Face Recognition Using Random Forest Classifiers Based on PCA, LDA and LBP Features

Armin Mehri

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Eastern Mediterranean University
January 2017
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Mustafa Tümer
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Computer Engineering.

Prof. Dr. Işık Aybay
Chair, Department of Computer Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Computer Engineering.

Asst. Prof. Dr. Adnan Acan
Supervisor

Examining Committee

1. Assoc. Prof. Dr. Mehmet Bodur
2. Assoc. Prof. Dr. Önsen Toygar
3. Asst. Prof. Dr. Adnan Acan

ABSTRACT

Face is the main part of human beings to distinguish from one another. Face recognition system mainly takes an image as an input and compares this image with a number of images stored in the database to identify whether the input image is in the database or not. Also, face recognition is the process of identification and verification of individuals by their facial images.

In this thesis, well-known databases such as FERET and JAFFE databases are used for experimental evaluations. Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Local Binary Patterns (LBP) are used for extracting facial features of individuals from the region of interests. Decision Tree (DT) and Random Forest (RF) are used as classify the faces based on extracted features. The Manhattan Distance measure is used to compare the difference between test and training images for face recognition. Based on the experimental evaluations, the achieved recognition rates are very close to those published articles in the literature.

Keywords: Local Binary Patterns (LBP), Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Random Forest (RF), Decision Tree (DT), feature extraction, classification.

ÖZ

Yüz, insanoğlunun birbirinden ayırt etmenin ana parçasıdır. Yüz tanıma sistemi esasen bir görüntüyü bir girdi olarak alır ve bu görüntüyü, girilen görüntünün veritabanında olup olmadığını belirlemek için veritabanında saklanan bir dizi resimle karşılaştırır. Ayrıca, yüz tanıma, yüz imgelerine göre bireylerin tanımlanması ve doğrulanması sürecidir.

Bu tezde, deneysel değerlendirmeler için FERET ve JAFFE veritabanları gibi tanınmış veri tabanları kullanılır. İlgi alanından kişilerin yüz özelliklerini çıkarmak için Temel Bileşen Analizi (PCA), Doğrusal Ayırtaç Analizi (LDA) ve Yerel İkili Örüntü (LBP) kullanılır. Ayıklanan özelliklere dayalı olarak yüzleri sınıflandırıcı olarak Karar Ağacı (DT) ve Rastgele Orman (RF) kullanılmıştır. Manhattan Distance ölçümü, yüz tanıma için test ve eğitim resimleri arasındaki farkı karşılaştırmak için kullanılır. Deneysel değerlendirmelere dayanarak, elde edilen tanıma oranları literatürde yayınlanan makalelere çok yakındır.

Anahtar Kelimeler: Yerel İkili Örüntü (LBP), Temel Bileşen Analizi (PCA), Doğrusal Ayırtaç Analizi (LDA), Rastgele Orman (RF), Karar Ağacı (DT), sınıflandırma, öznitelik çıkarma.

To My Beloved Family

ACKNOWLEDGMENT

Firstly, I would like to express my sincere gratitude to my advisor Asst. Prof. Dr. Adnan Acan for the continuous support of my M.Sc. study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my M.Sc. study. Secondly, I want to thank the faculty members of Computer Engineering department and the staff for all their hard work and dedication, providing me the means to complete my degree and prepare for a career as a computer scientist. And finally, my deep and sincere gratitude to my family for their continuous and unparalleled love, help and support. I am grateful to my brother for always being there for me as a friend.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Biometrics is the measurement and statistical analysis of people's physical and behavioral characteristics. The technology is mainly used for identification and access control, or for identifying individuals that are under surveillance. Face detection, face analysis and face recognition as just some of the plenty of struggling problems, have been investigated profoundly during the recent few decades. Obviously, large population of the authentication process consume salient amount of computing time. The identification of mankind face with a totally automate system of identification can be regarded as one of possible solutions which can be helpful to decrease the case space of authentication to almost half of the existing data and to save substantial amount of time [1][2].

The face, as the primary focus in social intercourses, plays an essential role in transmitting human identification. In fact, face recognition can be considered as the most efficient technique of human surveillance [4] [5]. Due to this fact, Applications such as smart human-computer interface, biometrics, security industry, and surveillance would benefit extensively from the knowledge of the attribute of the human subjects under scrutiny [5].

The verification system or an identification system, as face recognition system, can be used to compare the captured image with the samples which have been saved in system database and it can also check all images to discover specific images in the facial database. An identification system recognizes each individual via checking all face images in facial database to discover a match. Furthermore, the system has two fundamental solutions: the person is in system database who can be identified or no match of the person can be made.

Obviously, human ability to recognize each other's faces is vital, because it can make human able to recognize all faces, memorized during the span of life and even make them capable of the recognition of familiar faces after passing time during several years just by one glance. There is no wonder that this ability of preserving deep changes in the appearance due to age, expressions, viewing conditions or hairstyle changes or glasses is extremely potential [6].

Face recognition is including of two basic factors: appearance basis and structural basis. The primary one; the appearance is one basis of face appearance which can be read as major input and it can be regarded as major factor in decision making system while the second one is on the basis of structure which features geometric objects like nose, month, eyes and the place of facial features from feature vectors. Noteworthy, these feature vectors are useful to identify the subject [6] [7].

In this thesis, three feature extraction techniques are used to extract facial features, including: Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Local Binary Patterns (LBP). Besides, there are two well-known classification methods: Decision Tree (DT) and Random Forests (RF) which can be applied for face image classification.

The rest of this study include literature review; presented in chapter 2 and the preprocessing steps; discussed in chapter 3. Chapter 4 explores Principal Component Analysis, Linear Discriminant Analysis, Local Binary Patterns, Decision Tree and Random Forest algorithms while chapter 5 presents the databases which have been used in this study. Chapter 6 demonstrates the experimental results of face recognition and the classification of face images; and finally, chapter 7 is the conclusion of the whole thesis.

Chapter 2

LITERATURE REVIEW

To identify subjects and developments in the computing capability over the recent few years, human-kind initiate to benefit from face images frequently. It is worth mentioning that the preliminary face identification algorithms apply geometric patterns of samples which need to find geometric features manually such as eyes, ears, and eyebrow. In contrary to the previous years when significant improvements guided face identification method into the searchlight, most of recognition procedures use the knowledge of sophisticated mathematical and pattern matching processes; nowadays.

In the 1960s, automated face recognition concept was developed. Initially, the face recognition system was not able to work as absolute automated system. So, it sounds essential to discover the location of features like: ears, eyes, mouth and etc. on the images before computing the distance to original data.

Strikingly, two researchers: namely Goldstein, Harmon, and Lesk [9] tried to use special intellectual markers to create automated identification; however, the method needed the feature locations; manually calculated. In 1988, two researchers who used a standard linear algebra technique to discover face identification issues [10], have figured out that exactly less than hundred value was required to be able to code and to normalize face images.

While researchers became too enthusiastic to reveal more data about recognition systems, various algorithms were developed. Multitude variety of algorithms have been scrutinized profoundly in face recognition during the history such as Local Binary Patterns (LBP), Linear Discriminant Analysis (LDA) and Principal Components Analysis (PCA) as described in the following subsections.

2.1 Principal Components Analysis (PCA)

In 1988 [12], Kirby and Sirovich created PCA technique which can be; generally, referred to the usage of eigenfaces.

It is worth to consider the fact that the input and gallery images must be the same size and they must be initially, be normalized to line up the eyes and mouth of the subjects within the images in PCA technique. Considerably, this kind of reduction in dimensions removes non-useful information and precisely decomposes the face structure into orthogonal (uncorrelated) components; known as Eigen faces. Each face image may be represented as a weighted sum (feature vector) of the Eigen faces, which can be stored in 1D array. Kirby and Sirovich [12] applied principal component analysis, a standard linear algebra technique, to the face recognition problem. This was considered somewhat of a milestone as it showed that less than one hundred values were required to accurately code a suitably aligned and normalized image.

An input image can be compared with image gallery via measuring the distance between their respective feature vectors. Typically, the PCA technique needs the complete frontal face to be presented each time; otherwise, the results of image are poor performance. The magnificent pron of this technique is its ability to decrease the required data to identify the individual.

In 2002, the recognition rate using PCA method on FERET database; achieved by Baek, was 80%. Respectively, in 2005, Delac et al. [14] have focused their study on PCA, ICA and LDA. Interestingly, they achieved 82.26%, 81.51% and 82.76% recognition rate, respectively.

2.2 Linear Discriminant Analysis (LDA)

LDA is known as Fisher's Linear Discriminant algorithm. In order to find a linear combination of features, the LDA technique is used which separates or characterizes two or more classes of objects in pattern recognition and machine learning. PCA and LDA are approximately, similar to each other [16]. The LDA technique tries to model the difference between the classes of data, whereas Principal Component Analysis is unsupervised learning which ignored classes label.

LDA [17] is a statistical technique which can be used for the sample classification of non-discovered classes; on the basis of training samples with discovered classes. The main object of this techniques is to maximize between-class (i.e., across users) variance and minimize within-class (i.e., within user) variance. LDA searches for those vectors in the underlying space that discriminate among classes in the ideal way (rather than those that best describe data as in Principal Component Analysis). LDA, is a linear composition of an independent features which yields the largest mean diversities between the desired classes. The main opinion of LDA is to find a linear transformation such that attribute clusters are most separable after the transformation which can be computed via scatter matrix analysis. The aim of LDA is to maximize the S_b measure as minimizing the S_w measure.

In 2005, Delac et al. [14] worked on Linear Discriminant Analysis on FERET database and he estimated the accuracy of 82.76%. Furthermore, in 2013, Shih-Ming Huang and Jar-Ferr Yang [15] applied Linear Discriminant Analysis on FERET database and they achieved 84.8% accuracy on FERET database with LDA (Fisher face).

2.3 Local Binary Patterns (LBP)

The original Local Binary Patterns (LBP); presented by Ojala et al. [18], was on the basis of the hypothesis that texture has locally two complementary views, a pattern and its strength [19]. Recently, LBP has converted to energetic popular topic in computer vision and image processing.

LBP; known as a non-parametric technique optimizes local structure of pictures impressively by means of comparing the whole pixels with neighboring ones. Considering monotonic illumination changes and its computational simplicity, the most essential properties of LBP is its tolerance. LBP which has been approved as the simple yet powerful approach to explain the local structure, was presented to analyze the texture basically. Remarkably, face image analysis, image and video detection, surroundings modeling, visual audit, movement assessment, biomedical and aerial image analysis, and remote sensing are multitude applications of LBP technique [20] [21].

In the basic LBP, each pixel is contrasted with its eight neighbors in a 3×3 neighborhood by subtracting the center pixel value; the resulting strictly minus values are encoded with 0, and the others with 1. For each existed pixel, a binary number gets by concatenating all these binary values in a clockwise way, which starts from one of its top-left neighbor. The corresponding decimal value of the produced binary number is then used for labeling the given pixel. The derived binary numbers are repetitive to be the LBPs or LBP codes [22].

In 2011, Meena and Suruliandi [21] applied Local Binary Patterns on JAFFE database. The maximum recognition rate is 81% and also in 2006 Ahonen et.al [25] applied LBP on FERET database and they achieved 93% recognition rate.

2.4 Decision Tree (DT)

The Decision Tree, introduced by Bittencourt and Clarke (2003) [23], is a binary Decision Tree for clustering that can be seen as a non-parametric method in pattern recognition. Hierarchical representation can be produced from a decision tree of the feature space in which patterns x_i are assigned to classes w_j (j=1,2,...,k) pursuant to the results acquired by following decisions made at a sequence of nodes at which branches of the tree diverge. The basic method of a decision tree specified by Breiman et al. (1984) is used in this study. Classification and Regression Trees (CART) show that Decision Tree might be used not only as a replacement method for regression analysis which the value of dependent variable is computed but also may be used to classify entities into discrete number of groups.

Decision trees contain repeated divisions of feature into two sub-spaces. Final nodes are associated with class w_j. A suitable Decision Tree is the one which has less number of branches and also contains less number of middle nodes from which these branches split. A Decision Tree has high predictive in which entities are correctly classified on the terminal nodes.

Advantages of Decision Trees are as follow [24]:

- Decision tree is simple and tree model can be perceived after short explanation.
- It is not vital to include much preprocessing against other techniques.
- The decision tree has the ability to overcome both categorical data and numerical data but other methods are normally specialized in analyzing datasets that have only one type of variable.

In 2013, Mohsen et al. [26] used decision tree to classify and also to extract features which they applied Linear Discriminant Analysis on JAFFE database and 72.6% recognition rate is achieved.

2.5 Random Forest (RF)

Recently, ensemble-learning algorithms are receiving more and more interest in the field of classification methods. Ensemble methods are learning algorithms that construct a set of many individual classifiers called weak learners to form a unique classification system. Random Forest belongs to this ensemble method category; can correspond on combination of decision tree-type classifier, in the way that per tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. RF can be seen as combination of two types of ensemble-method, boosting and bagging. In fact, it is built by randomly sampling a feature subset for each decision tree as boosting, and by randomly sampling a training data subset for each decision tree as in Bagging.

Ensemble learning [27] algorithms which produce sets of classifiers. To achieve a good improvement in classification accuracy, it is necessary to grow a group of trees and make them vote for well-known class has created. Mostly, in the classifier to control the growth of each tree random vectors are made.

The advantages of Random Forest are as [28]:

1. The ability to obtain high accurate classifier in various datasets.
2. RF handles an enormous amount of input variables.
3. RF forecasts the significance of variables in achieving classification.
4. RF produces an inner unbiased calculation of the generalization error as the forest structure processes.
5. Fast learner.

In 2007, Kouzani et al. [27] has studied about Random Forest as the main classifier on FERET database with multitude variety number of trees like 5, 250, 255, 295 and 2480. Respectively, the classification rate of 76.3%, 97.5%, 100%, 96.3% and 93.8% are obtained.

Chapter 3

PREPROCESSING

Individual images recorded by a digital camera are usually inappropriate for recognition due to diversity in background, number of intensity levels, contrast of images, state of head, size of the pictures and etc. Some of the programs could not automatically work out all of these issues. Therefore, some of the deficiencies of these images should be solved before defining them to the program as input and some of them should be solved during process of recognition. Figure 1 illustrates the procedures of preprocessing that are used to make images ready for face recognition algorithms.

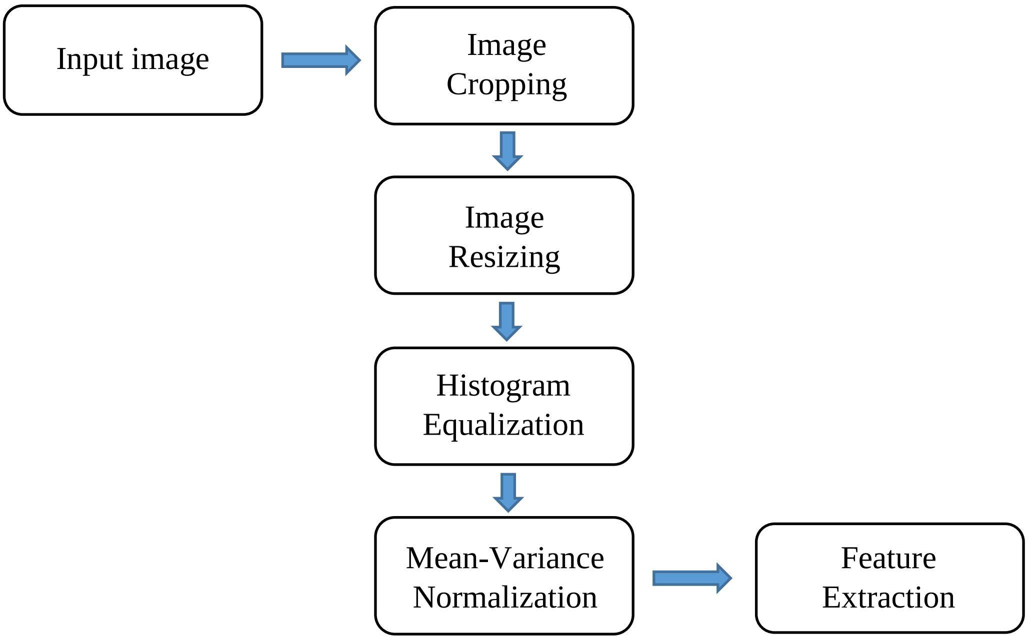

Figure 1: Steps of preprocessing images

3.1 Image Cropping

First step of preprocessing is cropping image in order to increase the speed of detecting face and decrease memory consumption, it is necessary to remove useless data from the image. The second reason of cropping image is the background of the images because some part of the images are sources of failure in recognition such as neck and hairs. Figure 2 and Figure 3 show the difference between original images and cropped images.

Figure 2: Original pictures from FERET and JAFFE databases

Figure 3: Cropped images according to original images

3.2 Image Resizing and Interpolation of Images

Next step of preprocessing the images before defining image as an input image is resizing image because, each image may have different size therefore it is necessary to unify the size of all images. In this study, in order to unify the size of all images, bicubic interpolation technique had been used. Different varieties of interpolation techniques such as nearest neighbor, bicubic and bilinear exist. Bicubic interpolation preserves details of images better than the other interpolation methods.

3.3 Histogram Equalization

In computer vision, histogram refers to nearby frequencies that takes place in gray levels of an image. An image can have several amount of intensity levels with different density and it degrades the performance of face recognition descriptor.

In the same database the intensity levels of two images might be different. In order to distribute their levels of intensities, histogram equalization (HE) technique can be used. HE increases the range of intensity and spreads the intensity distributions which are better than having flattened peaks and valleys for an image in terms of a histogram [11]. This operation increases contrast of the low contrast areas without affecting the overall contrast of the image.

Histogram equalization [12] method improves the accuracy of face recognition by equalizing the intensity levels of unlike images. Images with different intensity values can have equalized histogram values by using Histogram equalization. Figure 4 and figure 5 illustrate after and before applying histogram equalization on images.

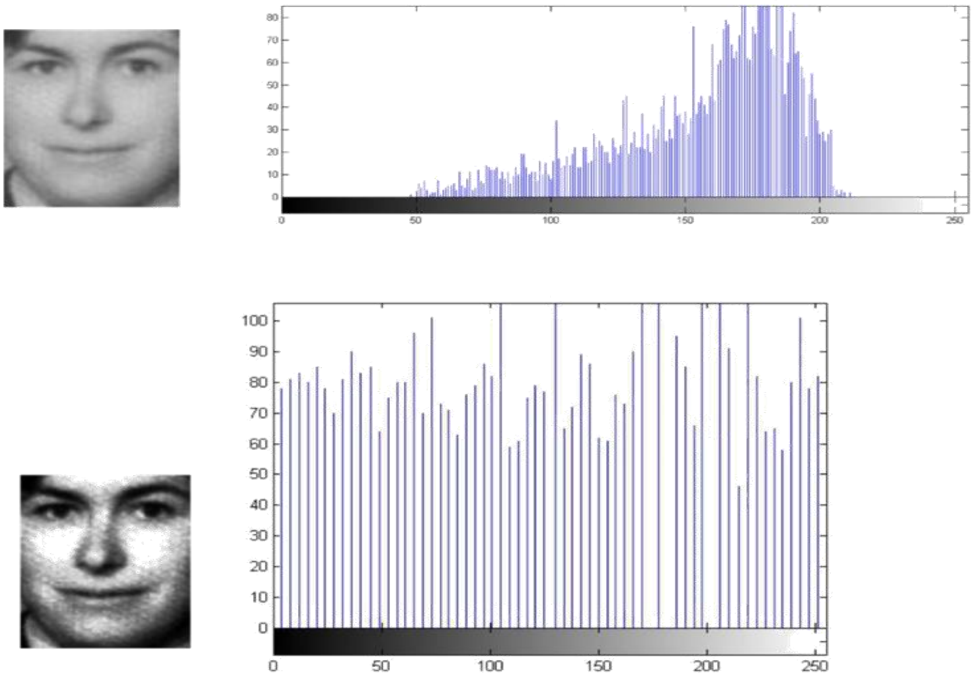

Figure 4: Histograms of bright image before and after histogram equalization

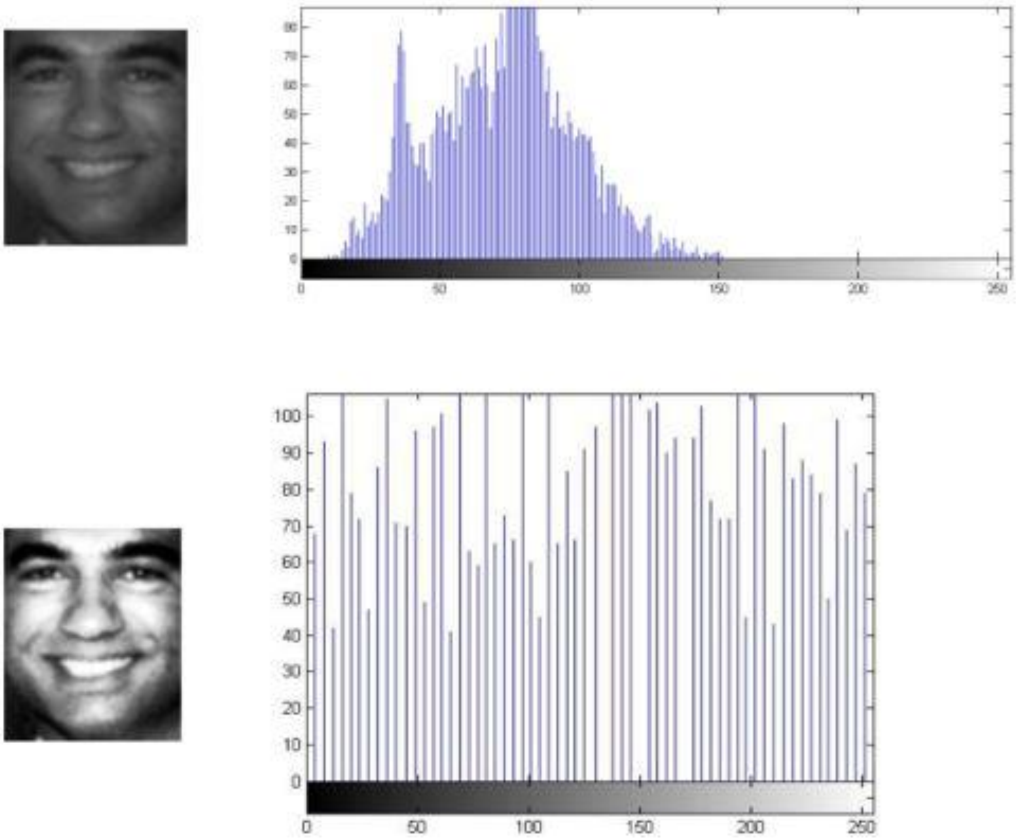

Figure 5: Histograms of dark image before and after histogram equalization

As Figure 4 and 5 indicates, dispensation of intensity level of both images is almost close to each other. Figure 6 illustrates the diversity of images before and after using HE techniques. Contrast of images after accomplishment of HE techniques is near to one another. For instance, picture A form the beginning was dark and after using HE method changed into lighter or picture B originally was light and after HE technique it turns into darker.

Figure 6: Face image before and after HE

Histogram equalization is assessed by the following equation:

$$S_k = (L-1)\sum_{j=0}^{K} P_{r_{(rj)}} \qquad (3.3.1)$$

Where L is the total amount of feasible intensity levels, k = 0, 1, 2, …, L – 1 and P_r is the calculation of the probability of incidence of intensity level in a picture. Histogram equalization is a technique of rounded $S_k$and n_k where n_k is value of pixels that have intensity value r_j, where $r_j$is the intensity level of input image.

3.4 Mean-Variance Normalization

Normally, for increasing the robustness of recognition features Mean-Variance Normalization is used to largely reduce the actual mismatch between training and testing condition. In this study, for significantly improve the recognition rate both mean-variance normalization and histogram equalization have been used [12].

MVN is obtained as follows:

$$R = X - M_x \text{ , } MVN = R / \text{std}(R)$$

where X is a matrix consisting of the intensity values of a grayscale image, $M_x$is mean value of X and std is standard deviation of R.

Chapter 4

METHODOLOGY

In this chapter, Principal Component Analysis, Linear Discriminant Analysis and Local Binary Patterns as features extraction and Decision Tree and Random Forest as classifiers are described and we also present how to extract features of a given image by using these feature extractors and the main steps of Decision Tree and Random Forests.

4.1 Principal Component Analysis

Principal Component Analysis (PCA) is one of the statistical feature extraction method and also is an unsupervised learning method that ignores class labels. In general, PCA is a dimensionality reduction method which is widely used in pattern recognition and image compression [9].

Principal Component Analysis [9] which is used for feature extraction and feature reduction is an efficient technique in computer vision. It loosely based on simple linear algebra and it is a lossless technique. Before introducing the PCA steps, it is necessary and important to describe eigenvectors in mathematical forms as one of the significant part in PCA algorithm. The mathematical definition of an eigenvector can be defined as AC=λC, where A is a square matrix, C is the eigenvector of A and it is not a zero vector, and real or complex must be present. Eigenvectors can be found only for square matrices but not all square matrices have them, and if they have, the square matrix has the dimension $m\ x\ m$, then there are m eigenvectors.

The main steps of PCA are described below [32]:

1) The training database consists of N face images which is same size. Take the all images X into a column vectors. The training set matrix X is the set of image vectors with

$$\text{Training set } X = \{X_1, X_2, X_3 \ldots, X_N\} \tag{4.1.1}$$

Where each X_i represents a d-dimensional column vector.

2) Calculate the mean vector of all of training images:

$$M = \frac{1}{N} \sum_{i=1}^{N} X_i \tag{4.1.2}$$

3) Subtract the mean from each column:

$$S = [(X1 - M), (X2 - M) \ldots (XN - M)] \tag{4.1.3}$$

Subtract the mean from each column to keep only distinguishing features of face images and delete the common features.

4) Calculate the covariance matrix:

$$C = S.S^T = \frac{1}{N} \sum_i \sum_j (x_{ij} - m)(x_{ij} - m)^T \tag{4.1.4}$$

Where S^T is the transpose of matrix S. Since S is $d \times N$ dimensional matrix, the size of covariance matrix C is $d \times d$.

5) Calculate the Eigen vectors of the covariance matrix:

$$CE = \lambda E \tag{4.1.5}$$

Here, $E = [e_1, e_{2,} \ldots, e_d]$, Where $e_1, e_{2,} \ldots, e_d$ are d eigenvectors of C.

6) Create an Eigen face by multiplying mean subtracted data matrix with the eigenvectors.

$$U = S.E \qquad (4.1.6)$$

Where U is eigenfaces, E is eigenvectors and S is subtracted matrix.

7) Computing the weight matrix:

$$\omega = U_i^T.S \qquad (4.1.7)$$

This weight matrix has been calculated by multiplying the transposed Eigen faces with the mean subtracted data matrix (S).

8) Test Image

After weight computation, all the above steps are used to compute the weight matrix of test images. After compute the projection matrix of the test images, the Manhattan Distance measure between the projection matrix of train images and projection matrix of test images are computed.

9) Output and Matching

To calculating the output, the image with smaller Manhattan Distance Measure is considered to be the face match score result.

4.2 Linear Discriminant Analysis

Linear Discriminant Analysis [32], is a popular feature extraction method for face recognition. Primarily, LDA method basically contains two-step: training and testing. Nowadays, LDA [32] [33] has become widely used technique and research area for extracting feature and reducing dimensions of images in pattern recognition. LDA tries to find the “best” project direction in which training samples belonging to different classes are best separated. In the training step, training samples use LDA to create the fisher space, and trained samples are mapped to the space for recognition.

Second step, an input face is anticipated to a similar fisher-space and classified by an appropriate classifier. The structure of LDA computing is shown in Figure 7.

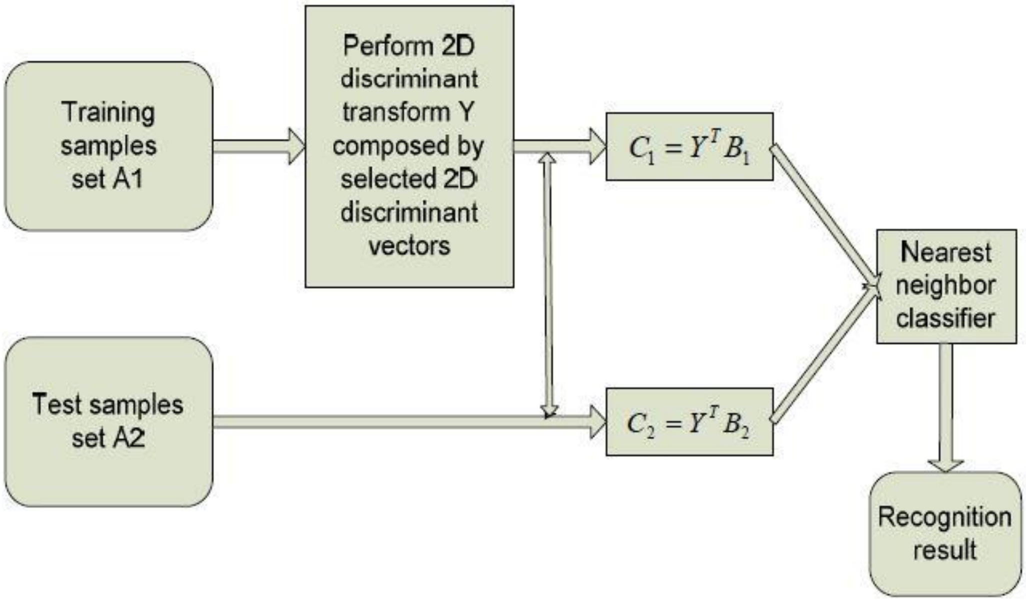

Figure 7: Face recognition system using LDA [32]

LDA is supervised learning method that consider the class labels but, on the other hand, PCA is unsupervised learning method that ignores the class labels. LDA is a topography data to a new space. LDA let us calculate a linear transformation that maps data from a high dimensional space to a lower dimensional subspace. The aim of LDA is to dimensionally reduce of the data while retaining as much as possibility of the variation present in the dataset. These vectors describe the subspace of face images. In figure 8 example of LDA subspace is illustrated.

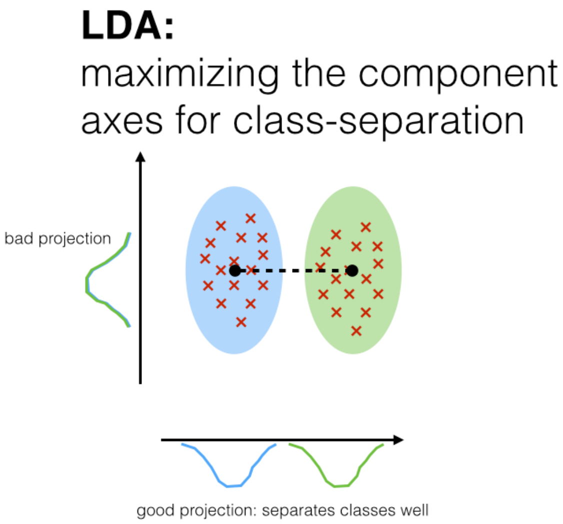

Figure 8: LDA subspace [32]

Face images from N-dimensional space transformed into C-1 dimensional subspace where N is number of pixels and C is number of subjects classes. LDA used similarity distance such as Manhattan and Euclidean between each pair of training and test face images to calculate recognition rate [34].

The main steps of performing LDA are explained as follows: [34]

1) Consider a set of N samples {x1, x2, …, xN} taking values in an n-dimensional space, and assume that each sample belongs to one C classes {X1, X2, …, X_C}.

$$\text{Training set } X = \{x_1, x_2, x_3, \ldots, x_N\} \tag{4.2.1}$$

2) Calculate the mean of each class:

$$m_i = \frac{1}{N} \sum_{i=1} x_i \tag{4.2.2}$$

3) The aim of LDA is to maximize $S_b$while minimize S_w. Compute S_w of the center images in the class.

$$S_i = \sum_{x \in X_i} (x - m_i)(x - m_i)^T \tag{4.2.3}$$

Where (m_i) is the mean of the sample of class.

The within class scatter matrix (S_W) is the sum of all scatter matrices.

$$S_W = \sum_{i=1}^{C}(S_i) \tag{4.2.4}$$

Where C is class number.

4) Calculate between class scatter matrix (S_B). The summation of the covariance matrices of the difference between the total mean and the mean of each class are computed.

$$S_B = \sum_{i=1}^{c} n_i\,(m_i - m)(m_i - m)^T \tag{4.2.5}$$

Where (m) is the mean of all images , (m_i) is the mean of images in the class. (n_i) is the number of images in the class.

5) Compute the eigenvectors of the projection matrix:

$$W = eig(S_W^{-1}\,S_b) \tag{4.2.6}$$

6) In the testing step, projection matrix of test face images are calculated and transformed into same subspace.

7) The projection matrix of test images is ready to compare with projection matrix of training image in subspace. Images are contrasted with Manhattan Distance. The training image that is the closest to the test image will be matched and used to identification.

4.3 Local Binary Patterns

The original Local Binary Patterns (LBP) was presented by Ojala et al. [18] and applied to grayscale images for texture explanation in computer vision. LBP has been introduced to be a robust feature for texture description and also is known as a very powerful texture classification technique [18] [19]. The LBP produces a descriptor or texture model using a set of histograms of the local texture neighborhood near each pixel and also LBP could be applied as an image processing operator.

The original type of Local Binary Patterns operator works in a 3×3-pixel block of an image. The LBP operator labels each pixel of an image, which thresholds the pixel's local neighborhood at its gray scale value into a binary number. The local neighborhood is around symmetric set of any number of pixels and radius. Then histogram of the labels could be applied as a texture descriptor. The normal LBP operator is illustrated in figure 9.

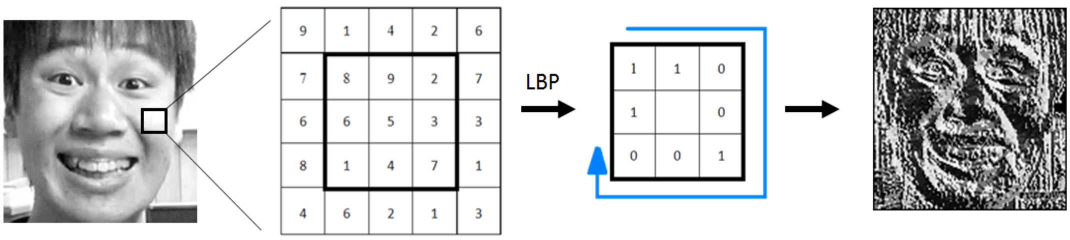

Figure 9: Example of an input image and the corresponding LBP image [18].

LBP produces a binary number 1 or 0. A binary number 1 indicates if the neighbor of the center pixel has bigger value than the center pixel. A binary 0 shows if the neighbor of center pixel is less than the center pixel. The eight neighbors of the center could then be illustrated with an 8-bit number as an unsigned 8-bit integer, making it a very well-set explanation. The detail of basic LBP is illustrated in figure 10.

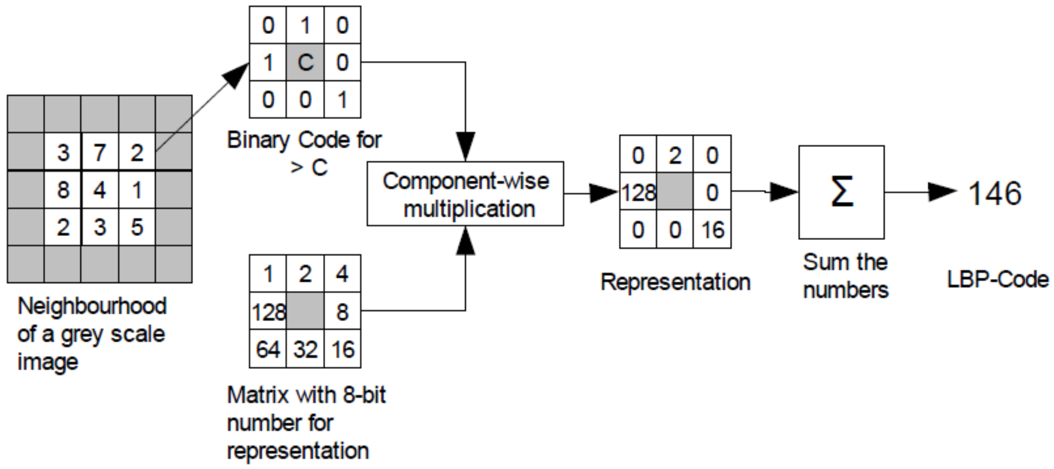

Figure 10: Example of how LBP-operator works [19].

The $\mathbf{LBP_{P,R}}$ operator is defined as

$$LBP_{P,R}(x_c, y_c) = \sum_{P=0}^{P-1} s(g_p - g_c)2^p,$$

$$s(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \qquad (4.3.1)$$

Where g_c is the gray value of the central pixel, g_p is the value of its neighbors, P is the total number of involved neighbors, and R is the radius of the neighborhood.

In exercise, this equation means the signs of the diversities in a neighborhood are explained as a P-bit binary number, resulting in 2^p distinct values for the LBP code. The local gray-scale distribution, i.e. texture, could thus be almost described with a 2^p-bin discrete distribution of LBP codes:

$$T \approx t(LBP_{P,R}(x_c, y_c)) \qquad (4.3.2)$$

In computing the $LBP_{P,R}$ the feature vector for a given $N \times M$ image sample ($x_c \in \{0, \dots, N-1\}, y_c \in \{0, \dots, M-1\}$), the central part is only noticed because a sufficiently large neighborhood cannot be used on the borders. The LBP code is computed for every pixel in the cropped image, and the distribution of the codes is used as a feature vector, denoted by S [12]:

$$S = t\left(LBP_{P,R}(x,y)\right), x \in \{\lceil R \rceil, \dots, N-1-\lceil R \rceil\}, y \in \{\lceil R \rceil, \dots, M-1-\lceil R \rceil\}$$

In this study the Manhattan Distance is used for PCA, LDA and LBP as similarity measure. Manhattan Distance is a metric in which the distance between two points is the sum of the (absolute) differences of their coordinates.

To found Manhattan Distance measure between the point P1 with coordinates (x1, y1) and the point P2 at (x2, y2) is:

$$D_m = |X_1 - Y_1| + |X_2 - Y_2| \tag{4.3.3}$$

To found Manhattan Distance measure of between two vectors X, Y of length n is

$$D_{(x,y)} = \sum_{i=1}^{n} |X_i - Y_i| \tag{4.3.4}$$

4.4 Decision Tree

Decision Tree [35] is one of the popular classifier and also is a tree-structured plan of a set of features to test in order to predict the output. Decision tree learning is a technique that generally applied in data mining and also is classification technique that results in flow-chart like structure where each node explains a test on a feature value and each branches demonstrate a result of the test. The leaves of tree demonstrate the classes [36].

The aim of Decision Tree is to make a model that is able to predicts the value of a target variable depends on multiple input variables. Each internal node shows one of the input variables. For each possible values, there are edges to children which belong to input variable. Each leaf shows a value of the target variable which has given the values of the input variables represented by the path from the root to the leaf. A tree can be able learned by splitting node of the source set into subsets based on a feature value test [37] [38]. When splitting no longer adds value to the predictions or, when the subset at all node has the same value of the target variable, the recursion is completed.

The important steps in Decision Tree are the selecting the best attribute. The Information Gain measure is used to select the best attribute at each node in the tree. Main steps of Decision Tree are as follows: [39]

1) Calculate entropy of training sample sets (S):

$$I(S) = -\sum_{i=1}^{m} P_i \log_2 P_i \tag{4.4.1}$$

Where m is distinct classes. P_i is the probability of class C in sample set.

2) Calculate the entropy of attribute A:

$$E(S, A) = \sum_{i=1}^{m} \frac{S_i}{S} I(S, A) \tag{4.4.2}$$

3) Calculate information gain of A:

$$Gain(S, A) = I(S) - E(S, A) \tag{4.4.3}$$

4) Find the best split of attribute (A). Compute Information Gain corresponding with segmentation points divided by $a_i (i = 1, 2, 3, \ldots, n - 1)$ and select the maximum value of Information Gain a_i as the split points of attribute classification.

4.5 Random Forests

Most of learning techniques contain some methods of randomization. Such goal is to select the best option at each step. Very popular one is Random Forest. Random Forest tries to produce randomization Decision Tree. For each repetition, the algorithm often generates superb predictor [24].

Random Forest's fundamental opinion is to find the average value of the noise. Very complex interaction tree could be obtained. The goal of Random Forest is an assemblage of Decision Tree by complex input space which can be calculated. It has illustrated that the collection of Random Forest, each Decision Tree trained randomly. Therefore, available data reduces the overfitting in comparison [24].

Ensemble learning refers to the algorithms that produce collections or ensembles of classifiers which learn to classify by training individual learners and fusing their predictions. Growing an ensemble of trees and getting them vote for the most popular class has provided a good enhancement in the accuracy of classification. Often, random vectors are built that control the growth of each tree in the ensemble [27].

Random Forest as classifier has two popular methods: bagging method and boosting method. In boosting method, extra weight is given to successive trees. For selecting prediction, a weighted vote is selected. Second method is bagging, trees are made independently by bootstrap samples of the data set and a simple majority vote is taken for predictions. Random Forests add an additional layer of randomness to bagging. In addition to constructing each tree by various bootstrap sample of the data, Random Forests change the manner of the regression or classification trees production. In Decision Trees, node is splitting with the best split via all elements, but in Random Forest, is not similar to standard Decision Tree. Node is split with randomly selected feature because each time is selecting features randomly. Random forest can achieve better performance compared with the various classifiers technique such as Neural Networks, Discriminant Analysis and Support Vector Machines [28].

In a Random Forest, pruning is not necessary and biggest tree are grown without pruning. The root of each tree involves a various bootstrap sample randomly taken from the original training data. The leaves of a tree include parameters having same class labels. The predictions for new data are the class title of the items in the leaf where the data achieved [40].

Summarized Random Forest algorithm: [41]

For b = 1 to B:

- Draw n_{tree} bootstrap sample from the training data.

- bootstrap samples, grow an unpruned classification tree, with the following modification: At each node, rather than choosing the best split among of all prediction, randomly sample n_{tree} of the predictors and choose the best split from among these variables.

- New data predict by aggregating the predictions of n_{tree} trees (random forest can used for classification (majority vote) and regression (average)).

Chapter 5

DATABASES

There are various number of face databases that are in use for training and testing of face recognition algorithms. Widely use facial databases are FERET, LFW, ORL, JAFFE, YALE, AT&T and etc. The goal of creating these databases generally is to use for face recognition and face detection. These databases are created by small group of researcher.

Some of face databases have been created under controlled situation to motivate the research of specific parameters on the face recognition problem. These specific parameters include such variables as lighting, different pose, different position, background. For controlling parameter of image acquisition there are several applications that capable this parameter, also there are many applications that have not any control or little over such parameters in face recognition.

In this chapter, two publicly available face databases have been applied for recognition and classification. The number of individuals and images in both databases are shown in table 1.

Table 1: Publicly available face datasets

Dataset	Number of images	Number of subjects
FERET	14126	1199
JAFFE	213	10

In my thesis, FERET and JAFFE databases are applied to show the performance of face recognition and classification. Both databases are explained in details as follow.

5.1 FERET Database

FERET [42] has been produced by Phillis and Rauss in 1994 that FERET database consists well quality grayscale images of 1199 different individuals and over 14,100 number of images. FERET database is one of the popular database, mostly applied for an estimate of face recognition system with different methods such as PCA, LDA, LBP and etc., also has been applied by many researchers for classification and recognition. The goal of FERET program is to extend algorithms on a joint database. The result of using FERET database are shown in literature, because of every researcher are using different scoring method and images from FERET database did not produce a direct comparison among algorithms.

More important thing in the FERET database and tests clarify the current state of the art in face recognition and point out general directions for future research. FERET database tests allow to find overall weakness in pattern recognition by computer vision researchers [40].

The face images in FERET database have a different kind of pose, and some variation in expression and illumination. The faces are noise-free, without background clutter and have consistent lighting. Figure 11 is shown cropped and resized face images, after face detection from FERET database.

Figure 11: Sample face images from the FERET Database

5.2 JAFFE Database

The JAFFE [43] database was created and congregated by three researchers from Psychology Department at Kyushu University namely Michael Lyons, Miyuki Kamachi, and Jiro Gyoba. It consists 213 images from 10 Japanese female models that have different facial expression. The JAFFE database has been selected because it contains less number of classes and more images for each class to compare with FERET database.

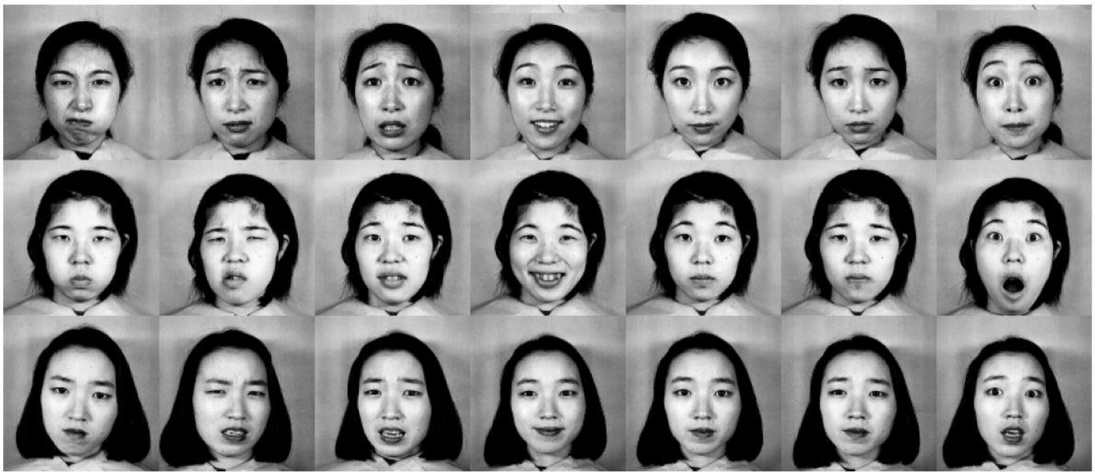
Figure 12: Samples from Japanese female facial expression image set.

Chapter 6

EXPERIMENTS AND RESULTS

In this chapter, the explanation of our implementation and the comparison performance of the Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Local Binary Patterns (LBP) and also an explanation of different classifiers that we used such as Decision Tree (DT) and Random Forest (RF) are going to be scrutinized.

6.1 Experimental setup

All methods discussed are implemented on Matlab2015b. Windows 10 Pro with Core i7 CPU and 8 GB RAM in a Personal Computer environment.

Initially, we conducted experiments via part of FERET face images of database in the training set. In this research, 1000 images; related to 250 individuals, have been used.

Then, in order to compare the result of recognition rate, JAFFE database have been used. 200 images which have been related to 10 subjects were chosen from JAFFE database.

Considering both databases, 50% of images are being called as training set while 50% of images are being described as testing set. The detail of both databases are shown in table 2.

Table 2: Databases used in experiments in details

Database	FERET	JAFFE
Number of subjects	250	10
Total number of images	1000	200
Number of Train subjects	500	100
Number of Test subjects	500	100

Generally, there are two main steps to recognize facial images: training and testing. In training phase first of all, a dataset of resized, cropped and normalized images are given to the feature extractor algorithms, then the output of train images is obtained. In test phase a dataset of resized, cropped and normalized images are given to the feature extractor algorithms. The last step is to compare available trained images and test images to find the similarities.

The size of images may be different in the database, therefore to achieve a high accuracy, it is important to make all image size unify. In order to make FERET database images to equal size with appropriate values, the images are resized to 80×65 from 384×256. Also for JAFFE database images, to achieve reasonable values, the images are resized to 100×100 from 250×250. In JAFFE database, we tried different image size but best accuracy achieved by images which resized to 100×100.

Later, PCA, LDA and LBP are applied respectively, in different experiments on all images to extract the features for training and test datasets. To classify the extracted features, Decision Tree and Random Forest have been used as classifiers. The details of each experiment are explained in details in following subsections:

6.2 Principal Component Analysis based experiment and results on FERET and JAFFE databases

Primarily, Principal Component Analysis on different databases namely, FERET database and JAFFE database are applied. Databases are divided into two sets which 500 images for training set and 500 images for testing set are used in FERET database and in JAFFE database 100 images for training set and 100 images for tasting set are used. As first steps, the cropped, resized and normalized images that have been explained in preprocessing chapter are read. In the next step, mean value (which is the average of the number, in the other words it is the sum divided by the count) have found of all training images and covariance matrix are calculated (covariance matrix is also known as dispersion matrix which its elements are i^{th} and j^{th} elements of a random vector). In the final step eigenvectors are calculated to find the projection matrix.

The same steps applied for training set as explained above; are also applied for testing set in order to compare our training and test projections. Manhattan Distance Measure is used to find the minimum distance between train and test images. In addition to that, the most resembling image to a test image is the one with the smallest distance.

To calculate the recognition accuracy, the equation below is used:

$$\text{Accuracy} = (\text{number of correct images} / \text{number of all images}) \times 100 \qquad (6.2.1)$$

We measured accuracies of the systems in each test setup separately. In this section, the result of PCA on FERET and JAFFE databases are shown in detail. The face recognition accuracies are computed and shown in table 3 and table 4.

Table 3: The results of recognition rate on Principal Component Analysis with FERET database

Image Size	Eigenvectors	Accuracy (%)
80×65	30	90.60
80×65	50	91.40
80×65	100	**91.60**
80×65	200	91.60
80×65	300	91.60
80×65	400	91.60

Table 4: The results of recognition rate on Principal Component Analysis with JAFFE database

Image Size	Eigenvectors	Accuracy (%)
100×100	10	85.00
100×100	20	88.00
100×100	30	**90.25**
100×100	40	90.25
100×100	50	90.25
100×100	60	90.25

According to these experiment results, the recognition rates on different databases are shown in Table 3 and Table 4. We achieved the recognition rate (91.6%) in the 100 most significant eigenvectors from total 500-1 (N-1, where N is the number of subjects

× number of images of each subjects for training) eigenvectors on FERET database. Recognition rate of (90.25%) is achieved with 30 eigenvectors from total 100-1 eigenvectors in JAFFE database.

6.3 Linear Discriminant Analysis based experiment and results on FERET and JAFFE databases

In this study, LDA is applied as second feature extraction on both FERET and JAFFE databases. Initially, the same as the Principal Component Analysis, databases are divided into two sets including: 50% images for training set and 50% of images for testing set. Then, the researcher took the training set of all images into a row vector and cropped, resized and normalized images which could be observed, vividly. After, mean value is computed, equal to sum of training image which have been divided by number of them. In the next step, the in-between class scatter matrix Sb is calculated; the scatter matrix (a statistic that is used to make estimates of the covariance matrix and within class Sw). Eigenvectors and eigenvalues for the scatter matrix are computed and sorted eigenvectors by descending order eigenvalues, and select eigenvectors with largest eigenvalues. In the last step, the eigenvector of matrix (weight matrix (W)) is calculated.

As explained above, the same steps that have done for training set are applied for testing set. To compare the test image's projection matrix with the projection matrix training image, the Manhattan distance measure is used. The training image which is the closest to the test image is the main result.

Below equation is used to calculate the recognition accuracy:

$$\text{Accuracy} = (\text{number of correct images} / \text{number of all images}) \times 100 \qquad (6.3.1)$$

In order to find more accurate result in LDA, we calculated the accuracies of the chosen various classes for LDA on FERET and JAFFE databases. Table 5 and table 6 clarify the recognition rate of LDA technique.

Table 5: The result of recognition rate on Linear Discriminant Analysis with FERRET database

Image Size	Classes	Accuracy (%)
80×65	30	85.20
80×65	50	87.40
80×65	100	90.75
80×65	150	**92.50**
80×65	200	92.50
80×65	249	92.50

Table 6: The results of recognition rate on Linear Discriminant Analysis with JAFFE database

Image Size	Classes	Accuracy (%)
100×100	4	51.40
100×100	5	62.00
100×100	6	83.20
100×100	7	**88.75**
100×100	8	88.75
100×100	9	88.75

The results of LDA method on FERET dataset and JAFFE dataset are shown that the maximum recognition rate with 150 classes from total 250 classes is (92.50%) in FERET dataset and it can also be concluded that the maximum rate obtained by using 7 classes from 10 classes is (88.75%) in JAFFE dataset.

6.4 Local Binary Patterns based experiment and results on FERET and JAFFE databases

Local Binary Patterns as third feature extractor in both FERET and JAFFE database are applied. As the first steps, the images in FERET database is cropped and resized to 78×66, 80×65, 80×64 and 78×66 pixels for 36, 25, 16 and 9 partitions respectively. For JAFFE database images are resized to 98×98, 96×96, 100×100 and 100×100 pixels for 49, 36, 25 and 16 partitions respectively. After that the images are normalized. As the next step, the resized and partitioned images are given as an input to feature extractor which is LBP. LBP is applied to each block to extract features. Thus, each face image was described by LBP histogram. Therefore, Local Binary Patterns histograms are extracted and concatenated into one feature histogram to represent the whole image.

As explained above, the same steps that has done for training set are applied for testing set. The results are in (%) and are highlighted in table 7 and 8 for the recognition rates.

Finally, after calculating LBP for each block, the researcher concatenated them into a single vector and; then, Manhattan Distance Measure is used to find the minimum distance between the test set and training set and compare them for face recognition. The recognition accuracy was calculated as follows:

$$\text{Accuracy} = (\text{number of correct images} / \text{number of all images}) \times 100 \quad (6.4.1)$$

Table 7: The results of recognition rate on Local Binary Patterns with FERET database

Image Size	Image Resize	Partition	Accuracy (%)
80×65	78×66	6×6	**93.20**
80×65	80×65	5×5	92.20
80×65	80×64	4×4	92.00
80×65	78×66	3×3	91.20
80×65	80×64	2×2	89.40

Table 8: The results of recognition rate on Local Binary Patterns with JAFFE database

Image Size	Image Resize	Partition	Accuracy (%)
100×100	98×98	7×7	90.25
100×100	96×96	6×6	**91.25**
100×100	100×100	5×5	88.00
100×100	100×100	4×4	85.75
100×100	99×99	3×3	85.00

According to our experiments, the recognition rates are slightly different on each dataset. In FERET database, images with 78×66 sizes which are divided into 36 partitions have the best performance with recognition rates of (93.20%) and in JAFFE database, images with 96×96 sizes which are divided into 36 partitions have the best performance with the classification rate of (91.25%).

Comparison accuracy of different feature extractors (Principal Component Analysis, Linear Discriminant Linear, and Local Binary Patterns) are illustrated in table 9.

Table 9: Feature extractor recognition accuracy

	Train Image	Test Image	PCA	LDA	LBP
FERET_DB	500	500	91.60%	92.50%	93.20%
JAFFE_DB	100	100	90.25%	88.75%	91.25%

According to these experiments, the recognition rates between approaches are slightly different on each dataset. However, LBP has better performance compared to the other approaches on FERET dataset and JAFFE dataset.

6.5 Decision Tree by using PCA, LDA, LBP experiments and results on FERET and JAFFE databases

Two known classifiers are used as clustering method; Decision Tree and Random Forest which will be explained in the next subsection. Decision tree is one of the most popular method for classification. To apply the Decision Tree method, firstly features are extracted from face images with Principal Component Analysis, Linear Discriminant Analysis and Local Binary Patterns respectively. Then, an empty Decision Tree is created which contains one node and all information of data sample, whose parent is the root. The next step is splitting node with the best features which Information Gain measure is used; to find IG of each feature for splitting. Entropy and conditional entropy (the maximum value of entropy can be 1 when is maximally impure) are calculated to find information gain. Next, feature with the highest information gain is selected. Decision node is created and divided the training node by finding a best split. In the last step, the Decision Tree grows by mean of progressive subdivisions until the leaf is constant or all inputs have the same feature values. When we reach this stage, the root node will not split further and automatically become a final node.

In the test part, after the extraction of features and the projection test matrix calculation with PCA, LDA and LBP, the predict function will be used to return a vector of predicated class labels for the predicator data in matrix to predict class.

$$y = \text{predict(tc,Prj\_image\_PCA\_test\_Matrix')} \tag{6.5.1}$$

Where tc is Decision tree output.

Below equation is used to calculate the Decision Tree accuracy:

$$\text{ind1} = \text{sum((y-testlabel') ==0)/length(testlabel)} \times 100 \tag{6.5.2}$$

Where the number of outputs equal to zero; the subtraction of predict function (y) and actual amount (test label). To findings clarify the correct predicted number which can be divided to test label to achieve the accuracy.

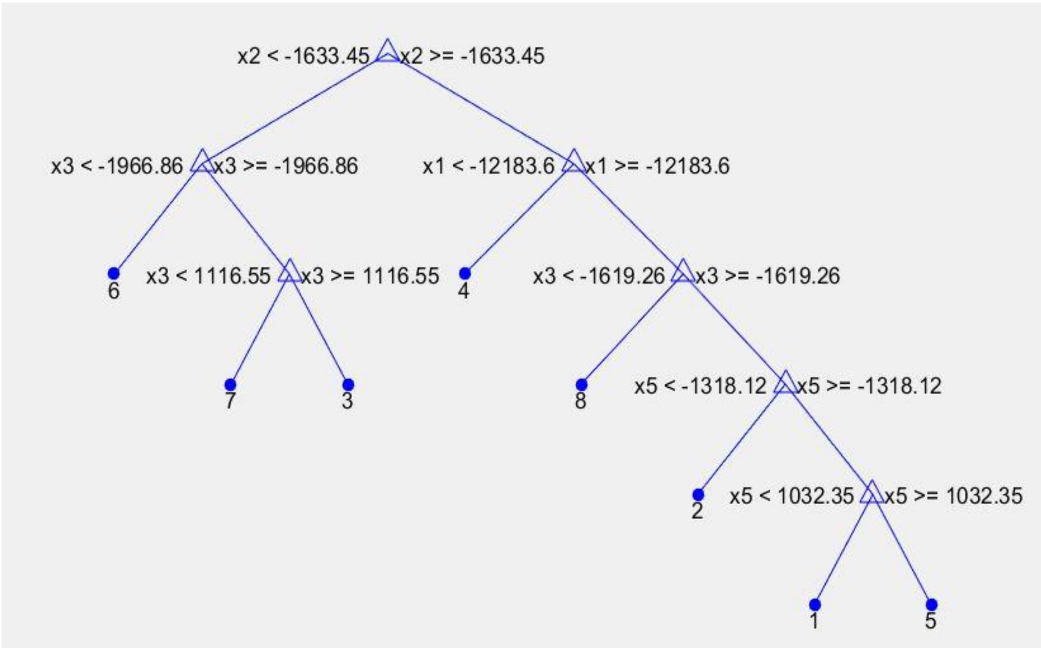

Figure 13: Classification tree

The above figure is one simple case of Decision Tree on JAFFE database with tree depth of 5. X2 is the root node of a decision tree; considered as the second feature in our dataset and if it is less than -1633.45 (x2 < -1633) the command will go to left otherwise; go to right and check each node until to reach the final node.

The classification rate for different databases and different tree depths are shown in table 10 and table 11 as follows.

Table 10: The results of classification rate of Decision Tree on PCA, LDA and LBP with FERET database

Tree depth	PCA_DT	LDA_DT	LBP_DT
5	7.2	8.4	5
10	11	11.8	6.8
15	17.2	16.6	10.8
20	21	**18.8**	**12.2**
25	**21.6**	18.8	12.2

In table 10, it is shown that we have not been able to achieve the reasonable accuracy in FERET database. Since the Decision Tree is powerful to classify the data with less number of classes and more images of each class. The FERET database have been used contains 250 classes and different trees depths are tried, but the maximum accuracy for PCA_DT is (21.6%) on tree depth 25, (18.8%) for LDA and (12.2%) for LBP on tree depth 20 are achieved.

Table 11: The results of classification rate of Decision Tree on PCA, LDA and LBP with JAFFE database

Tree Depth	PCA_DT	LDA_DT	LBP_DT
2	30	27.5	23
4	47	51.25	41
6	60	68.75	59
8	75	70	69
10	**81**	**76**	**83**

In table 11, it is obvious that due to the less number of classes (10 classes) and 20 number of images for each class high accuracy in PCA_DT, LDA_DT and LBP_DT on tree depth 10, (81%), (76%) and (83%) are achieved in JAFFE database respectively.

6.6 Random Forest by using PCA, LDA and LBP experiments and results on FERET and JAFFE databases

In this study, Random Forest is used as second classifier. Random Forest algorithm acts as a large collection of de-correlated trees.

First, the features of face images are extracted with Principal Component Analysis, Linear Discriminant Analysis and Local Binary Patterns respectively and projection matrix of each method is computed. In the next step, numbers of random subsets are created from our trained matrix sample which contains features of samples and trained classes where for each random subset, a decision tree is created and all trees are trained independently. Then, all of the trained decision trees are used to create a ranking of classifiers. In the training step, as mentioned in sub section 6.5, for decision trees information gain (entropy-conditional entropy) are calculated and features that have highest information gain are selected. However, in Random Forest due to the high number of Decision Trees trained for each round, features are selected randomly and calculate Information Gain from these features. Finally, class prediction is used to find prediction of each decision then majority vote is used to select class with high vote to find final node.

In the test part, after features are extracted and projection test matrix is calculated with PCA, LDA and LBP respectively, predict function used (returns a vector of predicted class labels for the predictor data in matrix) to predict our class labels that we trained with test projection matrix.

Predict function have two parameters, first one is forest matrix that we trained and second one is projection test matrix which used to returns a vector of predicted class labels. The equation below used to compute accuracy of Random Forest classification.

$$ind1 = sum((y-testlabel') ==0)/length(testlabel)*100 \tag{6.6.1}$$

Where the number of outputs equal to zero; the subtraction is of predict function (y) and actual amount (Test label). We sum up the correct predicted number, then divided to total labeled dates in order to achieve the accuracy.

Table 12: The results of classification rate of Random Forest based on PCA features with FERET database

	Tree Depth					
		5	10	15	20	25
Number of Trees	50	75.6	76.94	78.3	77.26	78.32
	100	83.1	83.73	84.61	84.46	84.64
	200	87.03	88.12	87.82	88.26	87.12
	300	87.98	88.18	88.9	88.8	89.1
	400	89.12	89.38	88.9	89.16	**89.82**

Table 13:The results of classification rate of Random Forest based on PCA features with JAFFE database

	Tree Depth					
		5	10	15	20	25
Number of Trees	50	96.2	94.5	95.97	95.625	96.25
	100	96.3	96.5	96.62	97.1	96.7
	200	97.15	97.25	97.65	97.25	97.625
	300	96.45	97.62	96.625	98.25	97
	400	97.125	97.47	97	97.25	**97.75**

Table 14: The results of classification rate of Random Forest based on LDA features with FERET database

	Tree Depth					
Number of Trees		5	10	15	20	25
	50	72.3	72.5	72.2	72.8	72.6
	100	77.9	79.8	79.6	78.5	81
	200	82	82.4	81.1	82.1	82.9
	300	84.5	83.5	84.5	83.5	84.8
	400	84.9	84.5	83.4	83.9	**85.3**

Table 15: the results of classification rate of Random Forest based on LDA features with JAFFE database

	Tree Depth					
Number of Trees		5	10	15	20	25
	50	91.5	90.625	92.5	89.375	92.5
	100	90.625	90.7	89.37	90	91.25
	200	92.5	91.25	92.5	91.25	92.5
	300	91.25	91.25	91.25	91.75	92.62
	400	92.375	91.5	92.5	93.12	**93.62**

Table 16: The results of classification rate of Random Forest based on LBP features with FERET database

	Tree Depth					
Number of Trees		5	10	15	20	25
	50	49.6	50.1	49.6	50.1	50.4
	100	74.1	73.2	74.1	73.5	73.2
	200	84.7	84.8	84.7	84.8	85.88
	300	89.3	89.3	89.3	89.3	90
	400	91.9	92.1	91.9	92.1	**92.8**

Table 17: The results of classification rate of Random Forest based on LBP features with JAFFE database

	Tree Depth					
Number of Trees		5	10	15	20	25
	50	97.62	97.125	98.84	97.625	98.7
	100	97.65	98.75	89.62	99.1	99.5
	200	**100**	100	100	100	100
	300	100	100	100	100	100
	400	100	100	100	100	100

AS illustrated in Random Forest tables, to discover the best accuracy on PCA, LDA and LBP features, Random Forest classifiers are applied with various number of trees 50,100,200,300 and 400 and different number of tree depths were 5,10,15,20 and 25. To calculate accuracy in random forest, features are selected randomly to create Decision Tree. This extensive program is executed; approximately, 30 times for each number of trees and tree depth (approximately, we executed 4500 times) and the average calculated to achieve the classification rate. We achieved random forest classification rate for PCA, LDA and LBP features; respectively, as follows (89.82%), (85.3%) and (92.8%) with 400 number of Decision Trees and 25 for tree depth on FERET database. In JAFFE database, classification rate (97.75%), (93.62%) and (100%) with 400 number of decision trees for PCA and LDA features with tree depth 25 and 200 number of trees for LBP features are achieved, respectively. remarkably, by increasing the number of trees and tree depth, high classification rate is achieved in the Random Forest; however, it is valuable to mention that the increasing tree depth and number of tree do not always be lead to the increase of accuracy.

Face recognition results on the FERET dataset demonstrate that LBP technique with 8 neighbors and radius 2 compare with other methods such as PCA and LDA as feature extractors achieves the best accuracy and also, face recognition results on the JAFFE dataset shown that LBP with 8 neighbor sand radius 2 achieves the best accuracy.

Decision Tree results shown that to achieve high accuracy, it is necessary to choose database with less number of classes and more images for each class. The results of Decision Tree on FERET database demonstrate that, we could not achieve high accuracy because of 250 number of classes and maximum classification rate is belong to feature which extracted with PCA method whereas, on JAFFE database, we achieved high classification rate. The results shown that the features which extracted with LBP has better performance to compare with other methods due to less number of classes and more images for each class in JAFFE database.

In Random Forest classifiers in order to find the best accuracy different number of trees and different tree depths are applied. The results illustrate that on FERET and

JAFFE databases the features which extracted with LBP method has best performance to compare with features which extracted from other features extraction.

Chapter 7

CONCLUSION

In this thesis, we carried out experiments on several state-of-the-art face recognition and classification techniques. We compared the performance of Local Binary Patterns approach, Principal Component Analysis approach and Linear Discriminant Analysis as face recognition and Decision Tree and Random Forest as classification. Different databases for face images namely FERET, JAFFE are used to compare these approaches. Databases are divided into two parts including training images and test images and PCA, LDA and LBP have been applied. In classification part, Random Forest and Decision tree are used as classification technique to find out which method has a better performance in different databases.

First of all, we obtained the fact that the database used has affected the classification accuracy a lot and this should be taken into account when doing experiments. In addition to that, after applying different feature extractor on both FERET and JAFFE database, we found that LBP performance is better than other methods compare with PCA and LDA in face recognition, but is not very different between methods. For the classification with Decision Tree, we found the best performance on features that are extracted by using the LBP. For Random Forest we achieved best classification accuracy in features that are extracted with LBP on both databases.

As future works, we intend to develop our experiments on the different available public database that are sensitive to different poses, illumination, occlusion, expression and etc. Additionally, various and more robust classifiers could be used such as Support Vector Machine (SVM) and Neural Network and also a different feature extractor such as EBGM, HOG to compare the results.

REFERENCES

[1] Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4), 399-458.

[2] Tolba, A. S., El-Baz, A. H., & El-Harby, A. A. (2006). Face recognition: A literature review. *International Journal of Signal Processing*, 2(2), 88-103.

[3] *Handbook of Face Recognition*, S.Z. Li and A.K. Jain, eds. Springer, 2005.

[4] Singh, S., Sharma, M., & Rao, D. N. S. (2011). Accurate face recognition using pca and lda. *In International Conference on Emerging Trends in Computer and Image Processing* (pp. 62-68).

[5] Mahmud, F., Haque, M. E., Zuhori, S. T., & Pal, B. (2014, April). Human face recognition using PCA based Genetic Algorithm. *In Electrical Engineering and nformation & Communication Technology (ICEEICT), 2014 International Conference on* (pp. 1-5). IEEE.

[6] Etemad, K., & Chellappa, R. (1997). Discriminant analysis for recognition of human face images. *JOSA A*, 14(8), 1724-1733.

[7] Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1), 71-86.

[8] A. Javed, “Face Recognition based on Principal Component Analysis”, *Image Graphics and Signal Processing*, vol. 2, pp. 38-44, 2013

[9] Goldstein, A. J., Harmon, L. D., & Lesk, A. B. (1971). Identification of human faces. *Proceedings of the IEEE*, 59(5), 748-760.

[10] Sirovich, L., & Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Josa a*, 4(3), 519-524.

[11] Turk, M. A., & Pentland, A. P. (1991, June). Face recognition using eigenfaces In Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., *IEEE Computer Society Conference on* (pp. 586-591). IEEE.

[12] Pujol, P., Macho, D., & Nadeu, C. (2006, May). On real-time mean-and-variance normalization of speech recognition features. *In Acoustics, Speech and Signal rocessing, 2006. ICASSP 2006 Proceedings*. 2006 IEEE International Conference on (Vol. 1, pp. I-I). IEEE.

[13] Baek, K., Draper, B. A., Beveridge, J. R., & She, K. (2002, March). PCA vs ICA: A Comparison on the FERET Data Set. In *JCIS* (pp. 824-827).

[14] Delac, K., Grgic, M., & Grgic, S. (2005). Independent comparative study of PCA, ICA, and LDA on the FERET data set. *International Journal of Imaging Systems and Technology*, 15(5), 252-260.

[15] Huang, S. M., & Yang, J. F. (2013). Linear discriminant regression lassification for face recognition. *IEEE Signal Processing Letters*, 20(1), 91-94.

[16] Lu, J., Plataniotis, K. N., & Venetsanopoulos, A. N. (2003, September). Boosting linear discriminant analysis for face recognition. *In Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on* (Vol. 1, pp. I-657). IEEE.

[17] Lu, J., Plataniotis, K. N., & Venetsanopoulos, A. N. (2003). Face recognition using LDA-based algorithms. *IEEE Transactions on Neural networks*, 14(1),195-200.

[18] Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1), 51-59.

[19] L. Nanni, A. Lumini. (2010). A Local Binary approach based on local binary patterns and its Variants texture descriptor. Expert syst. Appl. 37, 7888-7897.

[20] Turtinen, M., Pietikainen, M., & SilvÉn, O. (2006). Visual characterization of aper using isomap and Local Binary Patterns. *IEICE transactions on information and systems*, 89(7), 2076-2083.

[21] Meena, K., & Suruliandi, A. (2011, June). Local Binary Patterns and its variants for face recognition. In Recent Trends in Information Technology (ICRTIT), 2011 *International Conference on* (pp. 782-786). IEEE.

[22] Huang, D., Shan, C., Ardabilian, M., Wang, Y., & Chen, L. (2011). Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(6), 765-781.

[23] Bittencourt, H. R., & Clarke, R. T. (2004). Feature selection by using classification and regression trees (CART). *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*.

[24] Ratanamahatana, C. A., & Gunopulos, D. (2002). Scaling up the naive Bayesian classifier: Using decision trees for feature selection.

[25] Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12), 2037-2041.

[26] Mohseni, S., Kordy, H. M., & Ahmadi, R. (2013, September). Facial expression ecognition using DCT features and neural network based decision tree. *In ELMAR, 2013 55th International Symposium* (pp. 361-364). IEEE.

[27] Kouzani, A., Nahavandi, S., & Khoshmanesh, K. (2007, January). Face classification by a random forest. *In 2007 IEEE Region 10 Conference: TENCON 2007* (pp. 1-4). IEEE Xplore.

[28] Salhi, A. I., Kardouchi, M., & Belacel, N. Fast and efficient face recognition system using Random Forest and Histograms of Oriented Gradients. *In 2012 IOSIG-Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG).*

[29] Weyrauch, B., Heisele, B., Huang, J., & Blanz, V. (2004, June). Component-sed face recognition with 3D morphable models. *In Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on* (pp. 85-85). IEEE.

[30] Pujol, P., Macho, D., & Nadeu, C. (2006, May). On real-time mean-and-variance normalization of speech recognition features. *In 2006 IEEE, International Conference on Acoustics Speech and Signal Processing Proceedings* (Vol. 1, pp. I-I). IEEE.

[31] Z. Gengtao, Z. Yongzhao, Z. Jianming, “Facial Expression Recognition Based on elective Feature Extraction”. *Sixth International Conference on Intelligent Systems Design and Applications*, 2006. ISDA '06. Volume 2, 412-417, Oct. 2006.

[32] O’Toole, A. J., Deffenbacher, K. A., Valentin, D., & Abdi, H. (1993). Low-imensional representation of faces in higher dimensions of the face space. *JOSAA*, 10(3), 405-411.

[33] Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM computing surveys* (CSUR), 35(4), 399-458.

[34] Jelšovka, D., Hudec, R., & Brezňan, M. (2011, August). Face recognition on FERET face database using LDA and CCA methods. *In Telecommunications and Signal Processing (TSP), 2011 34th International Conference on* (pp. 570-574). IEEE.

[35] Gayatri, N., Nickolas, S., Reddy, A. V., Reddy, S., & Nickolas, A. V. (2010). Feature selection using decision tree induction in class level metrics dataset for oftware defect predictions. *In Proceedings of the World Congress on Engineering and Computer Science* (Vol. 1, pp. 124-129).

[36] Gupta, G. K. (2014). Introduction to data mining with case studies. PHI Learning Pvt. Ltd.

[37] Quinlan, J. R. (2014). C4. 5: programs for machine learning. *Elsevier*.

[38] Jain, A. K., & Dubes, R. C. (1988). Algorithms for clustering data. Prentice-Hall, Inc.

[39] Hu, J., Deng, J., & Sui, M. (2009, December). A new approach for decision tree ased on principal component analysis. *In Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on* (pp. 1-4). IEEE.

[40] L. Breiman. Random forest. Machine Learning, vol .45, 2001, pp. 1-33.

[41] Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical 4 learning (Vol. 1). *Springer*, Berlin: Springer series in statistics.

[42] Phillips, P. J., Moon, H., Rizvi, S. A., & Rauss, P. J. (2000). The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10), 1090-1104.

[43] Lyons, M. J., Akamatsu, S., Kamachi, M., Gyoba, J., & Budynek, J. (1998). The Japanese female facial expression (JAFFE) database. 1998). *http://www. kasrl, org/jaffe, html.*