.

FLD mixture model applied on a sample image with taken from ORL database with a size 32*32 for 4 Gaussian mixtures and we have got 40 major Eigen vectors as highlighted ten feature coefficient for every mixture. It will represent the similarity within class discrete data points features and different feature for between class feature data points we illustrated with an example as shown below figure (Figure 5) and it plots a different feature obtained from different mixture models. To end up this many Gaussian components used make a mixture of clusters and Expectation Maximization (EM) applied for this to get maximum likelihood data, hence we can accomplish exceptionally discriminating features in feature space. Further, while classification these features can be utilized in large image dataset to assign class label information

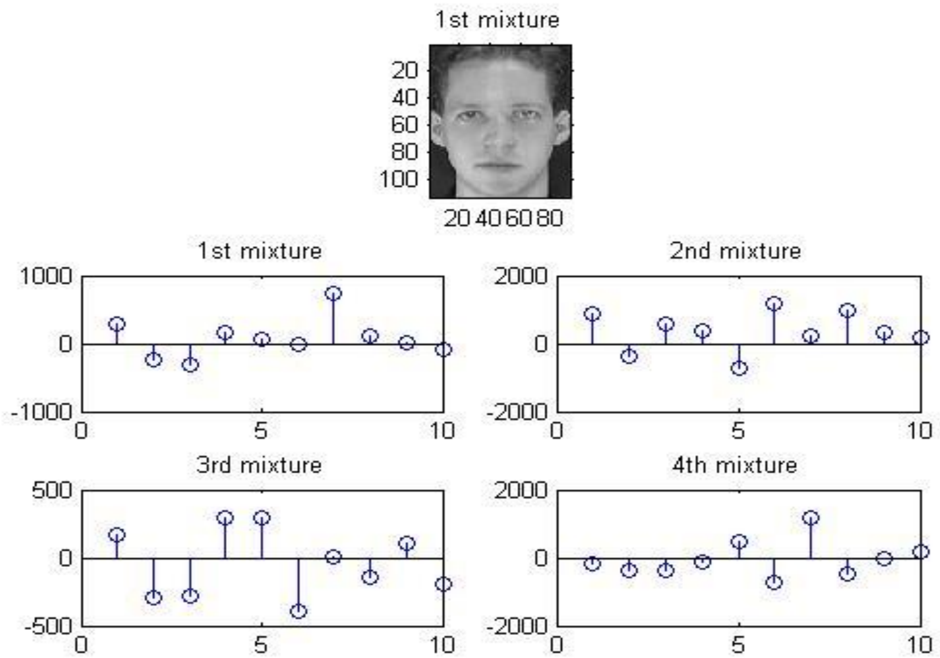

Figure 5: Example image and four Gaussian mixtures with ten principal components for each mixture.

2.4. CLASSIFICATION

In classification mainly we have two types as mentioned in the before section (literature survey), supervised classification and unsupervised classification. Execution of classification is based on some characteristics such as texture, density. A sample of classification will be as fallowing steps:

1. Class classification:

Classification class should be defined as of image characteristics.

2. Finding the object and extracting the features:

Finding object is detection of position of object in an image and other characteristics of an image and finding features from the object we detected like texture and some other properties.

3. Training data:

Selection of some training set which describe the whole pattern meaningful.

4. classification:

Here step we will make it into separate class of the detected features or characteristics.

Below figure (Figure 6) shows procedure of classification, Generally, classification will be carried out by the use of mathematical classification techniques.

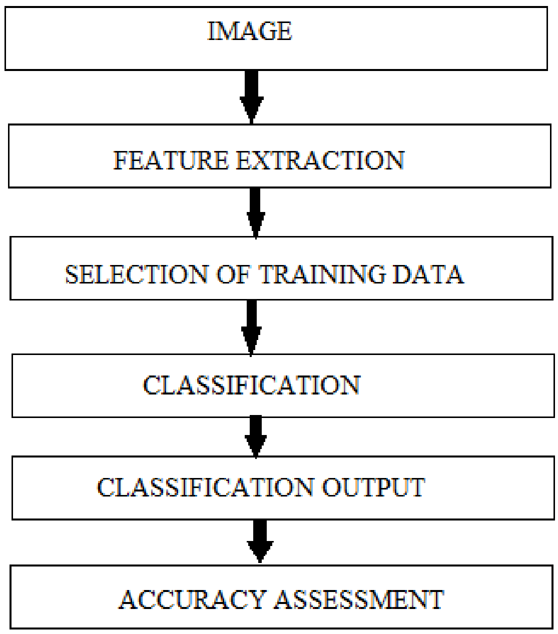

Figure 6 : Steps of classification

In this project we only concentrated on supervised classification with the distance or likeness measure methods. In this technique Similarity/distance is defined as level that separate the two objects logically by some attributes. There are many distance measure techniques by in our thesis we concentrated only on some selected one like Manhattan distance measure, Minkowski distance measure, Euclidean distance measure, modified squared Euclidean Distance measure, Correlation Coefficient based distance measure, and Angle-based distance measure techniques. Let X, Y be eigenfeature vectors of length n. for the fallowing distance between vectors will be calculated from the given below table (Table 1) distance measures,

Table 1: Distance measures with definitions.

DISTANCE MEASURE	DEFINITION
Euclidean distance	$D_{Euclidean}(x, y) = \sqrt{\sum_{i=1}^{l}(x_i - y_i)2}$
The Squared Euclidean Distance	$D_{SquEuclidean}(x, y) = \sum_{i=1}^{l}(x_i - y_i)2$
Manhattan or city-block distance	$D_{Manhattan}(x, y) = \sum_{i=1}^{l}(x_i - y_i)$
Minkowski distance	$D_{Minkowski}(x, y) = p\sqrt{\sum_{i=1}^{l}(x_i - y_i)p}$
Correlation Coefficient based distance	$D_{Minkowski}(x,y)= -\frac{\sum_{i=1}^{l} x_i y_i - \sum_{i=1}^{l} x_i \ \sum_{i=1}^{l} y_i}{((n \sum_{i=1}^{l}(x)2 - (\sum_{i=1}^{l} x_i)2)(n \sum_{i=1}^{l}(y)2 - (\sum_{i=1}^{l} y_i)2))1/2}$
Angle based distance	$D_{Angle}(x,y)=- \frac{\sum_{i=1}^{l} x_i y_i}{(\sum_{i=1}^{l}(x_i)2)1/2 \ \sum_{i=1}^{l}(y_i)2}$

1. Euclidean distance **-** Euclidean distance is widely used. This is also called as 'distance' instead of Euclidean distance. Squared root of distance between the pair of coordinates (Pythagoras theorem). For the test image and image to be identified in the database we used Euclidean distance to calculate the minimum distance between these two images. We calculate the minimum Euclidean distance between image projection and later image with the minimum Euclidean distance we will separate from the set of faces. If the distance between these two images are small, we will conclude that these images are similar and later we search for which is the most similar image in whole database. Which can be measured using fallowing formula,

$$D_{Euclidean}(x, y) = \sqrt{\sum_{i=1}^{l}(x_i - y_i)2} \tag{12}$$

2. The Squared Euclidean Distance - Without squared root for Euclidean distance we obtain,

$$D_{SquEuclidean}(x, y) = \sum_{i=1}^{l}(x_i - y_i)^2 \tag{13}$$

3. Manhattan or city-block distance - A city block distance among two vectors is computed by first obtaining the difference between every two corresponding vector elements, then adding all these differentiations. Here this metric travel from pixel to pixel in the pixel grid lines. Therefore, it is called as city block here diagonal moves are not allowed

$$D_{Manhattan}(x, y) = \sum_{i=1}^{l}(x_i - y_i) \tag{14}$$

4.Minkowski distance - Minkowski known as generalization of Euclidean distance and Manhattan distance measures because when p= 1 and p=2 it becomes respective distance measures. When, p=∞ it becomes Chebyshev distance measure.

$$D_{Minkowski}(x, y) = \sqrt[p]{\sum_{i=1}^{l}(x_i - y_i)^p} \tag{15}$$

5.Correlation Coefficient based distance- It measures similarities instead of measuring distance or dissimilarity. It represents the cosine angle between two vectors. Higher the angular separation it represents the similarities are more. The angular separation is [-1, 1] similar to cosine.

$$D_{Minkowski}(x, y) = -\frac{\sum_{i=1}^{l} x_i y_i - \sum_{i=1}^{l} x_i \sum_{i=1}^{l} y_i}{((n \sum_{i=1}^{l}(x)^2 - (\sum_{i=1}^{l} x_i)^2)(n \sum_{i=1}^{l}(y)^2 - (\sum_{i=1}^{l} y_i)^2))^{1/2}} \quad (16$$

6.Angle based distance

$$D_{Angle}(x, y) = -\frac{\sum_{i=1}^{l} x_i y_i}{(\sum_{i=1}^{l}(x_i)^2)^{1/2} \sum_{i=1}^{l}(y_i)^2} \tag{17}$$

2.4.1 Different Distance Measure Techniques

Similarity/distance is defined as a quantitative degree that enumerates the logical separation of two objects represented by a set of measurable attributes/characteristics. Few of the important properties that are most commonly found in distance computation tasks are enumerated as follows:

1. Symmetry, d (x, y) = d (y, x).
2. Positive definitive.

   $d(x,y) > 0$ for $x \neq y$ and $d(x,x)=0$
3. Triangle inequality, sometimes useful in making a metric

   $d(x,y) \leq d(x,z) + d(y,z)$
4. In addition, the following are usually described in some distance measure properties:

   d (x, y) should have a physically meaningful interpretation.

   d (x, y) should be efficiently computable

Classification has wide range of approaches in decision theory to identify the image. Classification algorithms assumes that the all images consist of one or more properties/features, and these features one of few different classes. If classes were determined by earlier by an expert, it is supervised classification and if the given data clustered by itself automatically it is unsupervised classification. Classification of an image break down the different features of an image and make into classes. It consists of mainly two phases training and testing as mentioned before. While training, trademark properties of image features are separated and with these a unique description of every category classification will be made that is training class. These features will be used during test phase to separate the features of image. It is important to describe training classes in classification process. Before, idea of likelihood distribution will be used in supervised method while in unsupervised method classification based automatic separation of the training data from the cluster algorithms.

Although there are some standards for building training classes that is free or independent here if any changes in the description made in one training class should not affect the other one that it should not change in values of another. Different features should have different representations. Another way is Dependable here all images inside training class should share some regular descriptions of that group. To build the parametric description in a better way is feature vector (X_1, $X_2 \ldots X_m$), here m is attributes which depict the image features of each image and training class. From these we can highlight each training class as subspace and features as processing points with respect to n classification dimensional space while classification it has to recognize which feature belong to which subspace.

For example, in below figure (Figure 7) consider we have to differentiate between three different types of objects based on some attributes. If we extract the feature from training images we can plot these result as in below figure. Now we should choose how to partition the image feature space in order to decide which feature vector belongs to which class. For this a simple way is minimum distance classifier. Each training class represented by mean vector,

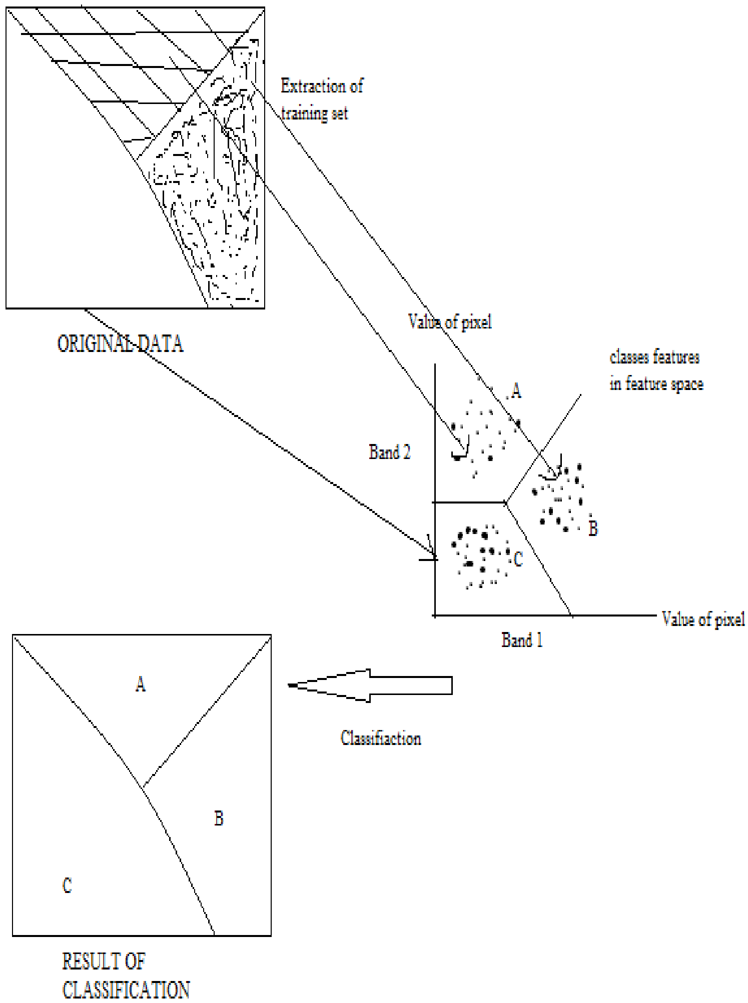

Figure 7 : concept of classification

$$m_j = \frac{1}{Nj} \sum_{x \in w_j} x \ , \quad j=1,2,3\ldots..M \tag{18}$$

Here Nj is training pattern vector of class wj, we have to calculate the mean of all those objects. Based on these results we have assigned any pattern x to class of its nearest prototype by measuring its proximity to each m_j. If we use Euclidean measure, then distance to the mean is given by

$$D_j(x) = \|x - m_j\| \quad \text{for } j=1, 2....M \tag{19}$$

Then we calculate the decision function that is a function it takes dataset as input and gives the decisions as output. This output decision depends on the problem that is in estimating will be answer for the estimation problems, for classification problems the decision is to classify the obtained data into different classes or category. Then decision boundary separates two classes W_i and Wj is given by,

$$d_i(x) - d_j(y) = 0 \tag{20}$$

In reality the minimum distance classifier works good when the distance between means are large when compared with the randomness of individual class with respect to its mean.

2.5 INDEXING

After performing classification, we will go for indexing to verify it belongs to which class. This is a post processing and analysis of data. In general, incoming data will be saved organized with an entry in this index. Each index entry includes a reference to the data. Pseudo code as given below,

```
function [Person_Index,View_Index] = INDEX(person_num,no_of_views)

View_Index = mod(person_num,no_of_views);
if (View_Index == 0)
    View_Index =no_of_views;
end
if ( mod(person_num,no_of_views)==0 )
    Person_Index = fix(person_num/no_of_views);
else
    Person_Index = fix(person_num/no_of_views)+1;
end
if (person_num == no_of_views)
    View_Index = no_of_views;
    Person_Index = 1;
End
```

3.EXPERIMENT AND RESULT DISCUSSION

Standard Experimental Procedure:

ORL database: ORL database which also known as AT&T is used in this work. The ORL database consist of 400images, which consist of 40 persons with 4 females and 36 males with 10 different images each [33]. Some properties of ORL database as shown in below table (Table 2).

Table 2 : ORL properties

Properties	**Description**
Number of subjects	40
Number of images	400
Purpose	Primarily used for face recognition
Static/video	Static
Gray/color	Gray
Format	.pgm
Resolution	92 * 112
Face Pose	Moderate pose variation (up and down, frontal view)
Facial expression	Three: natural, smiling, closed eye
Background	Dark homogenous
Accessories	Glasses

Train images/ dataset

In existing ORL database there are 10 images as in below sample figure (Figure 8).

Figure 8 : ORL database samples

To recognize the image in real time, system has to be trained before with the set of images or databases. Training images are given to the algorithm which calculate the principle components for each person. In training phase, we used only 1 image among all set of 10 images to train the system, basically, started with the different Gaussian mixture quantities and final chosen the number 4 as optimal for experiment. Some of the trained output of ORL database with the 10 principle component for each of 4 Gaussian mixture. As shown in below figure (Figure9).

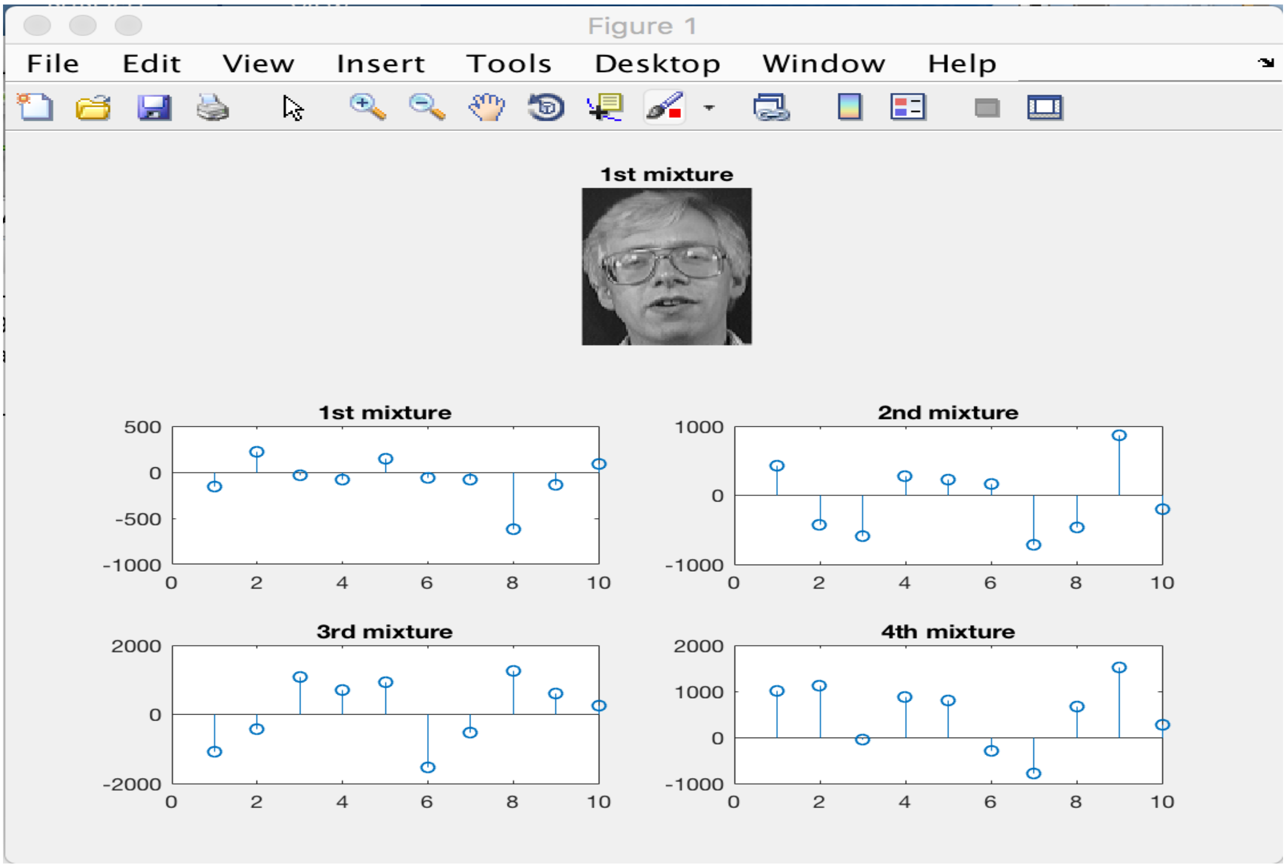

Figure 9 : Gaussian four mixtures with ten principle component for each mixture.

Test images/dataset

Once we finished the proper training of the system, then identification process can start. We are comparing the remaining nine images of a single person in ORL database. Test on a set of images which are having different views of the same person or image with the different images. Then system uses the certain number of features of the input image to calculate and compare with the trained images to identify the input face. Here we are showing best match of all images. As shown in below figure (10).

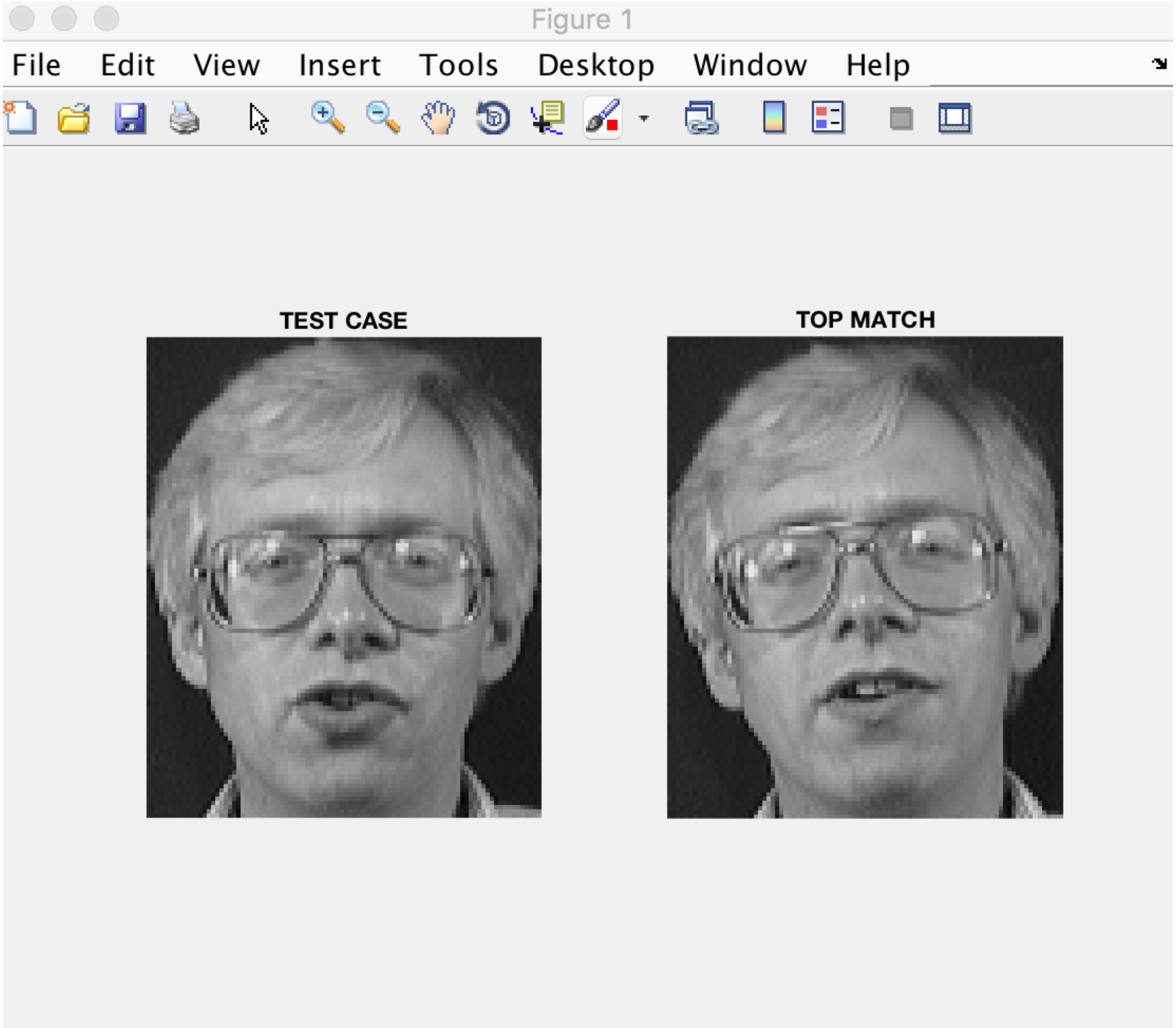

Figure 10 : Test case top match

Performance analysis

Analysis of performances in terms of recognition accuracy in total of 400 faces with respect to ORL database 400. Face recognition accuracy results for six different distance measure techniques. Analysis of FLD mixture on ORL database. Recognition accuracy is ratio of number of faces detected to the total no of faces tested. Some of the experimental result from other research works are given in below Figure (Figure 11).

Method	No of (train x test) images	Fea Vec Dimension	RA (in %)
(2D)2PCA[24]	5 x 5	27 x 26	90.5
SFA[25]	6 x 4	40	96.07
c Kmeans+NN[26]	6 x 4	-	96.97
PCA+LDA+SVM[27]	5 x 5	9	97.74
2FNN[28]	5 x 5	-	98.5
1D HMM+SVD[29]	5 x 5	-	99
Proposed System	5 x 5	25	100

Figure 11 : Best recognition accuracy on ORL database.

Although there are challenges in ORL database like varying in light intensity, shadow, person wearing glasses. We overcome with all this challenges with maximum recognition percentage when compared results of existing experiments. Although we are taking only single sample per person to train the face recognition system.

We achieved overall **98%** recognition accuracy for ORL database.

Table 3 : ORL database recognition accuracy

Sl.no	Distance measure	RA(%) , k=2	RA(%),k=3	RA(%),k=4
1	Manhattan distance	81%	96%	97.7 %
2	Minkowski Distance	80%	93%	97.7%
3	Modified Squared Euclidean Distance	75%	89%	92.2%
4	Euclidean Distance	80%	90%	95.5%
5	Correlation coefficient based Distance	58%	65%	70%
6	Angle-based Distance	75%	83%	85%

CONCLUSION

As the title of thesis suggests, the focus of our work can be explained as follows:

- We have introduced FLD-Mixture Models to investigate the efficient training method and influence of linear combination of Gaussian mixture distributions to observe latent variables in finding the following distribution over objects. An effort to integrate over all possible positions and orientations of an object in reduced feature space.
- and the choice of K = 4 mixtures is highly prejudiced and noteworthy to achieve better classification rate. Since Mixture models contains smaller number of components compared to training data, it results in faster evaluation during testing phase with increase in numerical stability.
- The use of different similarity distance measure techniques to obtain an average classification rate is first of its kind in the literature and affirmed to yield progressive result.

REFERENCE

[1] X. Lu, "*Image Anlysis for Face Recognition,*" Michigan State University.

[2] M. L. a. B. yuan, "*2DLDA; a statistical linear discriminant analysis for image matrix,"* Pattern recognition letters, 2005.

[3] J.-M. f. N. K. a. C. V. D. M. Laurenz Wiskott, "*face ecognition by Elastic bunch graph mathching*," In intelligent Biomatric technique in Fingerprint and face recoginition, 1999.

[4] j. C. V. J. b. martin Lades, "*Distortion Invariant Object Recognition in the Dynamic Link Architechture,"* IEEE Transcations on Computer , vol. 42, March 1993.

[5] G. E. a. C. T.F.Coots, *"Active Appearance Models,"* IEEE Transactions on pattern aalysis and machine intelligence , JUNE 2001.

[6] M. T. a. A.Pentland, *"Eigen faces for recognition*," Journal of Cognitive Neurosceince, 1991.

[7] R. Fisher, "The Statistical Utilization of Multiple Measurements," 1936.

[8] J. P. H. a. D. J. K. Peter N.Belhumeur, *"Eigenfaces vs Fisherfaces : Recognition using Class Specific Linear Projection,"* IEEE Transaction of Pattern Analysis and machine Intelligence, vol. Vol 90, 1997.

[9] J. a. T. M.S.Barlett, *"Face Recognition by Independent Component Analysis,"* IEEE Transaction on Neural networks, 2002.

[10] X. H. a. P. Niyogi, *"Face Recognition Using laplacianfaces*," IEEE Transactions on Pattern Analysis and machine Intelligence, March 2005.

[11] *"Recognising partially Occluded Epression,expression varient from single training image per person with SOM and soft Knn ensemble,"* IEEE Transaction on Neural network, 2005.

[12] L. Z. D. Z. Quan-Xue Gao, *"Sampled FLDA for Face recognition with Single traning image per person,"* Applied Mathematics and computations, 2008.

[13] P.J.Phillips, *"Support Vector Machine apllied to Face Recognition,"* Advance Neural Information processiong System, 1998.

[14] V. kumar, *"Correlation Pattren Recognition for face Recognition," p*roceedings of IEEE, 2006.

[15] X. t. Xiaoging wang, *"A unified framework for Subspace Face Recognition.,"* IEEE Transaction on Pattren Analysis and Machine Intelligence, Sep 2014.

[16] H. C. K. a. S. y. bang., *"Face Recognition using Mixture of eigenfaces,"* pattern reconition Letters., 2002.

[17] S. c. a. y. zhu, *"Sub Pattern based Principal Component Anlysis,"* Pattren recognition, 2004.

[18] M. j. Weilong chen, *"PCA and LDA in DCT domain,"* Pattern Recognition letters, JUlY 2015.

[19] J.-t. c. a. Chia-Chen, *"Dscriminant wavelet Faces and Nearest feature Classifiers for Face Recognition,"* IEEE transcation on pattern analysis and Machine inteligence, 2002.

[20] D. Z. Jian yang, "*2 Dimensional PCA a new Approach to appearance; based face representation and recognition,"* IEEE Transaction on Pattern analysis and machine intelligence, vol. VOL 26, Jan 2004.

[21] D. a. Z.h.zhou, *"2D PCA for Efficient face represention and recognition,"* Neurocomputing, 2005.

[22] h. k. a. S. Noushath.S, *"2D square LDA : An efficient approach for face recognition,"* pattern recognition , 2006.

[23] D. Z. a. S. chen, *"Diagonal principal componant analysis for face recognition,"* Pattern Recognition, 2006.

[24] X. L. X. W. W. H. a. S. j. Dacheng Tao, *"Supervized tensor Leraning,"* IEEE internation conferrence on Dta Minning, 2005.

[25] O. S. Adeoye, *“A survey of emerging Biometeric Technologies,”* International Journal of Computer Applications(0975-8887*)*, p. 01, November 2010.

[26] R. A. S.jyothi, *“An Effect of different distance measures for facial expressions,”* Dayanandha Sagar college of Engineering, Bangalore,India.

[27] A. M.Martinez, *"Recognizing Imprecisely Localized,partially occluded and Expression variant faces from single sample per class,"* IEEE transaction on pattern analysis and machine intelligence, vol. VOL 24, JUNE 2002.

[28] p. k. Surya kanth thyagi, *"Face Recognition Using Discrete Cosine Transform and Neraest Neighbour Discriminant Analysis,"* IACSI internationa journal of Engineering and technology, vol. VOL 04, JUNE 2012.

[29] s. l. j. y. Fengxi zong, "Orthogonalized fisher discriminant," Pattern recognition, vol. VOL 38, 2005.

[30] X. t. Xiaogang Wang, "Bayesian Face Recognition Based on Gaussian Model," IEEE 17th international conferrence on pattern recognition, 2004.

[31] S. s. S. A. Pooja kamavisdar, *International journal of advanced reasearch in computer and communication Engineering,* vol. VOL 2, JAN 2013.

[32] p. d. Siddharth Bhattacharya, Handbook of *research on Advance intelligence techniques and applications,* India: Information sceince referrence.

[33] AT&T Laboratories, Cambridge, 2002. [Online]. Available: http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[34] *"Computer vision,"* National sceince foundation, 17 march 2005. [Online]. Available: http://www.vision.caltech.edu/html-files/archive.html.

[35] D. spacek, *"Description of the collection of facial images,"* Computer vision sceince research project, 20 june 2008. [Online]. Available: http://cswww.essex.ac.uk/mv/allfaces/ .

[36] S. m. borrow, *"The Muct Face database,"* university of cape town, DEC 2008. [Online]. Available: http://www.milbo.org/muct/.

Appendix

Training phase:

```
%--------------------------Initialization---------------------------%
clear all;
clc;
Row = 32;
Col = 32;
NEV = 10;  % NEV being the number of the dominant eigen vectors.
K = 4;     %Number of Gaussian Mixtures.
no_of_persons = 40; %orl
no_of_views = 1;
total_faces = no_of_persons * no_of_views;
Source = 'C:\Users\sony\Desktop\ASIF\ORL_single_folder\image'; % .pgm format

%----------------------Gather the training samples  -------------------%
X = zeros(Row*Col,total_faces);
Count = 0;
for i=1:no_of_persons,
    for j=1:no_of_views,
        Path = [Source,int2str(i),'i_',int2str(j),'v.pgm'];
        Face = imread(Path);
        Face = double(Face);
        if((ndims(Face)==3) || (size(Face,3)==3))
         Face=rgb2gray(Face);
        end

        Face = imresize(Face,[Row,Col]);
        Count = Count + 1;
        X(:,Count) = reshape(Face,Row*Col,1);
    end
end
Avg = mean(X,2);
%--------------------EM algorithm to learn mixture of eigenface(3)--------%
X=double(X);
[W,M,V,L] = EM_GM_fast(X',K,[],[],1,[]);
Ci = M';
[W,M,V,L] = EM_GM_fast(X,K,[],[],1,[]);
%----------------------- Extract K mixture of Eigenfaces---------------%

Evecs = zeros(Row*Col,NEV,K);
for i=1:K,
    [SEvec,SEval] = eig(V(:,:,i));
    New_X = X - repmat(Ci(i,:)',[1,total_faces]);
    col_index = total_faces;
    for j=1:NEV,
        Dom_Eig_Vect = SEvec(:,col_index);
        Evecs(:,j,i) = ((1.0/sqrt(SEval(col_index,col_index)))*New_X)*Dom_Eig_Vect;
        col_index = col_index - 1;
    end
end

                    %% Feature Extraction step
Sb = zeros(Row*Col);
for i=1:no_of_persons,
    %fprintf(1,'\nSb for %d',i);
    Mu_i = mean(X(:,(i-1)*no_of_views+1:(i-1)*no_of_views+no_of_views),2) - Avg;
```

```
    Sb = Sb + (Mu_i * Mu_i');
end
Sw = zeros(Row*Col);
for i=1:no_of_persons,
    Mu_i = mean(X(:,(i-1)*no_of_views+1:(i-1)*no_of_views+no_of_views),2);

    for j=1:no_of_views,
        Diff = X(:,(i-1)*no_of_views+j) - Mu_i;
        Sw = Sw + (Diff * Diff');
    end
end
Count=0;
FM=zeros(NEV,total_faces,K);
P_FLD1=zeros(1024,10,K);


for J =1:K,
    Count=Count+1;
    P_PCA = Evecs(:,:,J);
    Sbb = P_PCA.' * Sb * P_PCA;
    Sww = P_PCA.' * Sw * P_PCA;
    % clear Sb Sw X;
    [V1 D1] = eig(Sbb,Sww);
    Ds = diag(D1);
    [tmp ndx] = sort(abs(Ds));
    ndx = flipud(ndx);
    EigVecs = P_PCA*V1(:,ndx);
%     clear V1 D1 Ds;
    P_FLD = EigVecs(:,1:NEV);
    P_FLD = P_FLD./repmat(sum(P_FLD.^2).^0.5,Row*Col,1);
    P_FLD1(:,:,J) = P_FLD;
    New_X = X - repmat(Ci(J,:)',[1,total_faces]);
    FM(:,:,J) = P_FLD'*New_X;
end


a=FM(:,1,1);
b=FM(:,1,2);
c=FM(:,1,3);
d=FM(:,1,4);

image=imread('C:\Users\sony\Desktop\ASIF\ORL_single_folder\image2i_1v.pgm');
subplot(3,2,1:2), imshow(image), title('1st mixture');
subplot(3,2,3), stem(a), title('1st mixture');
subplot(3,2,4), stem(b), title('2nd mixture');
subplot(3,2,5), stem(c), title('3rd mixture');
subplot(3,2,6), stem(d), title('4th mixture');

%-----------------------------------------------------------------
```

Testing Phase:

```
 II_val=40;
 diff1=10;
 diff2=1;
 diff=diff1-diff2;

%---------------------------------Testing (1)------------------------------%
clc;
NEV=1;
Source1 = 'C:\Users\sony\Desktop\ASIF\ORL_single_folder\image';
while (NEV <= 10)
Success = 0;
```

```
Failure = 0;
for II = II_val,  % change1
   for JJ = 2:diff1, %no_of_samples, %(no_of_views+1):no_of_samples,     % change1
   Path = [Source1,int2str(II),'i_',int2str(JJ),'v.pgm'];
      Test1 = imread(Path);
      Test_Image = imread(Path);
      Test_Image = double(Test_Image);
      if((ndims(Test_Image)==3) || (size(Test_Image,3)==3))
      Test_Image = rgb2gray(Test_Image);
      end
      Test = Test_Image;
      Test_Image = imresize(Test_Image,[Row,Col]);
      Test_Image = reshape(Test_Image,Row*Col,1);
      FLAG = zeros(1,K);
      for J=1:K,
         Evs = Evecs(:,:,J);
         Diff_Face = double(Test_Image) - Ci(J,:)';
         Fi = P_FLD1(:,:,J)'*(Diff_Face);
         Max_Sim = 999999999999.99;
         for i=1:total_faces,
            sim=0;
            for j=1:NEV,
             sim=sim+abs(Fi(j,1)-FM(j,i,J)); % Manahattan distance
            end
            if (sim < Max_Sim)
               Max_Sim = sim;
               Person_Num = i;
            end
         end

         [Person_Index,View_Index] = INDEX(Person_Num,no_of_views);
          FLAG(1,J) = Person_Index;
       end
       Vect = find(FLAG == II);
       if (length(Vect)>= 1)
          Success = Success + 1;
        figure(Success)
        Recog_Path = strcat(Source,int2str(Person_Index),'i_',int2str(View_Index),'v.pgm');
       Recog_Image = imread(Recog_Path);
       subplot(1,2,1),imshow(Test1),title('TEST CASE')
       subplot(1,2,2),imshow(Recog_Image),title('TOP MATCH');
      else
       Failure = Failure + 1;
      end
   end
end
NEV = NEV + 4;
disp('Recognition Accuracy for manhattan Distance');
RA = (Success/diff)*100   % change1
end
%---------------------------------Testing (2)-----------------------------%-
% no_of_samples = 10;
NEV=1;

while (NEV <= 10)
Success = 0;
Failure = 0;
for II = II_val,   % change2
   for JJ = 2:diff1, %no_of_samples, %(no_of_views+1):no_of_samples,    % change2

      Path = [Source1,int2str(II),'i_',int2str(JJ),'v.pgm'];
      Test_Image = imread(Path);
      Test_Image = double(Test_Image);
```

```
        if((ndims(Test_Image)==3) || (size(Test_Image,3)==3))
        Test_Image = rgb2gray(Test_Image);
        end
        Test = Test_Image;
        Test_Image = imresize(Test_Image,[Row,Col]);
         Test_Image = reshape(Test_Image,Row*Col,1);
        FLAG = zeros(1,K);
        for J=1:K,
          Evs = Evecs(:,:,J);
            Diff_Face = double(Test_Image) - Ci(J,:)';
          Fi = P_FLD1(:,:,J)'*(Diff_Face);
            Max_Sim = 999999999999.99
            for i=1:total_faces,
              sim=0;
             for j=1:NE
             sim = sim + abs(Fi(j,1)-FM(j,i,J)).^2; %Minkowski Distance
                end
                   sim = sim.^0.5;
                if (sim < Max_Sim
                  Max_Sim = sim
                    Person_Num = i;
                end
          end
          [Person_Index,View_Index] = INDEX(Person_Num,no_of_views);
            FLAG(1,J) = Person_Index;
        end
       Vect = find(FLAG == II);
        if (length(Vect)>= 1)
          Success = Success + 1;
)
        else

            Failure = Failure + 1;
        end
    end
end

NEV = NEV + 1
disp('Recognition Accuracy for minkowski Distance');
RA = (Success/diff)*100
 End

%---------------------------------Testing (3)------------------------------
% no_of_samples = 10;
NEV=1;
while (NEV <= 10)
Success = 0;
Failure = 0;
for II = II_val,% change1
    for JJ = 2:diff1, %no_of_samples, %(no_of_views+1):no_of_samples,% change1
        Path = [Source1,int2str(II),'i_',int2str(JJ),'v.pgm'];
        Test_Image = imread(Path);
        Test_Image = double(Test_Image);
        if((ndims(Test_Image)==3) || (size(Test_Image,3)==3))
        Test_Image = rgb2gray(Test_Image);
        end
        Test = Test_Image;
        Test_Image = imresize(Test_Image,[Row,Col]);
        Test_Image = reshape(Test_Image,Row*Col,1);
        FLAG = zeros(1,K);
        for J=1:K,
            Evs = Evecs(:,:,J);
            Diff_Face = double(Test_Image) - Ci(J,:)';
```

```
            Fi = P_FLD1(:,:,J)'*(Diff_Face);
            Max_Sim = 999999999999.99;
            for i=1:total_faces,
                sim=0;b=0;c=0;
              for j=1:NEV,
                 b = b + Fi(j,1).^2;
                 c = c + FM(j,i).^2;
                 sim = sim+(Fi(j,1)-FM(j,i)).^2; %Modified Squared Euclidean Distance
                end
                sim = sim/(b*c)
                if (sim < Max_Sim)
                    Max_Sim = sim;
                    Person_Num = i;
                end
            end
            [Person_Index,View_Index] = INDEX(Person_Num,no_of_views);
            FLAG(1,J) = Person_Index;
        end
        Vect = find(FLAG == II);
        if (length(Vect)>= 1)
            Success = Success + 1;
        else
            Failure = Failure + 1;
        end
    end
end

NEV = NEV + 1
 disp('Recognition Accuracy for Modifies Squared Eucledian Distance');
 RA = (Success/diff)*100
end

%----------------------------------Testing (4) ------------------------------%
NEV=1;

while (NEV <= 10)
Success = 0;
Failure = 0;
for II = II_val,% change1
    for JJ = 2:diff1, %no_of_samples, %(no_of_views+1):no_of_samples,% change1
        Path = [Source1,int2str(II),'i_',int2str(JJ),'v.pgm'];
        Test_Image = imread(Path);
        Test_Image = double(Test_Image);
        if((ndims(Test_Image)==3) || (size(Test_Image,3)==3))
        Test_Image = rgb2gray(Test_Image)
        end
        Test = Test_Image;
        Test_Image = imresize(Test_Image,[Row,Col]);
        Test_Image = reshape(Test_Image,Row*Col,1);
        FLAG = zeros(1,K)
        for J=1:K,
            Evs = Evecs(:,:,J);
            Diff_Face = double(Test_Image) - Ci(J,:)';
            Fi = P_FLD1(:,:,J)'*(Diff_Face);
            Max_Sim = 999999999999.99;
            for i=1:total_faces,
                sim=0;
                for j=1:NEV,
                 sim = sim + abs(Fi(j,1)-FM(j,i)).^2; %Euclidean Distance
                end
                   sim = sqrt(sim);
                if (sim < Max_Sim)
                    Max_Sim = sim;
```

```
            Person_Num = i;
          end

        end

        [Person_Index,View_Index] = INDEX(Person_Num,no_of_views);
        FLAG(1,J) = Person_Index;

      end
      Vect = find(FLAG == II);
      if (length(Vect)>= 1)
        Success = Success + 1
      else

        Failure = Failure + 1;
     end
   end
end

NEV = NEV + 1
disp('Recognition Accuracy for eucledian Distance');
RA = (Success/diff)*100

end



%---------------------------------Testing (5)----------------------------%
% no_of_samples = 10
NEV=1;
 while (NEV <= 10)
Success = 0;
Failure = 0;
for II = II_val,% change1
   for JJ = 2:diff1, %no_of_samples, %(no_of_views+1):no_of_samples,% change1
      Path = [Source1,int2str(II),'i_',int2str(JJ),'v.pgm'];
      Test_Image = imread(Path);
      Test_Image = double(Test_Image);
      if((ndims(Test_Image)==3) || (size(Test_Image,3)==3))
      Test_Image = rgb2gray(Test_Image);
      end
      Test = Test_Image;
     Test_Image = imresize(Test_Image,[Row,Col]);
      Test_Image = reshape(Test_Image,Row*Col,1);
      FLAG = zeros(1,K);
      for J=1:K
         Evs = Evecs(:,:,J);
       Diff_Face = double(Test_Image) - Ci(J,:)';

         Fi = P_FLD1(:,:,J)'*(Diff_Face);
         Max_Sim = 999999999999.99;
         for i=1:total_faces,
         sim=0; d=0;e=0;f=0;g=0;h11=0;ii=0;jj=0
            for j=1:NEV
                d = d + (Fi(j,1)*FM(j,i))
                e = e + Fi(j,1);
                f = f + FM(j,i);
                g = g + (Fi(j,1).^2);

 %-------------- Correlation coefficient based Distance………………...…..%

                h11 = h11 + Fi(j,1);
                ii = ii + (FM(j,i).^2)
```

```
                jj = jj + FM(j,i)
            end
sim = (((NEV*d) - (e*f)) / sqrt( ((NEV*g) - (h11.^2)) * ((NEV*ii) - (jj.^2))) );
             sim = -(sim);
            if (sim < Max_Sim)
              Max_Sim = sim;
                Person_Num = i;
            end
          end
          [Person_Index,View_Index] = INDEX(Person_Num,no_of_views)
          FLAG(1,J) = Person_Index;
        end
        Vect = find(FLAG == II);
        if (length(Vect)>= 1)
          Success = Success + 1;

        else

          Failure = Failure + 1;
        end
    end
end
NEV = NEV + 1
disp('Recognition Accuracy for correlation Distance');
RA = (Success/diff)*100
end

%--------------------------------Testing (6)-----------------------------

% no_of_samples = 10;

NEV=1;

while (NEV <= 10)
Success = 0;
Failure = 0;
for II = II_val,% change1
    for JJ = 2:diff1, %no_of_samples, %(no_of_views+1):no_of_samples,% change1
        Path = [Source1,int2str(II),'i_',int2str(JJ),'v.pgm'];
        Test_Image = imread(Path);
        Test_Image = double(Test_Image);
        if((ndims(Test_Image)==3) || (size(Test_Image,3)==3))
        Test_Image = rgb2gray(Test_Image);
        end
        Test = Test_Image;
        Test_Image = imresize(Test_Image,[Row,Col]);
        Test_Image = reshape(Test_Image,Row*Col,1);
        FLAG = zeros(1,K);
        for J=1:K,
          Evs = Evecs(:,:,J);
          Diff_Face = double(Test_Image) - Ci(J,:)';

          Fi = P_FLD1(:,:,J)'*(Diff_Face);
          Max_Sim = 999999999999.99;
          for i=1:total_faces,
            sim=0;a=0;b=0;c=0;
            for j=1:NEV,
              a= a + (Fi(j,1)*FM(j,i));
              b = b + Fi(j,1).^2;           %Angle-based Distanc
              c = c + FM(j,i).^2;
            end
             sim = a / sqrt(b*c);
             sim = -(sim);
```

```
            if (sim < Max_Sim)
                Max_Sim = sim;
                Person_Num = i;
            end
        end

        [Person_Index,View_Index] = INDEX(Person_Num,no_of_views);

        FLAG(1,J) = Person_Index;
    end
    Vect = find(FLAG == II);
    if (length(Vect)>= 1)
        Success = Success +1;
    else
       Failure = Failure + 1;
    end
  end
end

NEV = NEV + 1
disp('Recognition Accuracy for Angle based Distance');
RA = (Success/diff)*100
end
```