In the previous chapter, a brief literature review on conventional and soft computing techniques for face recognition has been discussed. In this chapter the theory and experimental results related to PCA and FLDA are discussed.

Principal Components Analysis (PCA) is an appearance based technique used widely for the dimensionality reduction and it records a great performance in face recognition. PCA based approaches typically include two phases: training and classification (Draper et al 2003). In the training phase, an Eigen space is established from the training samples using PCA and the training face images are mapped to the Eigen space for classification. In the classification phase, an input face is projected to the same Eigen space and is classified by an appropriate classifier.

The PCA approach is used to reduce the dimension of the data by means of data compression basics and reveals the most effective low dimensional structure of facial patterns. This reduction in dimensions removes information that is not useful and precisely decomposes the face structure into orthogonal (uncorrelated) components known as Eigen faces. Actually in
PCA, each face image may be represented as a weighted sum of the eigenfaces i.e., feature and vector which are stored in a one dimensional (1D) array.

3.1.2 Theory of PCA

In PCA, Eigenfaces recognition derives its name from the German prefix ‘eigen’, meaning ‘own’ or ‘individual’. PCA is a popular technique used to derive a set of features for both face representation and recognition (Kirby and Sirovich 1990). This process captures most of the face variation into a set of images. Basic theory of PCA applied to training images and test images is presented as shown in Figure 3.1.

![Figure 3.1 Basic theory of PCA](image-url)
The main idea of using PCA for face recognition is to form Eigen space projection which expresses the large 1-D vector of pixels of 2-D facial image into the compact principal components of the feature space. Eigen space is calculated by identifying the eigenvectors of the covariance matrix derived from a set of facial image vectors. PCA classification is multi-modal, i.e. resulting in a vector of independent measures that could be compared with other vectors in a database.

PCA is based on Eigen value and Eigenvector analysis of the covariance matrix. In a covariance matrix the diagonal values are the variance for each dimension and the off-diagonal elements are the covariance between measurement types. Also large term in the diagonal correspond to interesting dimensions, whereas large values in the off-diagonal correspond to high correlations (redundancy). Actually all eigenvectors are perpendicular, i.e. at right angle with each other.

An eigenvector of a matrix is a vector such that, if multiplied with the matrix, the result is always an integer multiple of that vector. This integer value is the corresponding Eigen value of the eigenvector. Principal components are orthogonal, and their covariance matrix is diagonal. The reduction in dimensionality is achieved by taking only the dominant principal components and the transformation significantly reduces the computation time.

Eigen faces are ranked in the decreasing order of their Eigen values. The eigenvectors corresponding to the larger Eigen values are the most significant than those with smaller Eigen values. A test image is compared against a training image by measuring the distance between their respective feature vectors. Recognition is performed by projecting a new image into the subspace spanned by the Eigen faces (‘face space”) and then
classifying the face by comparing its position in face space with the positions of known individuals.

Mathematically, PCA simply finds the principal components of the distribution of faces or the eigenvectors of the covariance matrix of the set of face images by treating an image as a vector in a very high dimensional space. The eigenvectors are ordered, each one accounting for a different amount of the variations among the face images. Each of the individual faces can be represented exactly in terms of linear combinations of the Eigen faces.

A sample face image is represented in matrix form, according to the image size (by its pixel values) as presented below.

\[
X = \begin{bmatrix}
253 & \ldots & 255 \\
\vdots & \ddots & \vdots \\
245 & \ldots & 251 \\
\end{bmatrix}
\]

The PCA algorithm is as follows:

Step 1: Acquire an initial set of face images (the training set and test set) and form its feature vector representation.

Step 2: Calculate the covariance matrix as per equation number 3.3.

Step 3: Form the Eigen-faces according to the highest Eigen value of the covariance matrix.

Step 4: Classify the given face image, according to the Euclidean distance and threshold values.
PCA algorithm consists of mathematical calculation of shape vector, mean vector of all face images, covariance matrix and Eigen vectors. The shape vector of a face image is the one dimensional vector of pixels of two dimensional facial images into the compact principal components of the feature space. The mean vector consists of the means of each variable and the variance-covariance matrix consists of the variances of the variables along the main diagonal and the covariance’s between each pair of variables in the other matrix positions.

Principal Component Analysis is mainly used to form a lower dimensional space, where each face is described by a shorter vector. The Eigen vectors, also called Eigen faces, can be considered as a set of features that together characterize the variation between face images. Once a set of Eigen faces is computed, a face image can be approximately reconstructed using a weighted combination of the Eigen faces. The projection weights form a feature vector for face representation and recognition.

When a new test image is given, the weights are computed by projecting the image onto the Eigen face vectors. The classification is then carried out by comparing the distances between the weight vectors of the test image and the images from the database. Conversely, using all of the Eigen faces extracted from the original images, one can reconstruct the original image from the Eigen faces so that it matches the original image exactly.

Eigen space projects images into a subspace such that the first orthogonal dimension of this subspace captures the greatest amount of variance among the images and the last dimension of this subspace captures the least amount of variance among the images. The computational time of Eigen space projection is directly proportional to the number of eigenvectors used to create the Eigen space. Therefore, by considering the dominant Eigen values, some part of eigenvectors is removed and computational time is
decreased. Furthermore, by removing additional eigenvectors that do not contribute to the classification of image, performance can be improved.

The basic block diagram of PCA analysis is as shown in Figure 3.2.

![Figure 3.2 PCA Approach for Face Recognition](image)

Actually, the Eigen faces are the Eigen functions of the several covariances of ensembles of faces which are based on information theory approach of coding and encoding face images (Moghaddam et al 1998). These emphasize the significant local and global features. The original space of an image is just one of infinitely many spaces in which the image can be examined. The specific subspace is the subspace created by the eigenvectors of the covariance matrix of training data. Thus the Eigen space optimizes variance among the images.
3.1.3 Recognition Process

The whole recognition process involves two steps:

a. Initialization process
b. Recognition process

The Initialization process of PCA:

1) Acquire the initial set of face images called as training set.

2) Calculate the eigen faces from the training set, keeping only the highest eigen values. As new faces are added, the eigen faces can be updated.

3) Calculate the corresponding distribution in M-dimensional weight space for each known individual, by projecting their face images on to the “face space”.

These operations can be performed from time to time whenever there is a excess operational capacity. This data is used in the further steps eliminating the overhead of re-initializing, decreasing execution time thereby increasing the performance of the entire system.

Having initialized the system, the next process recognition process involves following steps:

1) Calculate a set of weights based on the input image and the eigenfaces by projecting the input image onto each of the eigenfaces
2) Determine if the image is a face at all (known or unknown) by checking to see if the image is sufficiently close to a “free space”.

3) If it is a face, then classify the weight pattern as either a known person or as unknown.

3.1.4 Algorithm for PCA Based Face Recognition

Let the training set of images be $\Gamma_1, \Gamma_2, \ldots, \Gamma_M$. The average face of the set is defined by

$$\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n$$  \hspace{1cm} (3.1)

Each face differs from the average by vector

$$\Phi_i = \Gamma_i - \Psi$$  \hspace{1cm} (3.2)

The co-variance matrix is formed by

$$C = \frac{1}{M} \sum_{n=1}^{M} \Phi_n \Phi_n^T = A^T A$$ \hspace{1cm} (3.3)

where the matrix $A = [\Phi_1, \Phi_2, \ldots, \Phi_M]$.

This set of large vectors is then subject to principal component analysis, which seeks a set of $M$ orthonormal vectors $u_1 \ldots u_M$. To obtain a weight vector $\Omega$ of contributions of individual eigen-faces to a facial image $\Gamma$, the face image is transformed into its eigen-face components projected onto the face space by a simple operation.
\[ \omega_k = u_k^T (\Gamma - \Psi) \]  

(3.4)

For \( k=1,\ldots, M' \), where \( M' \leq M \) is the number of eigen-faces used for the recognition. The weights form vector \( \Omega = [ \omega_1, \omega_2, \ldots, \omega_{M'} ] \) that describes the contribution of each Eigen-face in representing the face image \( \Gamma \), treating the eigen-faces as a basis set for face images. The simplest method for determining which face provides the best description of an unknown input facial image is to find the image \( k \) that minimizes the Euclidean distance \( \varepsilon_k \).

\[ \varepsilon_k = \| (\Omega - \Omega_k) \|^2 \]  

(3.5)

where \( \Omega_k \) is a weight vector describing the \( k \)th face from the training set. A face is classified as belonging to person \( k \) when the minimum \( \varepsilon_k \) is below some chosen threshold \( \varepsilon_\beta \), otherwise, the face is classified as unknown.

### 3.1.5 Procedure for Computing Eigen Faces

In this work, the shape vector is formed for Yale database of size \( 77760 \times 1 \), where 77760 is the size of one face image \( 243 \times 320 \) from Yale database and 1 represents single dimensional vector form of the facial image. Then the covariance matrix, dominated Eigen values and its vectors of covariance matrix are derived. Multiplication of this dominant vector and original shape vector gives the reduced dimensional vector. All these vector computations are carried out using “MATLAB-VERSION 7.0”.

The procedure for computing the Eigen value of the given image is presented below. Let \( X \in \mathbb{R}^N \) be a random vector representing a shape of an image, where \( N \) is the dimensionality of the corresponding shape or image space. For a shape, the vector consists of the coordinates of the representing face shape. This vector represents the main features of the face in matrix
form. This X vector of size [77760*1], for one such sample image is shown below:

By equation (3.3), covariance matrix is calculated with the size of [49*49] vector. Let \( \begin{bmatrix} u_1, u_2, \ldots, u_n \end{bmatrix} \in \mathbb{R}^{N \times N} \) be the Eigen vectors of covariance matrix \( C \) and \( (\lambda_1, \lambda_2, \ldots, \lambda_n) \) be the corresponding Eigen values. Then \( C \cdot u_k = \lambda_k \cdot u_k \).

The dimensionality of the transformed space is reduced to “m”. The Principal Component Analysis of a random vector \( X \) factorizes the covariance matrix \( C \) into the following form:

\[
C = A \Lambda A^T
\]

(3.6)

where \( A = [u_1, u_2, \ldots, u_n] \in \mathbb{R}^{N \times N} \) is the orthonormal eigen vector matrix of \( C \).

Here \( \Lambda = \text{diag} (\lambda_1, \lambda_2, \ldots, \lambda_n) \in \mathbb{R}^{N \times N} \) is the diagonal eigen value matrix with diagonal elements in decreasing order (\( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \)). Actually, the covariance matrix is of size [49*49]. After factorization, the eigen vectors corresponding to the 42 dominant eigen values of the covariance matrix is designated as “P” and it has a size of [49*42]. This P vector is the dominant eigen vector matrix used for dimensionality reduction using equation (3.6).

An important property of PCA is decorrelation, i.e., the components of the transformed data are correlated since the covariance matrix of \( X \) is diagonal. Following this property, an immediate application of PCA is the dimensionality reduction. The lower dimensional vector “\( Y \in \mathbb{R}^{m} \)” captures the most expressive features of the original data \( X \). This vector \( X \) is multiplied with the transpose of dominant Eigen vector \( P \) and is given as

\[
Y = P^T X
\]

(3.7)
where \( P = [u_1, u_2, \ldots, u_m], m < N \) and \( P \in \mathbb{R}^{N \times m} \). Thus the image size is reduced as \([42 \times 1]\), by considering the dominant eigen values of “\( u \)”. For this case, where 42 dominant Eigen values are considered, \( P \) is of size \([49 \times 42]\) and, as per equation (3.7), \( Y = P^T X = [42 \times 49] \times [49 \times 1] = [42 \times 1] \).

Thus the shape vector’s dimensionality is reduced from \([77760 \times 1]\) to \([42 \times 1]\). The important property of principal component analysis is its optimal signal reconstruction in the sense of minimum mean square error where only a subset of principal components is used to represent the original signal. The performance of PCA depends on the task statement, the subspace distance metric, and the number of subspace dimensions retained.

### 3.1.6 Experimental Results of PCA

The results of PCA based face recognition for different databases like ORL, Yale, FERET and Real time are presented as shown in Figures 3.3, 3.4, 3.5 and 3.6 respectively.
Figure 3.3 Results of PCA for ORL Database
Figure 3.4 Results of PCA for YALE database
Figure 3.5 Results of PCA for FERET Database
Figure 3.6 Results of PCA for real time database images
3.1.7 Advantages, Disadvantages and Applications of PCA

Principal Component Analysis (PCA) is a standard tool in modern data analysis in diverse fields from neuroscience to computer graphics because it is a simple, non-parametric method for extracting relevant information from the confusing data sets. Main advantages of PCA are its ability to recognize face images quickly and its easy implementation. Therefore, a higher correct recognition rate, a better efficiency and less running time can be achieved. In PCA, as the Eigen faces (Terzopoulos and Waters 2001) method derives the most expressive and inspired features, they provide a good discrimination.

This PCA based holistic approaches to face recognition have the advantage of distinctively capturing the most prominent features within the face image. Eigenfaces (Turk and Pentland 1991) algorithm has some shortcomings due to the use of image pixel gray values. In PCA, the size and location of each face image must remain the same. Different illumination, head pose and facial expressions lead to reduced recognition rate. In statistical approach PCA, it is difficult to express structural information unless an appropriate choice of features is possible. Training is computationally intensive and it is hard to decide suitable thresholds. The methods deal with unknown faces and non-faces are not good enough to differentiate them from known faces.

The PCA approach typically requires the full frontal face to be presented each time; otherwise the image results in poor performance. Although the face recognition results were acceptable, the system using only eigenfaces (Vijaya Lata et al 2009) might not be applicable as a real system. It needs to be more robust and should have other discriminant features. PCA is translation variant, scale variant, background variant and lighting variant.
Noise is reduced because the maximum variation basis is chosen and hence features like background with small variation are automatically ignored.

FLDA with its application in Face recognition is presented in next section.

3.2 FISHER LINEAR DISCRIMINANT ANALYSIS

3.2.1 Introduction

LDA is a statistical approach for classifying samples of unknown faces based on training samples with known faces (Juwei Lu et al 2003). The PCA projections are optimal for reconstruction from a low dimensional basis, but they are not optimal from a discrimination standpoint. To overcome this obvious shortcoming, Fisher Linear Discriminant (FLD) technique (Moshe Butman and Jacob Goldberger 2008) is used for the purpose of achieving high separability between the different patterns.

PCA constructs the face space using the whole face training data as a whole, and not using the face class information (Martinez and Kak 2001). On the other hand, LDA uses class specific information which best discriminates among classes. LDA produces an optimal linear discriminant function which maps the input into the classification space in which the class identification of this sample is decided based on some metric such as Euclidean distance. Thus the objective of FLD is to find the optimal projection, so that the ratio of determinants of between-class and the within-class scatter matrices of the projected samples reaches its maximum.

3.2.2 Theory of FLDA

The main aim of the FLDA is finding a base of vectors providing the best discrimination among the classes, trying to maximize the between-
class differences, minimizing the within-class ones. FLDA takes into account the different variables of a face and works out and decides which group the face most likely belongs to. The FLDA develops a set of feature vectors in which variations of different faces are emphasized, while different instances of faces due to illumination conditions, facial expression and orientations are de-emphasized. The working of FLDA is as per the block diagram shown in Figure 3.7.

![FLDA Block Diagram](image)

**Figure 3.7 FLDA approach for face recognition**

Fisher’s Linear Discriminant is a projection into a subspace that maximizes the between class scatter while minimizing within class scatter of the projected data. In order to improve the generalization capability of this FLDA, it is decomposed into a simultaneous diagonalization of the two
within-class covariance matrices. This simultaneous diagonalization is stepwise equivalent to two operations of forming the between-class covariance matrix and applying PCA on the between-class covariance matrix using the transformed data (El-Bakry 2007). The robustness of the FLDA procedure depends on whether the within-class scatter captures reliable variations for a specific class or not. FLDA gives a projection matrix $W$ that reshapes the scatter of a data set, so as to maximize inter class separability, which is defined as the ratio of the “between class scatter matrix” to the “within class scatter matrix”. This projection defines features that are optimally discriminating.

The FLDA algorithm is applied to all face images as below:

Step 1: Acquire the training set and test set of face images and form its feature vector representation.

Step 2: Calculate the within class and between-class covariance matrices $S_w$ and $S_b$ matrices as per equations (3.10) and (3.11).

Step 3: Form the transformation matrix with higher separability as per equation (3.12).

Step 4: Classify the given face image, according to the Euclidean distance and threshold values.

A major contribution of the method is that discriminant vectors obtained by FLDA are used to determine salient local features (Belhumeur et al 1997), the positions of which are specified by discriminant pixels. In conventional FLDA of face patterns, the criterion of measuring the discriminatory power of the projection vectors is to maximize the between-class scatter and in the meantime to minimize the within-class scatter of the
projected samples. The shortcoming of this approach involves solving the eigenvalue problem for a very large matrix. In the case where there are more than two classes, the analysis used in the derivation of the Fisher discriminant can be extended to find a subspace which appears to contain all of the class variability.

3.2.3 FLDA Algorithm Explanation and Experimentation Results

The Eigen value spectrum of the within-class covariance matrix in the reduced PCA space can be derived and different spectra are obtained corresponding to different number of principal components utilized. Now, one has to simultaneously optimize the behavior of Eigen values in the reduced PCA space with the energy criteria for the original image space. After finding the covariance matrix as defined by equation (3.3), the between-class and within-class covariance matrices are determined as per the following procedure. Let $k_1$, $k_2$, $k_3$… $k_n$ and $N_1$, $N_2$, $N_3$… $N_L$ denotes the classes and the number of images within each class respectively. The transformation Matrix is needed for the classification of the given input image.

The Inter and Intra covariance matrices are termed as the within class and between-class covariance matrices $S_w$ and $S_b$. The between class covariance matrix ($S_b$) and within class covariance matrix ($S_w$) are determined as per the equations (3.8) and (3.9). The Eigen vector and the Eigen values of $S_w$ are $\Theta$, $\Gamma$. The multiplication of transpose of this transformation matrix with the reduced shape vector of the given image classifies in to the corresponding class. The EFC method employs the transformation matrix “T” on the given input image feature vector “X”, to implement the recognition. When an unknown image is given to the FLDA classifier, the shape of that image is calculated and thereby the reduced vector “Y” is derived. To calculate matrix $W$, consider a case of K class $\{c_1, c_2, \ldots, c_k\}$. Calculate $\mu_i$, the mean of all samples by
\[ \mu_i = \frac{1}{N} \sum_{i=1}^{N} x_i \]  

(3.8)

\[ \mu_{sk} = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} x_{ki} \]  

(3.9)

where \( N \) is the total number of samples in the classes. \( x_i \) is \( i \)th sample of a data and \( x_k \) is \( k \)th sample of a class. The between class scatter matrix is

\[ S_b = \sum_{k=1}^{K} N_k (\mu_{sk} - \mu_s)(\mu_{sk} - \mu_s)^T \]  

(3.10)

where \( \mu_{sk} \) the mean of class \( K \). the within class scatter matrix is defined by:

\[ S_w = \sum_{k=1}^{K} \sum_{i \in C_k} (x_i - \mu_{sk})(x_i - \mu_{sk})^T \]  

(3.11)

In particular, the stepwise FLD procedure derives the Eigen values and Eigen vectors as the result of the simultaneous diagonalization of \( S_w \) and \( S_b \). The transformation matrix \( W \) with the highest seperablity is the one which maximizes the ratio \( J \) as,

\[ J = \frac{\det( W^T S_b W )}{\det( W^T S_w W )} \]  

(3.12)

The transformation matrix \( W \) is a matrix containing generalized Eigen vectors of \( S_b S_w \) with largest Eigen value. The transformation matrix \( W \) with the highest seperablity is the one which maximizes. This means that when \( \vec{\mu}^T \) is an eigenvector of \( \Sigma^{-1} S_b \), the separation will be equal to the corresponding eigenvalue. Since \( S_b \) is of most rank \( C-1 \), then these non-zero eigenvectors identify a vector subspace containing the variability between features. These vectors are primarily used in feature reduction, as in PCA.
The smaller Eigen values will tend to be very sensitive to the exact choice of training data, and it is often necessary to use regularization. The purpose of regularization is to reduce the high variance related to the eigenvalue estimates of the within-class scatter matrix at the expense of potentially increased bias.

The within class scatter matrix represents how face images are distributed closely within classes and between class scatter matrix describes how classes are separated from each other. When face images are projected into the discriminant vectors $W$, face images should be distributed closely within classes and should be separated between classes, as much as possible. Dimensionality reduction is done by discarding eigenvectors with smaller eigenvalue. If $X$ and $Y$ are denoted as the input vector and feature vector as per FLDA respectively, then $Y$ is as:

$$Y = W_m^T X$$

(3.13)

where $W_m$ is a matrix containing generalized eigen vectors of $S_b S_w^{-1}$ with largest eigen value. LDA subspace is spanned by a set of vectors $(W_1, W_2, \ldots W_M)$ and Euclidean distance is given by

$$E_k = \| (\Omega - \Omega_k) \|^2$$

(3.14)

where $\Omega_k$ is a weight vector describing the $k^{th}$ face. A face is classified as belonging to person $k$ when the minimum below some chosen threshold, otherwise, the face is classified as unknown.

Various experiments have been conducted using standard databases like ORL, Yale, FERET and some real time images are presented in Figures 3.8, 3.9, 3.10 and 3.11 respectively.
Figure 3.8 Results of FLDA for ORL database
Figure 3.9 Results of FLDA for YALE database
Figure 3.10  Results of FLDA for FERET database
Figure 3.11  Results of FLDA for real time database
3.2.4 Advantages and disadvantages of FLDA

Fisher Linear Discriminant Analysis (Heseltine et al 2007) method provides better ability to recognize a face and provides better discrimination between faces. FLDA works well for different illumination and different facial expressions. This statistically motivated method maximizes the ratio of the determinant of between-class scatter matrix and within-class scatter matrix and in this sense attempts to involve information about classes of the patterns under consideration providing lower error rate.

The solution for recognition of facial image samples involves segmentation of faces from the cluttered scenes or background, extraction of feature vector from the face region, identification and matching. The images used in this work, can have variations in head orientation, scaling and lighting. This approach requires a high degree of correlation between the pixel intensities of the sample and the test images. FLDA (Yu and Yang 2001) has been successfully applied to computer vision and pattern recognition applications in the past few years.

The tests conducted on various subjects in different environments show that this approach has limitations over the variations in light, size and in the head orientation. Nevertheless, this method showed very good classifications of faces. A good recognition system should have the ability to adapt over time. Reasoning about images in face space provides a means to learn and subsequently recognize new faces in an unsupervised manner.

It is well-known that the applicability of FLDA to high-dimensional pattern classification tasks such as Face Recognition (FR) often suffers from the so-called Small Sample Size (SSS) problem arising from the small number of available training samples compared to the dimensionality of the sample space.
A difficulty in using the FLDA method for face recognition is the high-dimensional nature of the image vector. Actually, the variations between the images of the same face due to illumination and viewing direction are almost larger than image variations due to change in face identity.

### 3.2.5 Comparison of PCA with FLDA

When the training set is small, PCA can outperform FLDA and is not ideal for classification purposes as it retains unwanted variations occurring due to diversified lighting and facial expressions. Fisher faces project the faces in a sub-space which maximizes the ratio of inter-class and intra-class variability. When the number of samples is large, FLDA outperforms PCA and execution time is less in FLDA. The recognition rate of FLDA is higher than that of PCA, because FLDA (Zhao et al 1998) utilizes class specific information and the most discriminant features instead of the most expressive features.

All of the algorithms perform perfectly when lighting is nearly front view. In the Eigenfaces (Zhang et al 1997) method, the Linear Subspace method has error rates that are competitive with the Fisher face method. The Fisher face method has error rates lower than the Eigenfaces method and requires less computation time.

It seems that the Fisher face method chooses the set of projections which performs well over a range of lighting variation, facial expression variations and presence of glasses. Because of variation in facial expression, the images no longer lie in a linear subspace. Since the Fisher face method tends to discount those portions of the image that are not significant for recognizing an individual, the resulting projections W tend to mask the regions of the face that are highly variable.
On the other hand, the nose, cheeks and brow are stable over the within-class variation and are more significant for recognition. It is generally believed that, when it comes to solving problems of pattern classification, FLDA-based algorithms outperform PCA-based ones, since the former optimizes the low-dimensional representation of the objects with focus on the most discriminant feature extraction, while the latter achieves simply object reconstruction.

A major contribution of the method is that discriminant vectors obtained by FLDA are used to determine salient local features, the positions of which are specified by discriminant pixels. This approach involves solving the eigenvalue problem for a very large matrix. The purpose of regularization is to reduce the high variance related to the eigenvalue estimates of the within class scatter matrix at the expense of potentially increased bias.

Some of the performance metrics (Wangmeng Zuo, 2006) like Recognition Rate, False acceptance Ratio (FAR), False Rejection ratio (FRR) and accuracy are calculated using the following formula.

Recognition rate = \[
\frac{\text{Number of persons correctly recognized}}{\text{Total number of Trials}} \]
\tag{3.15}

\[
\text{FAR} = \frac{\text{Number of Unauthorized persons recognized as authorized person}}{\text{Total number of unauthorized Trials}}
= \frac{\text{UP to AP}}{\text{UT}} \tag{3.16}
\]

\[
\text{FRR} = \frac{\text{Number of Authorized persons recognized as unauthorized person}}{\text{Total number of authorized Trials}}
= \frac{\text{AP to UP}}{\text{AT}} \tag{3.17}
\]

\[
\text{Accuracy} = [1-(\text{FAR}+\text{FRR})/2] * 100 \%
\tag{3.18}
\]
Comparison of Recognition rate and Execution time of PCA and FLDA are shown in Figures 3.12 and 3.13 respectively. Comparison of FAR and FRR for YALE, ORL and FERET databases are presented in Figures 3.14, 3.15 and 3.16 respectively. Figures 3.17 shows the comparison of accuracy for PCA and FLDA for 100 ORL database images.

Table 3.1 Comparison of recognition rate of PCA and FLDA for ORL database

<table>
<thead>
<tr>
<th>No. of images</th>
<th>Recognition rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PCA</td>
</tr>
<tr>
<td>50</td>
<td>89</td>
</tr>
<tr>
<td>100</td>
<td>86</td>
</tr>
<tr>
<td>200</td>
<td>83</td>
</tr>
<tr>
<td>300</td>
<td>80</td>
</tr>
<tr>
<td>400</td>
<td>75</td>
</tr>
</tbody>
</table>

Figure 3.12 Comparison of recognition rate of PCA and FLDA ORL database
Table 3.2 Comparison of execution time of PCA and FLDA

<table>
<thead>
<tr>
<th>No. of images</th>
<th>Execution time (sec)</th>
<th>PCA</th>
<th>FLDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>42.46</td>
<td>37.46</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>48.1</td>
<td>43.02</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>55.32</td>
<td>50.1</td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>66.46</td>
<td>61.05</td>
<td></td>
</tr>
<tr>
<td>400</td>
<td>78.04</td>
<td>72.54</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.13 Comparison of execution time of PCA and FLDA
Table 3.3 Comparison of FAR and FRR for Real time images

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Authorized Trial (AT) Trained images</th>
<th>Unauthorized Trials (UAT) Test images</th>
<th>No. of (UAP–AP)</th>
<th>No. of (AP–UAP)</th>
<th>False Acceptance Rate (FAR) (%)</th>
<th>False Rejection Rate (FRR) (%)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
<td>2</td>
<td>2</td>
<td>20</td>
<td>20</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>20</td>
<td>4</td>
<td>5</td>
<td>20</td>
<td>25</td>
<td>77.5</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30</td>
<td>5</td>
<td>9</td>
<td>16.66</td>
<td>30</td>
<td>76.67</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>40</td>
<td>7</td>
<td>13</td>
<td>17.5</td>
<td>32.5</td>
<td>75</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>50</td>
<td>10</td>
<td>16</td>
<td>20</td>
<td>32</td>
<td>74</td>
</tr>
</tbody>
</table>

Figure 3.14 Comparison of FAR and FRR for Real time images Database
Table 3.4 Comparison of FAR and FRR of ORL database

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Number of test images</th>
<th>No. of Authorized Trials</th>
<th>No. of Unauthorized Trials</th>
<th>No. of (UAP –AP)</th>
<th>No. of (AP-UAP)</th>
<th>False Acceptance Rate (%)</th>
<th>False Rejection Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>10</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>15</td>
<td>2</td>
<td>3</td>
<td>13.33</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>20</td>
<td>4</td>
<td>5</td>
<td>20</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td>25</td>
<td>5</td>
<td>6</td>
<td>20</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>30</td>
<td>7</td>
<td>7</td>
<td>23.33</td>
<td>23.33</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.15 Comparison of FAR and FRR for ORL database
Table 3.5 Comparison of FAR and FRR-FERET database

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Number of test images</th>
<th>No. of UAP to AP</th>
<th>No. of AP to UAP</th>
<th>False Acceptance Rate (FAR) (%)</th>
<th>False Rejection Rate (FRR) (%)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Authorized Trial (AT) Trained images</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>Unauthorized Trials (UAT) Test images</td>
<td>20</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>Trained images</td>
<td>30</td>
<td>30</td>
<td>4</td>
<td>7</td>
<td>13.33</td>
</tr>
<tr>
<td>4</td>
<td>Test images</td>
<td>40</td>
<td>40</td>
<td>5</td>
<td>10</td>
<td>12.5</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>50</td>
<td>50</td>
<td>7</td>
<td>14</td>
<td>14</td>
</tr>
</tbody>
</table>

Figure 3.16 Comparison of FAR and FRR for FERET database

FAR & FRR - FERET database -PCA

No of images
Table 3.6 Comparison of Accuracy for PCA

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Number of images</th>
<th>Authorized trials</th>
<th>Unauthorized trials</th>
<th>ORL</th>
<th>Yale</th>
<th>FERET</th>
<th>Real time images</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
<td></td>
<td>90</td>
<td>85</td>
<td>85</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>20</td>
<td></td>
<td>87.5</td>
<td>85</td>
<td>82.5</td>
<td>77.5</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30</td>
<td></td>
<td>85</td>
<td>83.33</td>
<td>80.16</td>
<td>76.67</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>40</td>
<td></td>
<td>83.75</td>
<td>82.5</td>
<td>81.25</td>
<td>75</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>50</td>
<td></td>
<td>82</td>
<td>81</td>
<td>79</td>
<td>74</td>
</tr>
</tbody>
</table>

Figure 3.17 Comparison of accuracy for PCA
Face recognition system also employs a variety of techniques for selecting subspaces. As a result, it is difficult to assign credit to a particular component of a face recognition system, even when the details are not proprietary. The purpose of this work is to compare the performance of two subspace projection techniques on face recognition tasks in the context of a simple baseline system.

More importantly for this work, the matching techniques are enhanced. This is by pre-processing the images, selecting and in some cases generating training data, generating spatially localized features and optimizing classifiers for compressed subspaces. The eigen value spectrum of the covariance matrix is a good indicator for meeting the energy criterion and one needs further to derive the eigen value spectrum of the within-class covariance matrix in the reduced PCA space to facilitate a choice of the range of principal components so that magnitude criterion is met. The drawback of the FLD is that it requires large sample sizes for good generalization.

3.3 SUMMARY

PCA has better accuracy with frontal faces. Also, scale and orientation of an image will affect the accuracy greatly. PCA takes more processing time and is efficient in storage. The accuracy of Eigenfaces is satisfactory. FLDA is more complex than PCA in calculating the projection of face space. FLDA has better accuracy with different face expression. Calculation of the ratio of between-class scatter to within-class scatter requires some more processing time for recognition, in FLDA for large number of database. But the recognition is better in FLDA in all other aspects like different illumination, pose and expression etc. In the next chapter, Gabor Filter and Neural Network based approaches for face recognition are discussed.