Face Recognition Using Probabilistic Model

By

NS Jabeen Malik

Supervisor

Col(R) Imran Touqir, PhD

A thesis submitted to the faculty of Electrical Engineering Department, Military College of Signals, National University of Sciences and Technology, Islamabad, Pakistan, in partial fulfilment of the requirements for the degree of MS in Electrical (Telecommunication) Engineering

AUGUST 2022

Abstract

The effectiveness of real-time facial recognition systems is of great importance. In this work, hidden Markov model is used applying three states for face recognition with further reduction of feature vector coefficients. A sequence of overlapping blocks have been used for dividing the face images. The model is trained using an observation sequence containing the eigenvectors and coefficients of these blocks, and each subject is given a separate HMM. Applying DWT during the preprocessing stage has reduced the computational complexity of the proposed model and maximum noise filtration has been achieved. Moreover, singular value decomposition has been applied on face images and test images are rejected and accepted based on threshold singular value determined empirically. For feature extraction, principal component analysis is employed. Recognized test images are identified dependent on majority criteria utilizing different observation vectors of image. Experimental results of Yale and ORL directory in noisy environments such as Salt and Pepper and noise free environments with show that the recognition authenticity of the presented model is comparative to the existing methods with reduced computational cost.

Declaration

I certify that the work *"Face Recognition Using Probabilistic Model"* exhibited in this thesis has not been submitted in support of any other award or educational qualification either at this institution or at elsewhere.

Acknowledgements

All praises to Almighty ALLAH who showered his blessing and able me to complete such exploration work.

I would especially like to thank for an amazing support that my supervisor Col(R) Imran Touqir, PhD has been, throughout the course of my thesis. He helped me a lot to able me in completing my MS Degree, they were all time readily available for guidance and to help me throughout my thesis.

Lastly, I would like to show gratitude to my (late) mother (Mrs. Imtiaz Bibi), and my sisters for all their care, love and support through my times of stress and excitement.

Dedication

I dedicate my work to,

My (Late) mother who always encouraged my higher education, for her prayers, love and motivation and sacrifices all along.

My supervisor and sisters for their support and patience during all the phases of my MS.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Hidden Markov model	HMM
Face Recognition	FR
Principal Component Analysis	PCA
Discrete Wavelet Transform	DWT
Local Binary Pattern	LBP
Discrete Cosine Transform	DCT
Singular Value Decomposition	SVD
Support Vector Machine	SVM
Linear Discriminant Analysis	LDA
Independent Component Analysis	ICA
Neural Networks	NN
Elastic Bunch Graph Matching	EBGM
Dynamic Link Architecture	DLA
Receiver Operating Characteristics	ROC

Chapter 1

INTRODUCTION

1.1 Motivation

The individual human face plays a major role in the authentication and recognition of individuals and social experiences. This evolution has been gradually growing in electronics transactions for fast and effective identification and user authentication requests. For safety purposes, using human face, facial recognition technology has produced a relatively considerable amount of work in the past few years. There are many biometric identification methods, including the recognition of iris, fingerprints and face recognition (FR) [1]. The benefit of FR compared to other biometric systems is that face images cannot be detected intrusively.

1.2 Face Recognition

Facial recognition is the major tasks that people undertake so frequently that they do not even realize how often they accomplish it. Although automated facial recognition has been studied since the 1960s, the scientific world has only recently become aware of it. One of the most advanced methods for analyzing a person face has been developed in the recent decade. However, the scientific community is currently grappling with the issue of facial recognition system reliability.

1.3 General Face Recognition System

A mechanism consists of mainly four stages is generally involved in an automated FR system. The structural outline of a FR system is presented in Fig 1.1. Because of noise or lighting conditions, the test images produced from image devices such as cameras, might not appropriate for identification. Therefore, the first step is to eliminate noise, improve the lighting and normalize the color in the preprocessing stage. Then faces must be stored in instructional images and segmented. In order to create a feature vector, next step is to extract some predefined features. These features must provide distinctive details in the database about each individual so that we may identify the respective images depends upons these features. Lastly, the final level is the

identification. In this stage, a random image is classified based on feature vectors of a class using previously observed features.

Fig 1.1: Block diagram of a general FR system

1.3.1 Preprocessing

Preprocessing or image quality enhancement is the first step in FR system and many computer vision applications. The most commonly used approach for the image improvement when datasets have variants in illumination is histogram equalization [2]. By smoothing the pixel intensity histograms, histogram equalization improves recognition results for the images including the images under controlled lighting. In [3], Shan et al. suggested the Gamma concentration approach and implemented it to histogram equalization and illumination normalization. Their work also included the introduction of a region-based technique for gamma and histogram localization in specific regions of an image.

1.3.2 Face Detection

Another stage of the FR process involves finding the human face in the test image. The system can now detect a distinct face with a recognized dimension. An impressive objective that appears to have applications to real-time FR and realistic face tracking is face detection at frame rate [4]. It simply means that a system can recognize that a human face is present in a picture or a video. There are numerous uses for it, but facial recognition is just one of them. Cameras with autofocus can also use it. It can also be used to determine how many people have entered a specific place.

1.3.3 Feature Extraction

In this level, face features are obtained from test images. Direct use of these features for FR has a number of disadvantages, the first of which is that they often include over 1000 pixels per feature,

which is too much for an effective identification system. Second, facial characteristics can be distorted and captured in a variety of camera alignments, illuminations, and facial expressions. Numerous feature extraction techniques have been used such as Principal Component Analysis (PCA) [5], Discrete wavelet Transform (DWT) [6], Local Binary Pattern (LBP) [7], Discrete Cosine Transform (DCT) [8] and Singular Value Decomposition (SVD) etc. SVD is a robust feature extraction method because SVD faces are not influenced by variation of illumination conditions [9]. A crucial part of recognition is the taking out of appropriate characteristics from the face test images. It is therefore significant to select the correct feature extractor while developing a FR system with best accuracy rate.

1.3.4 Face Recognition

Once attaining the images of individual face, the final stage is to determine the characteristics of these images. Maintaining a facial database is essential if automatic recognition is to be achieved. Each person has multiple images taken, and the database is used to extract and save their features. Then, when an input sample image is provided, it starts face detection, face elements extraction, and its comparison with individual image class stored in the directory. We shall discuss various research and algorithm proposals that have been made to address this classification problem in later sections. Face identification is querying the system to identify the person in a given face image.

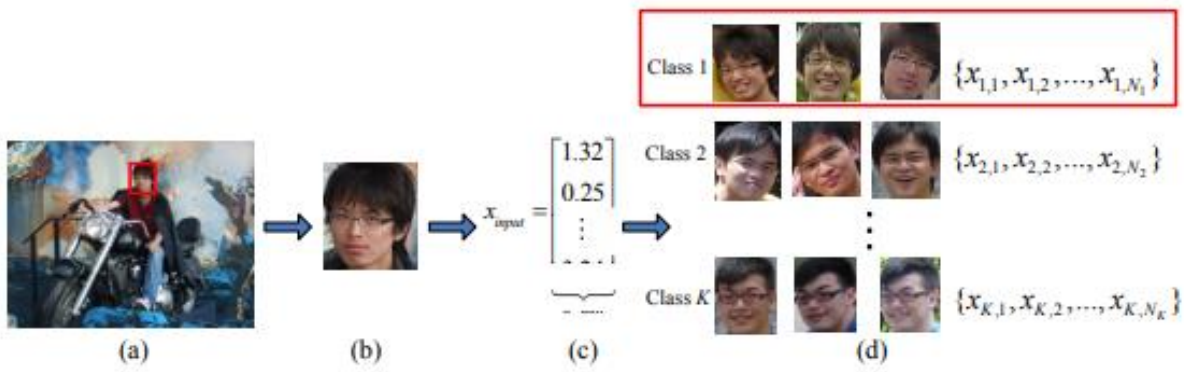


Fig 1.2: An example of general FR system working

1.3.5 Identification vs. Verification

For any one of the two purposes, identification or verification, biometric methods are used. This goal is closely linked to the choice and application of technology. Particularly, systems that are

intended for identification purposes will address the problem: "Who is the subject?". On the other hand, we want to check in the verification applications if the input image is the individual they affirm to be. It can be achieved via comparing the image observed features with a sample of features for the individual in the library. The question is answered by biometric approaches developed for authentication purposes: Is the subject who he says he is?

1.4 Problem Statement

There are two aspects to evaluate the FR algorithm.

- Recognition accuracy
- Computational complexity

There is always an interchange between accuracy rate and computational complexity rate. Most of the existing FR methods mainly emphasis on accuracy rate improvement. So main problems with these techniques are:

• Computational complexity of these algorithms is much higher.

• Recognition rate and computational complexity are not taken into account simultaneously to assess the effectiveness of algorithm.

• Untrained database test images are not rejected by state of the art FR systems.

1.5 Objective

The main objectives of the thesis are:

• To maximize classification accuracy and rejection rate.

• To minimize the computational cost.

• To employ an assessment method to evaluate the effectiveness of algorithm.

1.6 Applications

• **Use of Image database for inquiries**:

Examining image directories of police bookings, lost children, refugees, and certified drivers.

• **Identity verification:**

Banking, identification of newborns, electronic commerce, national IDs, passports, and employee IDs are just a few examples.

• **Surveillance:**

Facial recognition system can be integrated with IP Camera, and can be used in various surveillance fields.

1.7 Summary of Contribution:

- The proposed work offers acceptable recognition rate but considerably decreases computational cost.
- The two standard face image datasets; ORL and YALE are used for assessment and comparisons.
- Number of quantization levels have been reduced than existing techinque.
- Number of states have been reduced, results in low computational cost.
- An evaluation function is employed for the estimation of performance of proposed model via complexity and error rate at the same time.
- The end result authenticates that the recognition rate of the proposed model is comparable with the current methods for trained datasets.

1.8 Thesis Outline

There are six chapters in this thesis:

• **Chapter 1:** The chapter presents introduction, problem statement, objectives and the applications.

• **Chapter 2:** This chapter discusses literature review along with brief description of existing techniques.

• **Chapter 3:** In this chapter, Hidden Markov model (HMM) and its application in FR are explained in detail.

• **Chapter 4:** This chapter shows detailed description of proposed methodology.

• **Chapter 5:** In this chapter, experimentation and analysis of results are discussed in order to evaluate proposed methodology over previous proposals for FR systems.

• **Chapter 6:** This chapter concludes impact of proposed technique.

Chapter 2

LITERATURE REVIEW

This chapter will provide detailed information on the most relevant feature extraction and classification techniques used for FR. We start by having a look at some of the earlier methodologies that used fundamental techniques based on features. Afterthat, we review some advanced statistical and holistic approaches like PCA (Eigen faces), Independent component analysis (ICA), Linear Discriminant Analysis (LDA). Other methods discussed are Gabor wavelets, HMM and neural networks (NN).

2.1 Feature Extraction Techniques

Human face images are usually large and processing these large images is computationally expensive. Since all pixel values or feature coefficients may not contribute positively in recognition process. So, most informative feature values/coefficients are extracted from original data for reliable recognition. Some most common feature extraction techniques are discussed below.

2.1.1 Geometric method

In this method, features are retrieved by analyzing the image size and location relative to other relevant elements. An essential component direction and edges are first recognized, and then observation vectors are built from these edges and directions. Gradient analysis and the canny filter are often used in this approach. It is also possible to use feature blocks to transform the grayscale distribution of irrelevant components into a feature. As geometric features (location, orientation) of eyes, nose, lips and ears are different for each individual. In this technique, these geometric features are extracted [10]. Then image subtraction is performed between unknown/test image and known images. Difference between geometric features is calculated. Minimum distance classifier is used to classify face images i-e unknown face image will be classified to that person's face for which difference between geometric features is minimum.

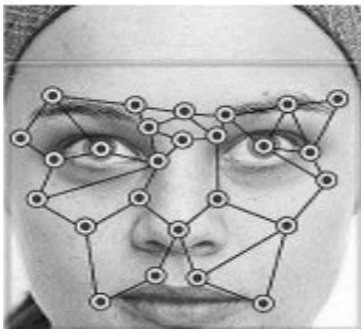


Fig 2.1: Geometric Representation of a person

2.1.2 Template Matching Technique

Using the suitable energy function, this approach allows for the extraction of certain facial features. Face features may be detected and described using deformable templates, as shown by Nate et al. [11]. In deformable templates, the feature of interest, such as an eye, is specified through a parameterized template. Energy functions are used to connect picture intensity peaks and troughs to the attributes of a given template. After that, the picture is compared to a template and deforms to find the best fit. The last parameter value is utilized as a descriptor. The first method uses an eye template to identify the eye in a picture. The eye templates and other overlapping parts of the facial picture are then determined to have a link with one another. The template has the most link with the area around the eyes [12].

2.1.3 Principal Component Analysis

The Eigen faces technique is discussed in this section. Decomposing face photos into a limited collection of typical feature images known as Eigen faces, they are utilized to represent both existing and new faces in this method of face recognition. Faces may be represented by their Eigen face components using PCA, which is an efficient and effective way to do so. It performs a covariance analysis between components in order to minimize data dimensionality. It is possible to discover correlations between samples or conditions when PCA is applied to a given dataset [13]. The Eigen faces that account for the greatest variation in the face database are used to represent each unique face.

2.1.4 Linear Discriminant Analysis

LDA is a unique method utilizes in pattern recognition, machine learning and statistics to identify a linear sequence of characteristics that distinguishes between objects of two or more classes. To avoid over-fitting and to minimize the computational costs, this technique provides a dataset onto a lower-dimensional space with stable class-reparability [14]. The derived sequence is adopted to reduce dimensionality before further classification.

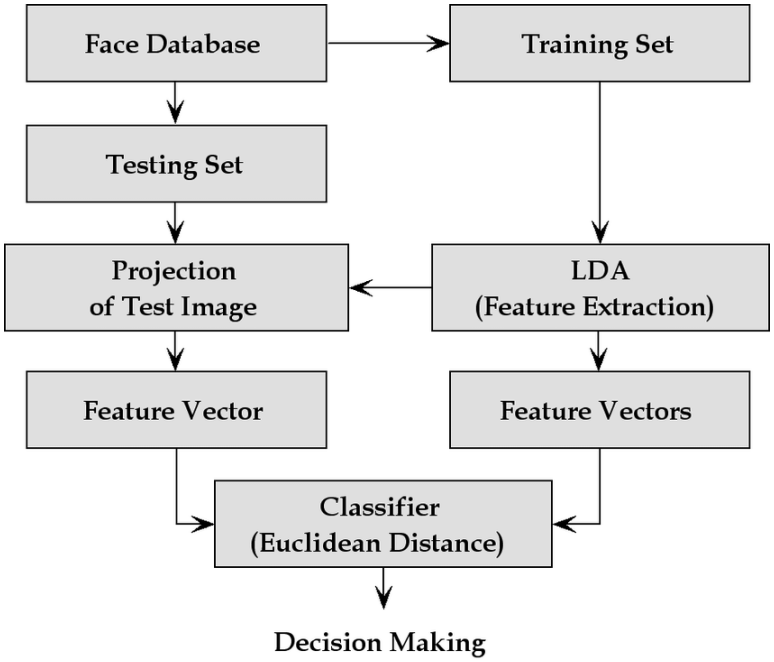


Fig 2.2: LDA based FR flow diagram

The line that best divides the points is known as LDA. This projection is best for low-dimensional image reconstruction. However, the technique does not use inter-class scattering between classes of face images of same person.

2.1.5 Independent Component Analysis

ICA is an approach used for extraction from a combination of statistically independent variables. The method is quite modern in the signal-processing world. The contribution coefficients of the two signals can be calculated using ICA, allowing us to isolate the two original signals from each other. In FR system, the significant information is in the high order relationships among the image picture elements. In addition to the second-order moments, ICA helps to classify the high order points of the input sample[15].

2.1.6 Gabor Wavelets

Wavelets offer a method for breaking complicated signals into sums of fundamental functions. They are comparable to Fourier decomposition techniques in this regard, although they differ significantly. In this way that they isolate frequencies rather than discrete occurrences of specific frequencies, Fourier functions are localized in frequency but not in space. In face recognition, the Gabor wavelets approach is frequently aided for face hunting and position evaluation. Whilst the Gabor wavelet technique used to represent an image gives both spatial information and spatial frequency structure [16]. It composites a property which allows it to distinguish between the characteristics of, spatial frequency selectivity, spatial localization and orientation, as illustrated in Fig 2.3. The abstraction of boundary and configure information works well with Gabor Wavelets, and it displays the faces in a compressed manner that is deeply relevant to facial elements-based approaches [17].

Fig 2.3: Original image and the reconstructed image with different number of wavelets

2.1.7 Elastic Bunch Graph Matching

For the elements classes with same structure, like faces in the same poses, EBGM is an development to elastic graph matching. The same kind of graph is used to represent each object of this class. Label graphs can be used to describe the same geometrical structure because it has distinct human faces [18]. As facial recognition is our requirement analyzing through multiple angles, the edges of the graphs directly connect to certain fiducial points, such as eyes, lips, the tip of the nose, and further features (Fig 2.7). To be able to match graphs from various postures index have to be developed to link respective nodes in the multiple graphs [19].

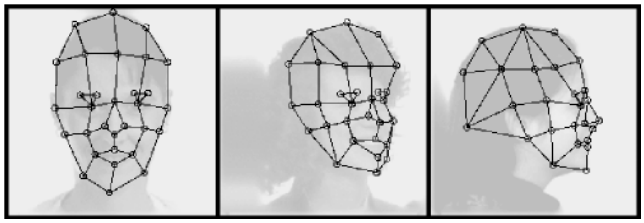


Fig 2.4: Graphs for faces in different views

2.2 Classification

In FR systems, several different classifiers have been used. Following are some of the mostly used classifier.

2.2.1 k-Nearest Neighbors

One of the most common machine learning algorithms as the classifier in their FR systems is the k-nearest neighbor algorithm. A new object is identified by majority-based mechanism of its nearest k neighbors and assigned to a class, where k is a usually small positive integer [20]. If k=1, in that case object is allocated to its closest neighbor class. The calculation measured to identify the new object is typically the Euclidian distance.

2.2.2 Neural Networks

Artificial neural networks (ANN) is a common procedure utilizes as a classifier in different pattern recognition functional devices inspirited by the structure of biological NN and are very useful in FR systems. A troop of artificial neurons is associated to each other in an artificial neural chain and forms a network of neurons resembling the biological NN. Researchers have proposed several different architectures for NN, such as multilayer perceptron, back propagation, radial basis function (RBF) NN, etc. Dynamic link architecture (DLA) are the most used techniques established on NN that dynamically group the neurons to higher order on the features extracted from face images and results in more effective FR against variations in the environment [21].

2.2.3 Multi-Layered Feed-Forward Networks

For classification purposes, a good method is the multi-layer perceptron neural network. Between its input and its output, it can approximate almost any regularity. The weights are modified by a prescribed back-propagation training technique [22]. The network nodes are linked together, and each of the links has a self-associated weight. An output of a node value is a weighted sum of all the nodes input values. We may change the effect of various input nodes by adjusting the differe nt weights of the input values. The input nodes will typically show the image pixel values of the test image to be recognized for face recognition. In the database, the output layer will refer to classes or individuals.

2.2.4 Radial Basis Function (RBF) Networks

Recently, RBF NN have gained extensive research interest in the neural network community. Due to local-tuned neurons, their learning speed is fast and they have a more flexible topology than other NN. The RBF network is a two-layer feed-forward network, with a concealed to output node monitored layer, a hidden to input unsupervised layer [23]. The impact of overlapping and locally tuned receptive fields is simulated by Gaussian functions for each of the hidden units.

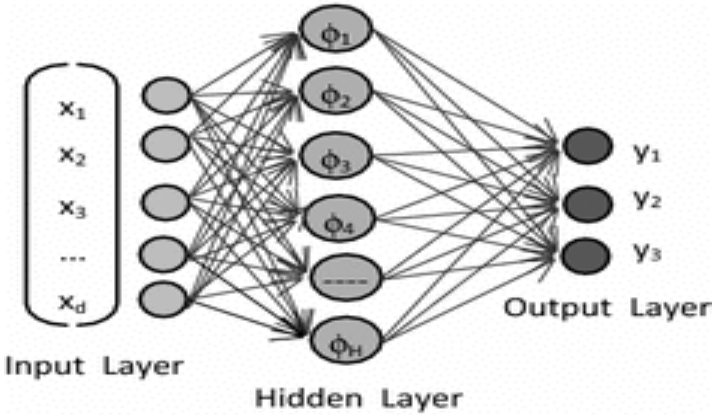


Fig 2.5: Three-layered architecture of RBFNN

2.2.5 Dynamic Link Architecture (DLA)

The image and all the models are represented in the dynamic link by layers of neurons identified by jets as local features. Jets are Gabor wavelet components vectors. If a model is identical to an image in the distribution of features, its initial connectivity matrix will link corresponding points with a high similarity of features [24]. By simultaneously enabling and creating correlations between corresponding regions in the image and model appear to match and synchronize.

2.2.6 Support Vector Machines

Simultaneously, SVM reduces the faults of classification and expand the geometric edge, so they are considered as maximum margin classifiers. An SVM generates a separate hyperplane that maximizes the margin between the data files in the feature space. To measure the margin, two parallel hyperplanes are constructed, one on each side of the separated one. These hyperplanes are forced against the two data sets, so that the hyperplane with the greatest outpost to the adjacent support vectors in couple classes achieves a good separation [25].

2.2.7 Hidden Markov Model

The HMM can effectively model the propagation of sequential data as a dual stochastic process. In face, handwritten, and speech recognition, HMM has been successfully used. The 2D implanted HMM recommended by Nefian [26] consists a set of super states, and a set of embedded states is associated with each super state.

Although if faces experience minor revolution in the image plane, these states still appear in the comparable array, exclusively. individual facial region will be defined to a state in a one-dimensional HMM from left to right. The purpose of the training phase is to optimize the parameters of the HMM to better explain the observations.

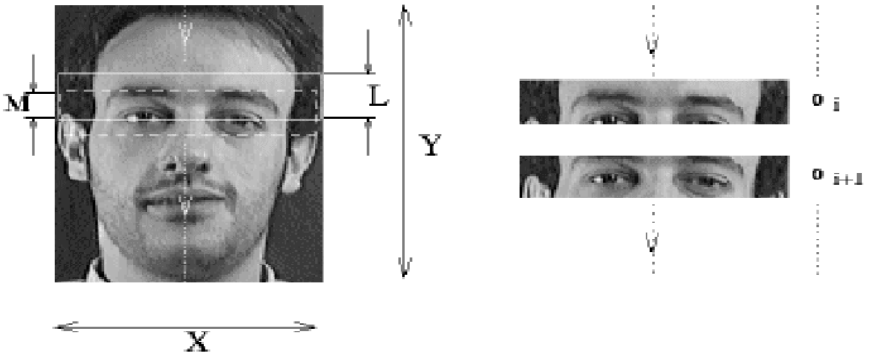


Fig 2.6: Image sampling technique for one-dimensional HMM

The model with the greatest probability shows the identity of unknown face. The number of features used to characterize the face besides computational complexity of the system are defined by the number of states. The computational complexity declines with decrease in recognition rate to 82.76% in two-state HMM [32]. On the contrary, by using five training image (ORL and Yale Database) in 7-state [31] , the expected HMM model achieves recognition rate of 97.78 % however, its computational complexity increases significantly. In MIT database 9 images were used in the training set in 5-state [30] and attained 90 % recognition. Its computational complexity is less than seven state HMM as it takes less sequential information, although its recognition rate reduced as compared to seven state. Decrease in the number of states leads to minimal memory occupation is evident but simulatneously accuracy compromises. Therefore, it is obvious that it is an interchange between computational cost and recognition accuracy. 3 state HMM is better as we achieved acceptable recognition rate with minimized computational complexity. The next chapter will explain HMM in more detail.

Chapter 3

Hidden Markov Model

3.1 Markov Model

A Markov chain is a collection of arbitrary variables $X_1, X_2, X_3, \ldots$ with the Markov property, specifically that the probability of any given state X_n only depends on its immediate previous state X_{n-1}. It can be shown mathematically as:

$$P(X_n = x \mid X_{n-1} = x_{n-1}, \ldots, X_1 = x_1) \tag{3.1}$$

The possible X_i values form a countable S set called the state space of chain. If the state space is finite and the time-homogeneous Markov chain, (i.e. the transition possibilities are constant in time), a matrix P= (p_{ij}) i, j∈S, called the transition matrix, represents the transition probability where p_{ij} can be written as:

$$p_{ij} = P(X_n = j \mid X_{n-1} = i) \tag{3.2}$$

The mechanism in which the result of a certain experiment could influence the result of forthcoming experiments is the Markov chain. It consists of certain known probabilities and finite number of states.

Let us see an example to understand Markov chain. Assume there are two types of weather,' sunny' and 'cloudy'. A news channel has to airing their weather report of the weather next week. To find out the weather for the upcoming week, the news channel hires a weather forecast company. There is 'sunny' weather conditions in that zone at present.

- The likelihood of the climate remaining ‘sunny’ the following week is 80%.
- Possibility of the climate varying from ‘sunny’ to ‘cloudy’ over a week is 20%
- Possibility of the climate conditions staying ‘cloudy’ the following week is 70%
- Possibility of the climate varying from ‘cloudy’ to ‘sunny’ over a week is 30%

For the entire week, though the climate is expected to be 'sunny', without making any transition calculations, one cannot be sure about the next week. The below matrix is given to define the transition:

$$P = \begin{bmatrix} 0.8 & 0.2 \\ 0.7 & 0.3 \end{bmatrix}$$

Transition matrix

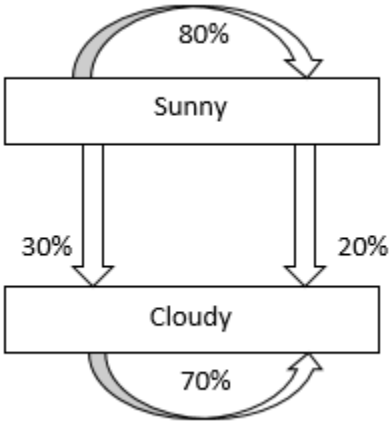


Fig 3.1: Transition diagram

Fig 3.1 shows the same changes and weather conditions as the matrix above. The following matrix multiplication can be used to perform the calculations:

Present State * Transition Matrix = Final State

S=Sunny; C= Cloudy

$$[100\% \quad 0\%] * \begin{bmatrix} 80\% & 20\% \\ 70\% & 30\% \end{bmatrix} = \begin{bmatrix} 80\% & 20\% \\ 70\% & 30\% \end{bmatrix}$$

It can be seen that there is an 80% possibility that it will be 'sunny' for the next whole week as well. However, around a 20% possibility that next week's climate might shift to 'cloudy'. This

process is called the Markov chain. We can also forecast the weather for further weeks if the transition matrix does not vary over time, using the same equation. The weather forecast in the two weeks' time can be calculated as follow:

$$[80\% \quad 20\%] * \begin{bmatrix} 80\% & 20\% \\ 70\% & 30\% \end{bmatrix} = [78\% \quad 22\%]$$

3.2 Hidden Markov Model

The HMM is a statistical model in which a Markov process with unknown parameters is supposed to be the system being modeled. To determine the hidden parameters from the observable parameters is the difficult part. The objective is to identify the most likely sequence of hidden states that may have formed a certain sequence of observed states given the model parameters. The state is directly observable to the observer in a regular Markov model, and so the probabilities of state transition are the only parameters. The state is not directly visible in a HMM. Rather, the observer sees an identifiable output. Each hidden state has a distribution of probability over the possible observable tokens, called emission probability. The HMM consists of a chain with a finite number of states, a probability matrix of state transition, and a distribution of initial state probability. Fig 3.2 illustrates a HMM.

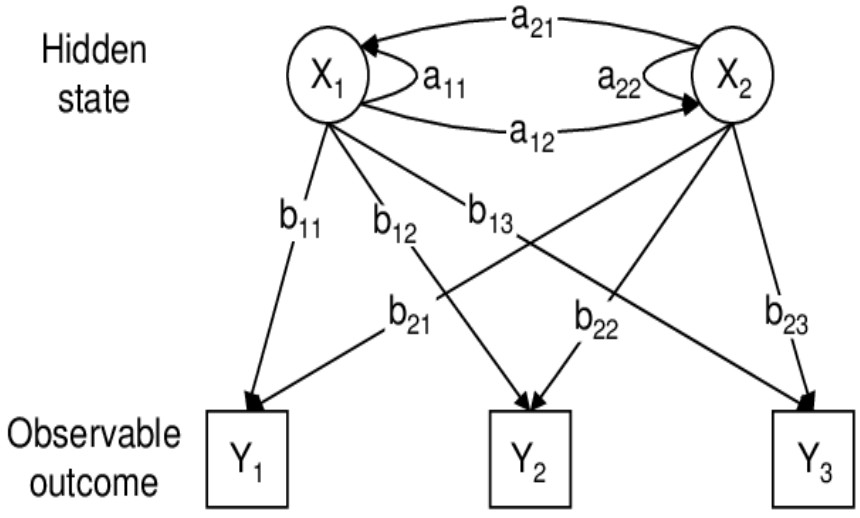


Fig 3.2: State diagram of two states HMM.

Although the states are invisible, each state produces observations that are drawn according to some distribution of probability (either discrete or continuous).

1. A set of N states and the individual states are denoted as, S= $\{S_1, S_2, \ldots, S_N\}$.
2. The initial state probability vector of size Nx1; = Π . This probability determines the state model from which the transition will initiate.
3. The state transition probability matrix of size NxN, A= $\{a_{ij}\}$, where it shows the probability of transition from s hidden state to any other state.

$$\sum_{j=1}^{N} a_{ij} = 1, \quad 1 \leq i \leq N \tag{3.3}$$

4. The emission probability matrix of size NxM; B= $\{b_{jk}\}$. This is the probability of a visible symbol emits from any state.

$$\sum_{j=1}^{N} b_j\,(k) = 1, \quad 1 \leq k \leq M \tag{3.4}$$

Hence, the classic HMM is uniquely defined by these parameters and written as

$$\lambda = [\,\mathbf{A}, \mathbf{B}, \Pi\,]$$

3.3 The Urn and Ball Model

Suppose that with colored balls, there are N large glass urns. There are *M* different colors of the balls, see Fig. 3.3.

The steps needed for generating an observation sequence are:

1. According to the initial state distribution π, select an initial state q_1= i. In the urn and ball model, the urn refers to a state.
2. Set *t* = 1 (i.e. clock, *t* = 1, 2. . . *T).*
3. According to the symbol probability distribution in state *i*, *bi* (o_t), pick a ball from the specified urn (state). The observation ot represents this colored ball. Return the ball to the urn. For example, in the first urn, the likelihood of a purple ball is 0.61, see Fig. 3.3.

4. Transit to a new state (urn) $qt+1 = j$ according to the state-transition probability distribution for state i, i.e. aij.
5. Set $t = t+1$; return to step 3 if $t < T$; otherwise, terminate the procedure.

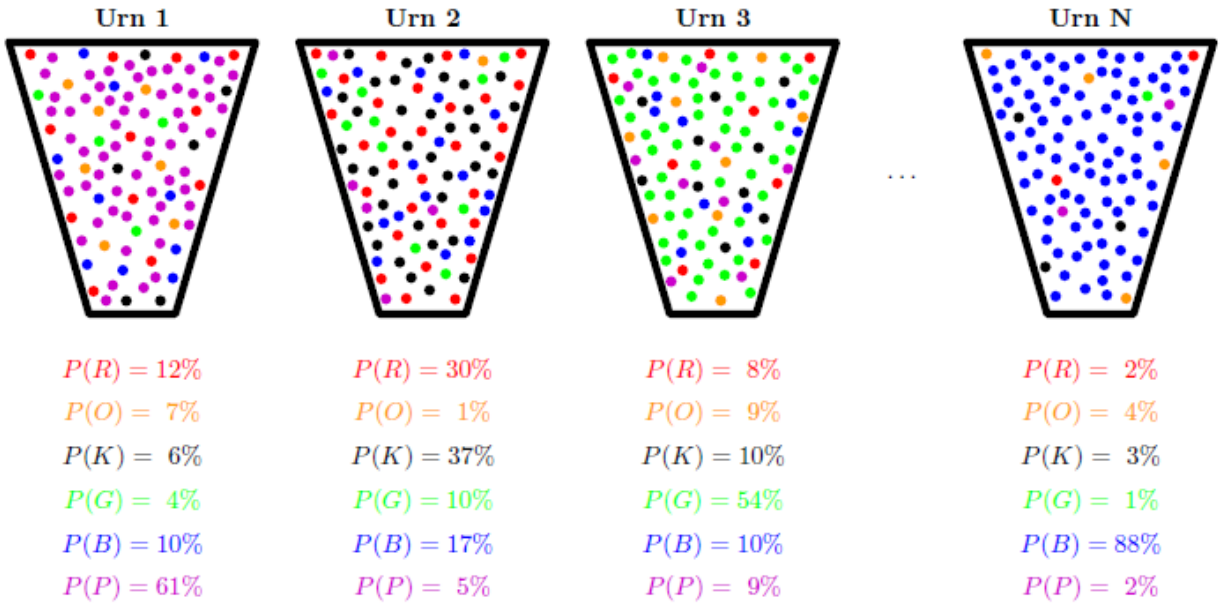


Fig 3.3: Urn and ball example with N urns and $M = 6$ different colors

When generating an observation sequence, these steps describe how the HMM behaves. It should be noted that balls of the same color can be found in the urns, and the difference between different urns is the way the set of colored balls is formed. An isolated observation of a ball of a specific color, therefore, does not immediately shows from which urn it is drawn. In addition, the connection between the example of the urn and ball and the HMM is that the urn refers to a state and the color refers to a feature vector (the observation).

3.4 Models of HMMs:

Depending on the nature of the state transition, HMMs have three different types of models:

1. **Ergodic Model:** Any state can be reached in a single movement from any other state in this model for a system with a finite number of N states. All the transition probabilities are

non-zero in a completely connected transition matrix. In Fig 3.4, the model is graphically represented.

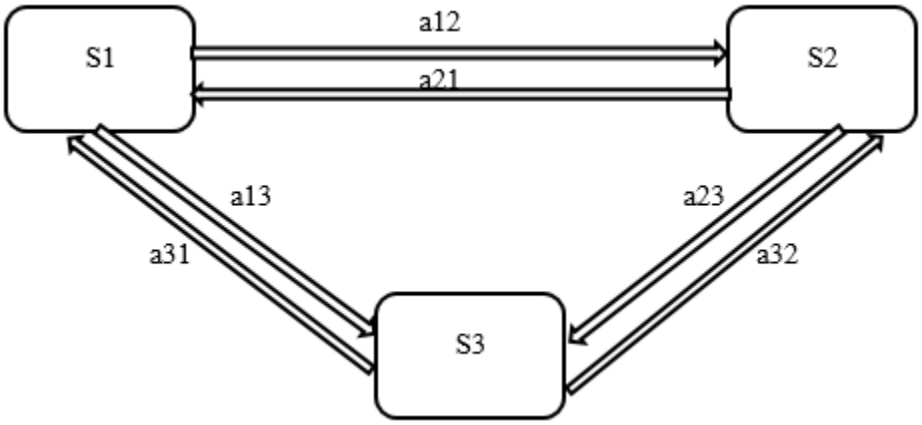


Fig 3.4: Ergodic Model

2. **Left-Right Model:** A state may switch to itself or the right of the state to it. All the transition probabilities have zero values in the transition matrix to the left of the state, and to the right of the states have non-zero values.

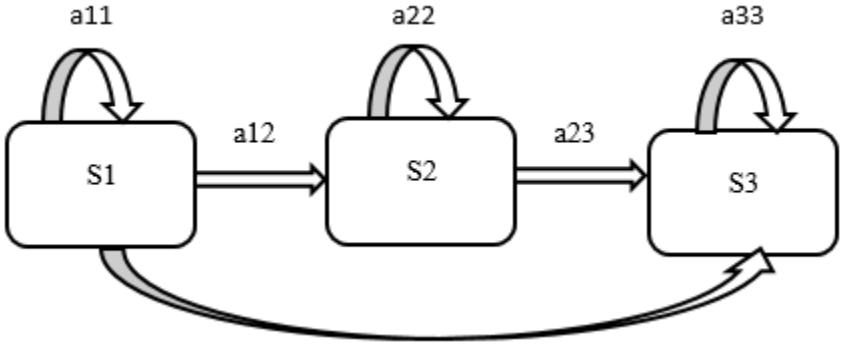


Fig 3.5: Left- Right Model

3. **Left-Right Banded Model:** A state may move to itself or the state next to it to the right side.

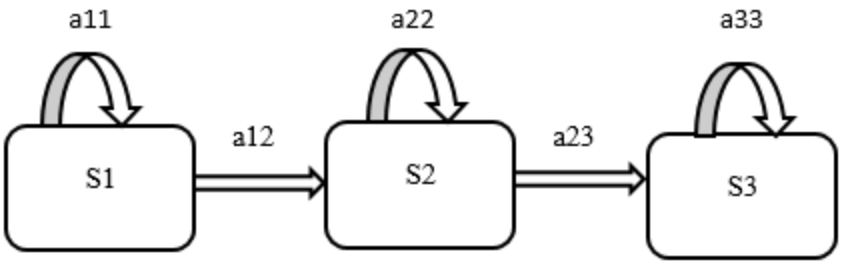


Fig 3.6: Left-Right Banded Model

One of most widely used topology in the gesture recognition is Left-Right Banded (LRB) model. In this model, the transition from a lower state to the same state or to the next higher state moves only forward but never backward. This topology is the most popular for the modelling process over time.

3.5 HMM for Face Recognition:

The goal of FR is to demonstrate supervised or unsupervised classifications. In the literature, different methods and applications have been discussed. Progressive face recognition, which can be seen as a specific case of statistical face recognition, is currently mainly approached by the use of the HMM. We begin with an image of a face that is to be recognized in order to use an HMM for face recognition. The observation sequence is obtained first by scanning the image and extracting the observation vectors by each block. Then, using the Forward algorithm, the likelihood of the observation sequence is found for each face given the HMM model. The model with the highest probability is taken, and the model shows the identity of the unknown face. ORL and Yale databases are used to verify the efficiency of the proposed algorithm in face recognition. ORL database consists of 400 face images of 40 persons with different face orientations and facial expressions [27]. Yale database contains 165 face images of 15 persons with varying illumination conditions and face behavior [28].

3.5.1 HMM-Learning Process:

In training process, first face images are mapped to an equivalent general structure of HMM. In general, structure only number of hidden states, visible symbols and transition of states are known as shown in Fig. 3.7. In our model facial regions are mapped by three states (S_1, S_2 and S_3) of

HMM and (q_1 and q_2) are visible symbols/features that are emitted by each state (facial region) during transitions. In learning process HMM parameters, i-e transition and emission probabilities are to be estimated using rough structure of model. Baum-Welch algorithm is used for this purpose.

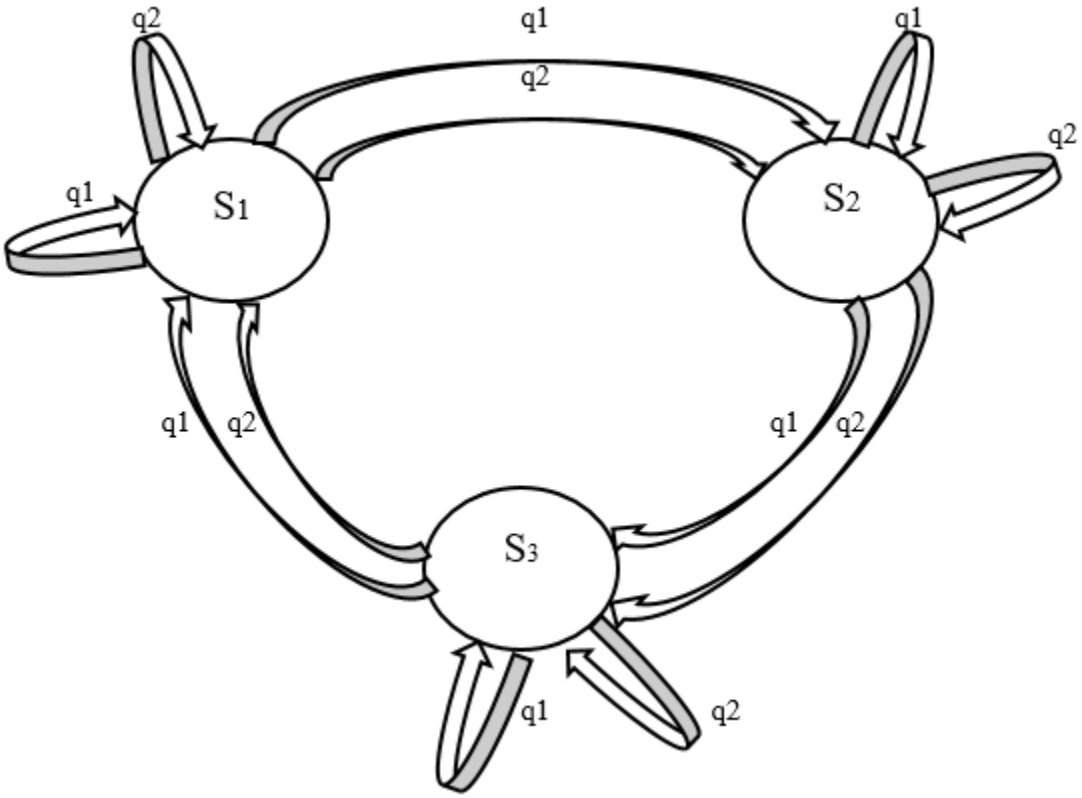


Fig 3.7: Structure of proposed model

Let say S_1 is the hidden state of HMM which represents the facial regions eyes and S_2 represents the facial regions nose and S_3 represents the facial regions mouth. q_1 and q_2 represent the image features let say these are two principal components of facial regions that are emitted during transition.

3.5.2 Baum Welch Algorithm:

It is a special case of the EM algorithm used to find the unknown parameters of a HMM. It makes use of the forward-backward algorithm to compute the statistics for the expectation step. Its purpose is to tune the parameters of the HMM, namely the state transition matrix **A**, the emission matrix **B**, and the initial state distribution $\boldsymbol{\pi}$, such that the model is maximally like the observed

data. There are a few phases for this algorithm, including the initial phase, the forward phase, the backward phase, and the update phase.

It is an iterative approach in which first of all model parameters are initialized and then expected counts of transitions are calculated using Eq. (3.5). Then model probabilities are re-estimated using Eq. (3.6). This process continues until the difference between probability values of two consecutive iterations is within a specified threshold.

$$C\ (s_i \xrightarrow{q_d} s_j\ ) = \sum_M P(S_M / \mathrm{q}^T)\ \mathrm{x}\ \mathrm{n}(s_i \xrightarrow{q_d} s_j\ ) \tag{3.5}$$

$$P\ \left(s_i \xrightarrow{q_d} s_j\right) = C\ \left(s_i \xrightarrow{q_d} s_j\right) \sum_{i=1}^{N} \sum_{k=1}^{T} C\ \left(s_i \xrightarrow{q_k} s_l\right) \tag{3.6}$$

Let say features/visible symbols $\{q_1, q_2, q_3\}$ are emitted by facial regions of a face image while transition from one facial region to other. Our task is to train HMM for this face image using its features. New transition and emission probabilities are calculated using these estimated probabilities as follows:

$$a_{ij} = \frac{\sum_k P(s_i \xrightarrow{q_k} s_j\ )}{\Sigma_k \Sigma_l P\ (s_i \xrightarrow{q_k} s_l\ )} \tag{3.7}$$

$$b_{jd}\ = \frac{\sum_l P\ (s_l \xrightarrow{q_d} s_j\ )}{\Sigma_k \Sigma_l P\ (s_l \xrightarrow{q_k} s_j\ )} \tag{3.8}$$

3.5.3 HMM Evaluation:

In evaluation problem observation sequence/feature coefficients of test face image and model θ are known parameters. We have to find out the probability of evaluation of observation sequence q^T using trained model θ.

$$\mathrm{P} = \mathrm{q}^T/\theta) = ? \tag{3.9}$$

To find out this probability we have to find out all possible sequence of hidden states/facial regions that may generate the given observation sequence. P (q^T/θ) is calculated by taking summation of probabilities of all these hidden state sequences.

$$\mathrm{P}(\frac{\mathrm{q}^T}{\theta}) = \sum_{r=1}^{rmax} \mathrm{P}(\mathrm{q}^T/{s_r}^T)\mathrm{P}({s_r}^T) \tag{3.10}$$

Where r_{max} represents the number of possible paths of hidden states through which model can make transitions while generating q^T. If N is the number of hidden states then r_{max} =N^T and ${s_r}^T$ means one of those possible hidden sequences S^T of length T that has generated q^T.

$$ {s_r}^T = \{S_1, S_2, S_3, \ldots, S_T\} \tag{3.11} $$

P (${s_r}^T$)is the probability of a particular hidden sequence that is given by the product of transition probabilities at different time instances. Similarly P ($q^T/{s_r}^T$)is given by the product of emission probabilities at different time instances.

$$ P({s_r}^T) = \prod_{t=1}^{T} P\left(\frac{s(t)}{s(t-1)}\right) \tag{3.12} $$

$$ P(q^T/{s_r}^T) = \prod_{t=1}^{T} P\left(\frac{q(t)}{s(t)}\right) \tag{3.13} $$

For time instant t=1, P $(q(t)/s(t)$ is the probability that model is at first state of a particular hidden sequence and it has generated first visible symbol of known observation sequence. By putting the values of P ($q^T/{s_r}^T$) and P (${s_r}^T$) in Eq. (3.10) we get

$$ P(\frac{q^T}{\theta}) = \sum_{r=1}^{rmax} P(\left(\frac{q(t)}{s_r(t)}\right) \prod_{t=1}^{T} P\left(\frac{s_r(t)}{s_r(t-1)}\right) \tag{3.14} $$

For an HMM with N hidden states and an observation sequence of T observations, there are N^T possible hidden sequences. For real tasks, where N and T are both large, N^T is a very large number, so we cannot compute the total observation likelihood by computing a separate observation likelihood for each hidden state sequence and then summing them. Complexity of Eq. (3.14) is O (N^TT). Instead of using such an extremely exponential algorithm, we use an efficient O (N^2T) algorithm called the forward algorithm.

3.5.4 Forward Algorithm

The forward algorithm is a kind of dynamic programming algorithm, that is, an algorithm that uses a table to store intermediate values as it builds up the probability of the observation sequence. The forward algorithm computes the observation probability by summing over the probabilities of all possible hidden state paths that could generate the observation sequence, but it does so efficiently by implicitly folding each of these paths into a single forward trellis. We compute the forward

probability $\alpha_t(j)$by summing over the extensions of all the paths that lead to the current cell. For a given state q_j at time t, the value $\alpha_t(j)$ is computed as

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{(t-1)}(i)\alpha_{ij}\ b_j(q_t) \tag{3.15}$$

The three factors that are multiplied in Eq. (3.15) in extending, the previous paths to compute the forward probability at time t are

$\alpha_{(t-1)}(i)$	The previous forward path probability from the previous time step
α_{ij}	The transition probability from previous state s_i to current state s_j
$b_j(q_t)$	the state observation likelihood of the observation symbol q_t given The current state j

1. Initialization:

$t \leftarrow 0, \alpha_{ij}, b_{jk}, \alpha_j(0)$

2. Recursion:

for $t \leftarrow t+1$

$\alpha_j(t) = (\sum_{i=1}^{N} \alpha_i(t-1)\alpha_{ij})\ b_{jk}q(t)$

3. Termination:

until t=T

Return $P(q^T/\theta) = \alpha_0(T)$

$\alpha_0(T)$ represents the probability of machine being at its final state S_0 after generating all "T" visible symbols. This is actually a probability that HMM model θ has generated observation sequence.

Example: Given an HMM λ = (A, B) and an observation sequence O, determine the likelihood P (O|λ).

The computation of the forward probability for our ice-cream observation 3 1 3 from one possible hidden state sequence hot hot cold is shown in Eq. 3.16. Fig 3.8 shows a graphic representation of this computation.

$$P(3\ 1\ 3|hot\ hot\ cold) = P(3|hot)\ \text{x}\ P(1|\text{hot})\ \text{x}\ P(3|\text{cold}) \tag{3.16}$$

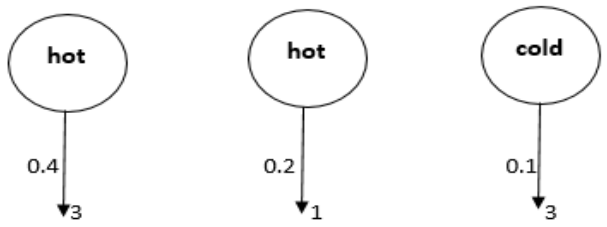


Fig 3.8: Observation likelihood for the ice-cream events hot hot cold.

The joint probability is computed for ice-cream observation 3 1 3 and one possible hidden state sequence hot hot cold and is shown in Eq. 3.17.

$$P(3\ 1\ 3, hot\ hot\ cold)$$
$$= P(hot|start) \text{x} P(hot|hot) \text{x} P(cold|hot) \text{x} P(3|hot) \text{ x P(1|hot) x P(3|cold)} \quad (3.17)$$

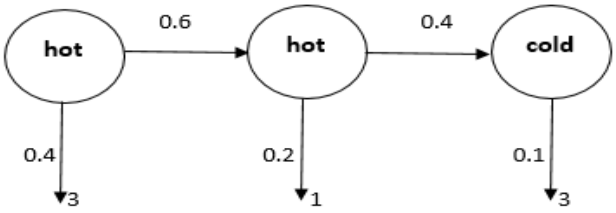


Fig 3.9: Joint probability of the ice-cream events hot hot cold

A trellis example for calculating the likelihood of 3 1 3 given the hidden state sequence hot hot cold is shown in Fig. 3.10. The probability of being in state j after observing the first t observations is represented by each cell of the forward algorithm trellis and is written as $\alpha_t(j)$. The value of each cell $\alpha_t(j)$ is determined by adding up the probability of all possible paths that could take us to that cell. The following probability is expressed formally in each cell:

$$\alpha_t(j) = P(o_1, o_2, \ldots, o_t, q_t = j|\lambda) \quad (3.18)$$

Here, $q_t = j$ means "the t^{th} state in the sequence of states is state j". We compute this probability $\alpha_t(j)$ by summing over the extensions of all the paths that lead to the current cell. For a given state q_j at time t, the value $\alpha_t(j)$ is computed from Eq. 3.15.

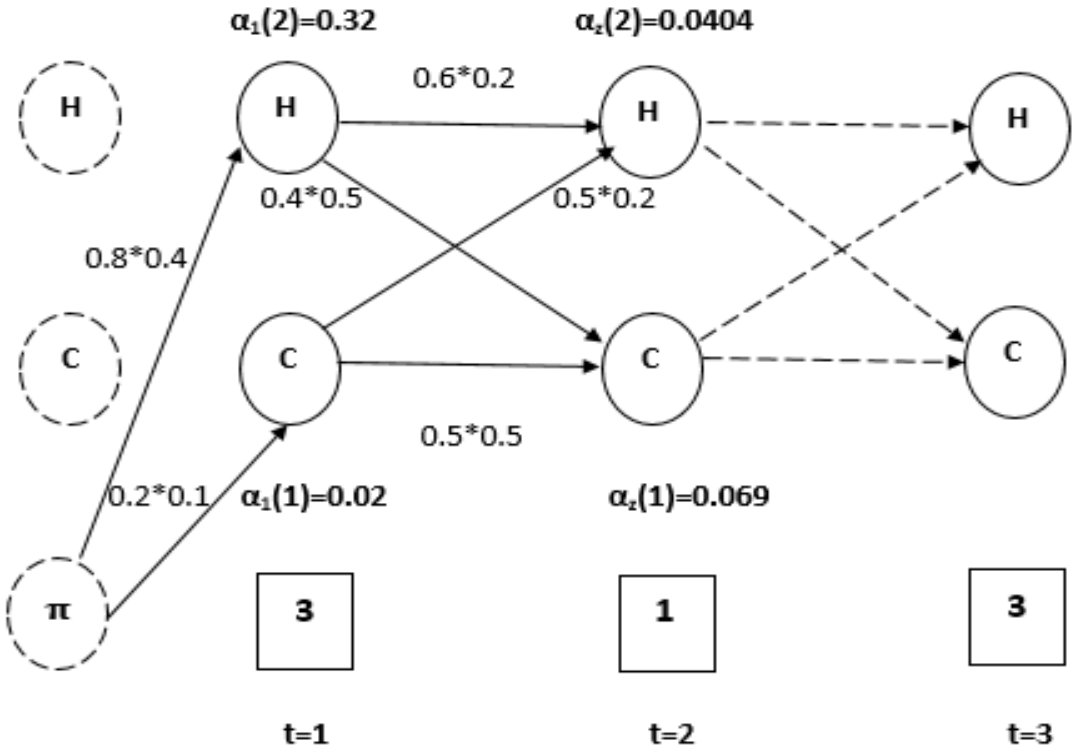


Fig 3.10: The forward trellis for observation likelihood. Circles and squares are showing Hidden states and observations respectively.

The figure demonstrates the computation of $\alpha_t(j)$ for two states at two time steps. In each cell, the computation follows Eq. 3.15. The resultant probability shown in each cell is calculted using Eq. 3.18.

Chapter 4

DESIGN OF A FACE RECOGNITION SYSTEM

In five HMM states, the image was split into five facial sector; forehead, eyes, hair, nose and mouth [30]. In 7 states HMM, the two other states of the face; chin and eyebrows were included [31]. The two-state HMM depends upon two distinct states of HMM [32]. The four regions have been eradicated from seven-state HMM to change it to three-state, which minimizes the computational complexity that is dependent upon states used in HMM. Moreover, in preprocessing maximum noise filtration has been achieved that increases recognition accuracy as compared with [32]. The method that we used is left-to-right 3-state HMM, where the states are used to represent eyes, nose and mouth are shown in Fig 4.1. In face recognition, feature coefficients of different facial states are visible symbols of HMM. This sequence of visible symbols is used to determine the evaluation probability that is to which person's facial regions this observation sequence may belong.

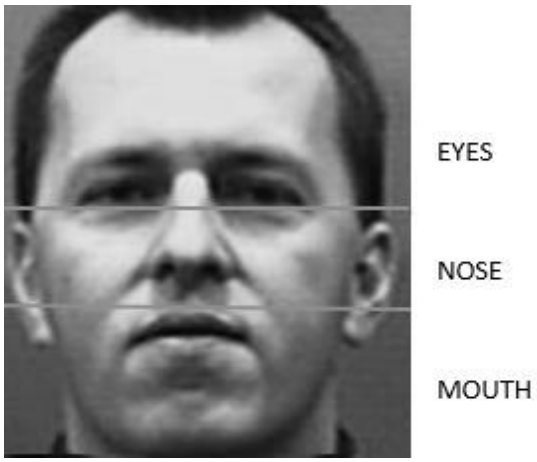


Fig 4.1: Three state HMM for human face image

4.1 Face Recognition System

FR system is mainly consists of two steps:

- Feature extraction
- Classification

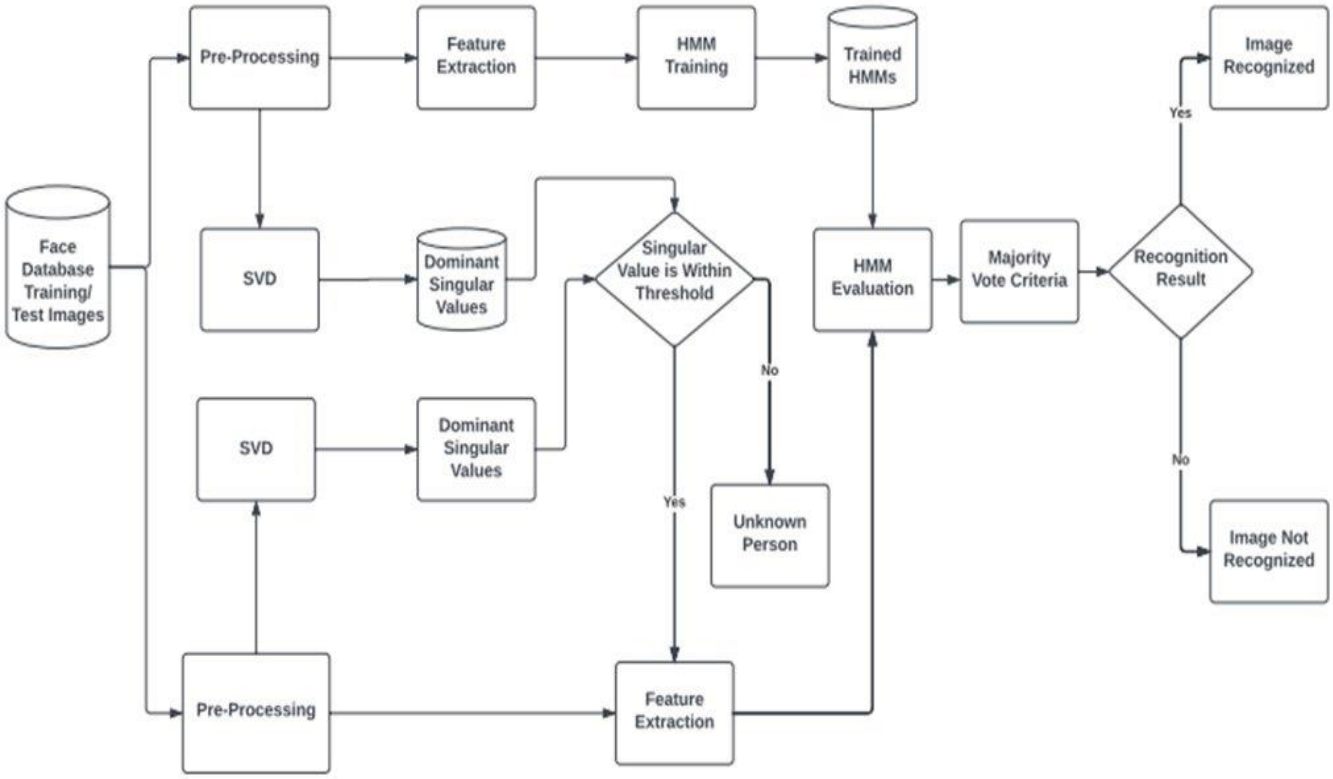


Fig 4.2: Proposed model Scheme

Before extracting different features, image preprocessing is involved that is consists of applying filtering and dimensionality reduction technique on images. Preprocessing is the initial stage in most of the FR systems and many other computer vision applications. Noise, poor lighting or unsuitable color might suffer from the input photos captured by video cameras. It suppressed undesired distortions and enhanced images characteristics.

4.1.1 Image filtering:

Image filtering is used for effective recognition in a FR system. It reduces noise in images. For improving, the performance of the system-preprocessing step is playing the vital role. In this section, CLAHE technique is enforced initially to the original images for further refinement in contrast [2]. Salt and pepper noise was included in the images that combined with the original noise of the pre-processed images. The integrated noise are then moved out with the help of two consecutive filters named Median and Morphology filters. These filters helped in maximum noise filtration [3]. The images obtained are converted into grayscale for further processing.

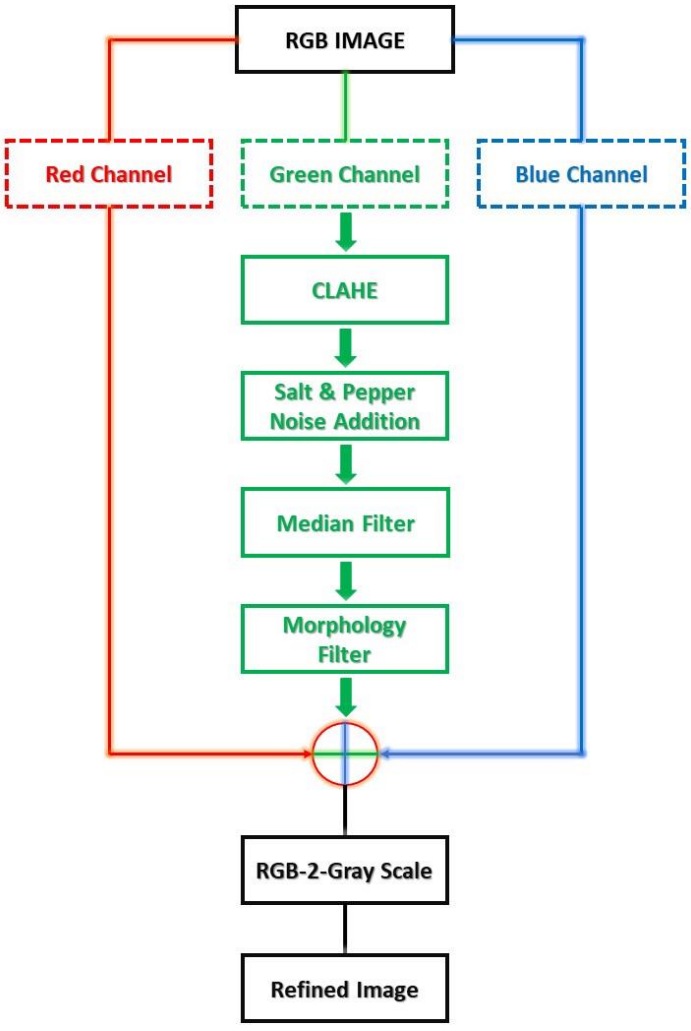


Fig 4.3: Image filtering

4.1.2 Data compression:

To process images in their original dimensions is computationally expensive so Haar wavelet is employed to reduce the dimensions of images. To calculate the DWT for a two-dimensional image, the actual real image is passed in the directions using low and high pass filters, as shown in Figure 4.4. Therefore, the four generated sub band images are LL, HL,LH and HH and contain the horizontal, diagonal and vertical approximation illumination of the image. LL is the coefficient of approximation that contains the maximum information about the image and is used for post-processing.

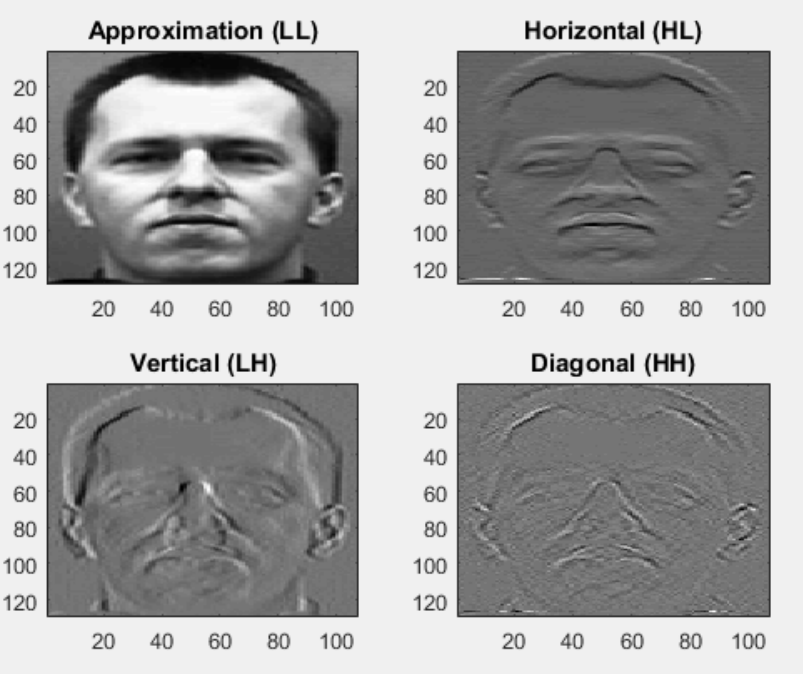


Fig 4.4: Haar Wavelet

4.1.3 Feature Extraction:

Image blocks of L rows are extracted from a face image of W by H dimensions (width x height) and use to form the observation sequence/ feature vectors (Fig 4.5). In block extraction, we get 1-D observation vector of features so that HMM may process these features. Adjacent blocks have overlap of 75% between two consecutive blocks by P rows. Instead of using the all pixel intensity values for each LxW image block, we form an observation vector from the coefficients from PCA.

The eigenvectors of the covariance matrix of all input images are obtained. Eq. (4.1) is used to obtain number of blocks for a specific image.

$$N = \frac{H - L}{L - P} + 1 \tag{4.1}$$

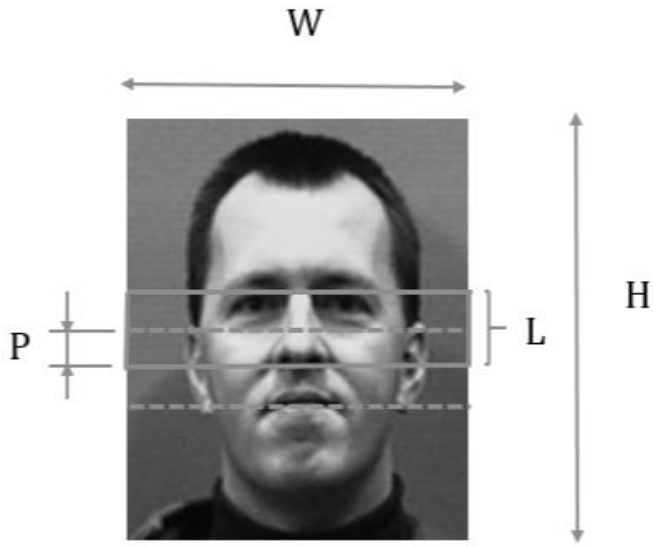


Fig 4.5: Sequence of overlapping blocks

Where N=50 is total extracted blocks using L=4, H=53 and P=3 that demonstrate elevation of image ensuring reducing dimensions, height of sampling window and overlap diameter correspondingly. We got 50 blocks for each image of size 53x44. Applying feature extraction techniques on high-dimensional image blocks without transforming them into low dimensions is computationally costly. Instead of using (LxW=4x53) 212 pixel values of each block, we computed SVD coefficients and used them as features. Let M is one of the image block, following MATLAB command is used to compute SVD:

$$[\mathrm{U}, \Sigma, \mathrm{V}] = \mathrm{svd}\big(\mathrm{double}\,(\mathrm{M})\big) \tag{4.2}$$

Where **Σ** is a transverse matrix that consist of singular values of the test image, **U** and **V** are symmetrical matrices consisting orthonormal irrational vectors. Singular values that are diagonal elements of Σ are selected as features. We used three features ($\sigma_{11}, \sigma_{22}, U_{11}$) as associating each

block ; first two matrix constants of **Σ** and first constant of matrix **U**. Consequently,a separate block is categorized by 3 different coefficients. These three coefficients are used for proposed HMM.

4.1.4 Quantization:

To get discrete values for each block, a quantization procedure is used as mentioned below:

- Let X is the vector coefficient and L is the vector having levels of quantization in which every component in L is related to the equivalent component in X.
- Afterthat, an evaluation is employed between two successive X vectors to specify the maximum (X_{max}) and minimum (X_{min}) values , thus the variation among the resultant values is computed using the equation mentioned below:

$$\Delta = \frac{X_{max} - X_{min}}{D} \tag{4.3}$$

After that, a vector Q is acquired to swap vector X:

$$Q = \frac{X - X_{min}}{\Delta} \tag{4.4}$$

4.1.5 HMM Training:

For individual class in the database, a distinctive model is used for training of ten images with Baum Welch algorithm [29], in which the method contains left to right three states HMM having separate densities. Figure. 4.6 demonstrates a diagram of state transition for suggested model.

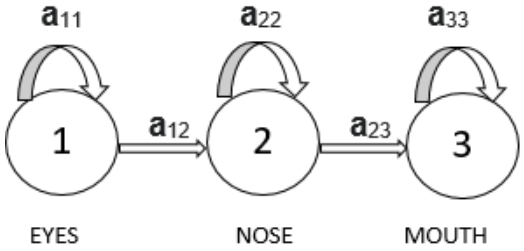


Fig. 4.6: Left to right state diagram of three states HMM

Normalized feature vectors obtained after quantization of block features are used as observation sequence for training of HMM. HMM training process for a given observation sequence of training images is depicted in Fig. 4.7. It is clear from the flow diagram that two major steps are involved in training of HMM. We have discussed how we got normalized observation sequence from an

image. Now we will see how to train HMM for FR using these observation sequences. In training process of HMM, Baum welch algorithm. In this algorithm, following steps are involved:

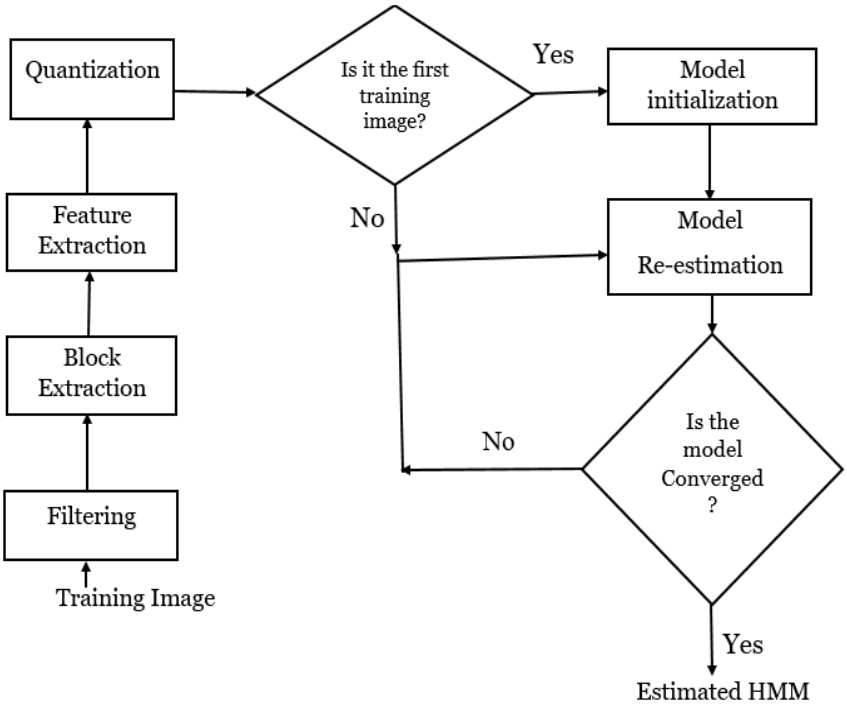


Fig. 4.7. HMM Training for face recognition

4.1.6 Model Initialization:

In this step, model parameters (A, B, π) are initialized. In proposed three states, HMM desired results are obtained using following initial parameters.

$$a_{i,i} = 0.5 \qquad 1 \leq i \leq 2$$
$$a_{i,i+1} = 0.5$$
$$a_{3,3} = 1$$
$$\pi = [1\ 0\ 0]$$
$$B = \frac{1}{M} Ones(N, M)$$

In the above equation, M represents all likely observation sequence acquired from process of quantization and N represents the number of states (in this method N is equals to 3). $a_{i,i}$ and $a_{i,i+1}$ are transition probabilities. π is initial probability vector and B is emission probability matrix.

Number of Hidden States (N) = 3

Number of Visible Symbols (M) = 1260

Initial Probability π = 1, for hidden state S_1

Where S_1 represents the forehead. HMM is initialized using following parameters:

.

$$A = \begin{array}{c} \\ S1 \\ S2 \\ S3 \end{array} \begin{array}{c} \begin{array}{ccc} S1 & S2 & S3 \end{array} \\ \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

$$B = \begin{array}{c} \\ S1 \\ S2 \\ S3 \end{array} \begin{array}{c} \begin{array}{cccc} q1 & q2 & \dots & q1260 \end{array} \\ \begin{bmatrix} \frac{1}{1260} & \frac{1}{1260} & \dots & \frac{1}{1260} \\ \frac{1}{1260} & \frac{1}{1260} & \dots & \frac{1}{1260} \\ \frac{1}{1260} & \frac{1}{1260} & \dots & \frac{1}{1260} \end{bmatrix} \end{array}$$

A and **B** represent transition and emission probability matrices of proposed model correspondingly. S_1, S_2 and S_3 are hidden states of three state HMM which represent eyes, nose and mouth of face image respectively.q_1, $q_2,\ldots, q_{1260}$ represent the visible symbols of HMM which are basically the features of face images extracted using PCA.

4.1.7 Model Re-estimation:

In first iteration of Baum-welch algorithm, these initial parameters are used to compute new estimated parameters using Eq. (4.5) and (4.6). We used a threshold value of 0.09. If the difference between estimated parameters and initial parameters is not within 0.09 then algorithm is re-iterated to re-estimate the model parameters. Estimated parameters of current iteration are used for next iteration. This procedures follows until model convergence or maximal number of iterations are achieved which are set to five in propose technique.

$$C\left(s_i \xrightarrow{q_d} s_j\right) = \sum_{M} P(S_M / \mathrm{q}^T) \text{ x n} \left(s_i \xrightarrow{q_d} s_j\right) \tag{4.5}$$

$$P\left(s_i \xrightarrow{q_d} s_j\right) = C\left(s_i \xrightarrow{q_d} s_j\right) \sum_{i=1}^{N} \sum_{k=1}^{T} C\left(s_i \xrightarrow{q_k} s_l\right) \tag{4.6}$$

The proposed three states HMM with transition probabilities and possible observation sequence is shown in Fig. 4.8 and parameters re-estimation process in depicted in Tab. 4.1 and 4.2 using only two visible symbols of three state with emission probability 0.5 for each visible symbol. This process continues for three states of HMM and all visible symbols.

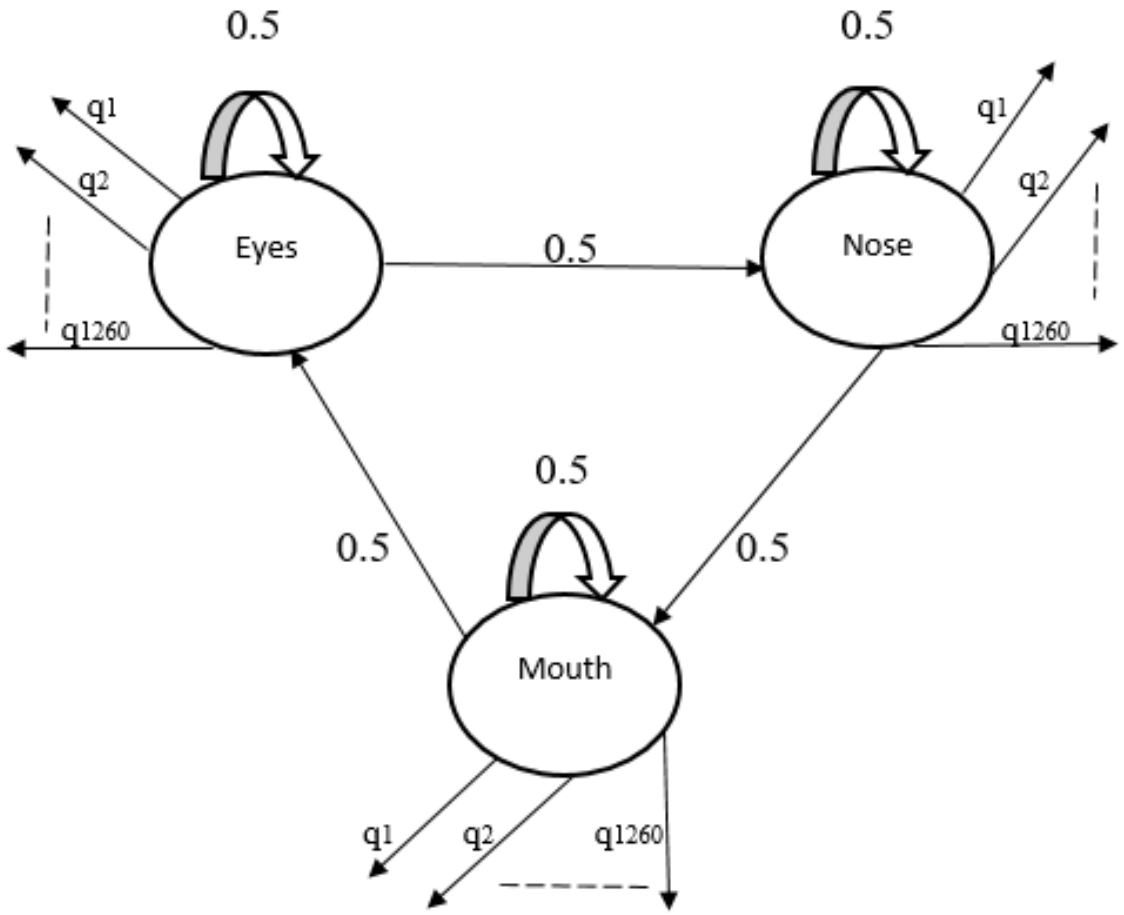


Fig. 4.8: Three states HMM for three facial regions

4.1.8 Classification:

In result of training process, each person is given a separate HMM. For testing of a face image, we are using Forward algorithm. Two different observation sequences are used to represent each test image that contain coefficients of eigenvalues and eigenvectors that are little bit different with each other. Probability of each of these two observation sequences of a trial image is to be calculated against each trained HMM. The probability of these observation sequences is checked contrary each HMM (acquired after training process).

Table 4.1: Parameters re-estimation using Baum-Welch algorithm from state S_1 to S_2

Hidden Sequences for given Observation Sequence				Path Probabilities	Values of Expected Counts for different transitions					
$E \rightarrow q_1$	$q_1 \rightarrow q_2$	$q_2 \rightarrow q_1$	$q_1 \rightarrow E$	P_{path}	$S_1 \rightarrow S_1$ (q_1)	$S_1 \rightarrow S_1$ (q_2)	$S_1 \rightarrow S_2$ (q_1)	$S_1 \rightarrow S_2$ (q_2)	$S_2 \rightarrow S_2$ (q_1)	$S_2 \rightarrow S_2$ (q_2)
S_1	S_2	S_2	S_2	**0.01758**	**0**	**0**	**0.01758**	**0**	**0.01758**	**0.01758**
S_1	S_1	S_1	S_2	**0.01758**	**0.01758**	**0.01758**	**0.01758**	**0**	**0**	**0**
S_1	S_1	S_2	S_2	**0.01758**	**0.01758**	**0**	**0**	**0.01758**	**0.01758**	**0**
S_1	S_1	S_1	S_1	**0.05273**	**0.10546**	**0.05273**	**0**	**0**	**0**	**0**
Total →				**0.10547**	**0.14062**	**0.07031**	**0.03516**	**0.01758**	**0.03516**	**0.01758**
New Probability Values using Eq. (4.6)					**0.53332**	**0.26666**	**0.13335**	**0.06667**	**0.66667**	**0.33333**

Table 4.2: Parameters re-estimation using Baulm-Welch algorithm for state S_2 to S_3

Hidden Sequences for given Observation Sequence				Path Probabilities	Values of Expected Counts for different transitions					
$E \rightarrow q_1$	$q_1 \rightarrow q_2$	$q_2 \rightarrow q_1$	$q_1 \rightarrow E$	P_{path}	$S_2 \rightarrow S_2$ (q_1)	$S_2 \rightarrow S_2$ (q_2)	$S_2 \rightarrow S_3$ (q_1)	$S_2 \rightarrow S_3$ (q_2)	$S_3 \rightarrow S_3$ (q_1)	$S_3 \rightarrow S_3$ (q_2)
S_2	S_3	S_3	S_3	**0.01875**	**0**	**0**	**0.01875**	**0**	**0.01875**	**0.01875**
S_2	S_2	S_2	S_3	**0.01354**	**0.01354**	**0.01354**	**0.01354**	**0**	**0**	**0**
S_2	S_2	S_3	S_3	**0.01593**	**0.01593**	**0**	**0**	**0.01593**	**0.01593**	**0**
S_2	S_2	S_2	S_2	**0.0767**	**0.1534**	**0.0767**	**0**	**0**	**0**	**0**
Total →				**0.1249**	**0.1828**	**0.0902**	**0.0322**	**0.01593**	**0.0347**	**0.01875**
New Probability Values using Eq. (4.6)					**0.5692**	**0.2809**	**0.1002**	**0.0496**	**0.6498**	**0.3511**

Probability of each of these two observation sequences of a test image is to be calculated for each competent HMM.

$$P\left(\frac{O_{seq}}{\theta_n}\right) = ? \qquad \text{where} \quad u = 1, 2 \qquad (4.7)$$

Majority vote rule is used to classify a test image based on the evaluated probabilities for identified observation sequences as given in Eq. (4.8).

$$I_t = \begin{cases} I_K & \text{if } P(Oseq_u)/\theta k = maxnP(O_s/\theta n) \\ Unknown & \text{otherwise} \end{cases} \quad (4.8)$$

Where P (O_s/θ_k) shows evaluation probability of the observation sequence O_s. Test image I_t is identified as k^{th} database image if evaluation probability for that observation sequences is maximum against trained HMM θ_k. This process is followed for the test images in the inventory. Recognition rate is calculated using Eq. (4.9).

$$Recognition\ Rate = \frac{No.\ of\ correctly\ classify\ images}{Total\ No.\ of\ test\ images} \text{ x}100 \quad (4.9)$$

Face images that do not belong to the trained database are successfully rejected by comparing the largest eigenvalue of each test image with the largest eigenvalues of trained images. If difference between largest eigenvalues of the test and trained images is within threshold then these test images undergo feature extraction and are classified using forward algorithm otherwise rejected by the model. Face rejection process using eigenvalue threshold is shown in Fig. 4.2.

4.2 Analysis of Computational Complexity

4.2.1 Computational Complexity of HMM Training

Complexity of HMM training process is directly dependent upon the complexity of Baulm-Welch procedure. Baulm-Welch procedure has time complexity per iteration of O (N_s^2T). Ns represents number of model states and T is observation sequence length. The following technique is used in observation sequence length for calculation:

$$O_{seq} = N_I \times N_B \times N_K \quad (4.10)$$

Where,
N_I : the number of images,
N_B : number of blocks per training image
N_K : number of feature coefficients per block.

Overall complexity of HMM. Training complexity for all individuals in the face database is:

$$\text{Complexity} = N_p \times N_{it} \times N_s^2 O_{seq} \quad (4.11)$$

where N_p is total number of persons and N_{it} is number of iterations of Baulm-Welch algorithm.

4.2.2 Complexity of HMM Evaluation

Forward algorithm is implemented for evaluation of proposed method. Computational complexity of forward algorithm is O(N^2K).

$$\text{Complexity of HMM evaluation} = N_p \times N_s^2 O_{seq} \quad (4.12)$$

This means that complexity of recognition process mainly depends upon the number of states of HMM as it alter quadratically with the number of model's states.

4.3 Figure of merit:

4.3.1 Confusion matrix:

The common method for dealing with FR system evaluation focuses on the idea of positive and negative detection as an actual data. The confusion matrix for the Yale database that consists of 15 classes and 165 face images, is shown in Table 4.3. The system will identify a portion of face images as true positives, while a portion of non-face images will be mistakenly identified as faces as false positives. The meaning of "true positive" in this context is the same as that of "detection rate" and "recall". False negatives refer to failing to identify people even while their image is in the database, while false positives refer to incorrectly matching the individuals with photos in the database.

4.3.2 Receiver operating characteristics

To organize and evaluate an overall performance of the system, a graph called the receiver operating characteristics curve (ROC) is used. Usually, these graphs are employed in research on data processing and aritficial intelligence. The transition between the false positive rate (FPR) and true positive rate (TPR) is shown graphically. While FPR denotes wrongly identified images along the x-axis, TPR denotes correctly classified images plotted on the y-axis.

Table 4.3: Confusion matrix for classes of Yale database

	Actual Classes															
Predicted Classes	Classes	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
	C1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	C2	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0
	C3	0	0	5	0	0	1	0	0	0	0	0	0	0	0	0
	C4	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0
	C5	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0
	C6	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0
	C7	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0
	C8	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
	C9	0	0	0	0	0	0	1	0	6	0	0	0	0	1	0
	C10	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0
	C11	0	0	1	0	0	0	0	0	0	0	6	0	0	0	0
	C12	1	0	0	0	0	0	0	0	0	0	0	6	0	0	0
	C13	0	0	0	0	0	0	0	1	0	0	0	0	6	0	0
	C14	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0
	C15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6

Chapter 5

RESULTS AND DISCUSSION

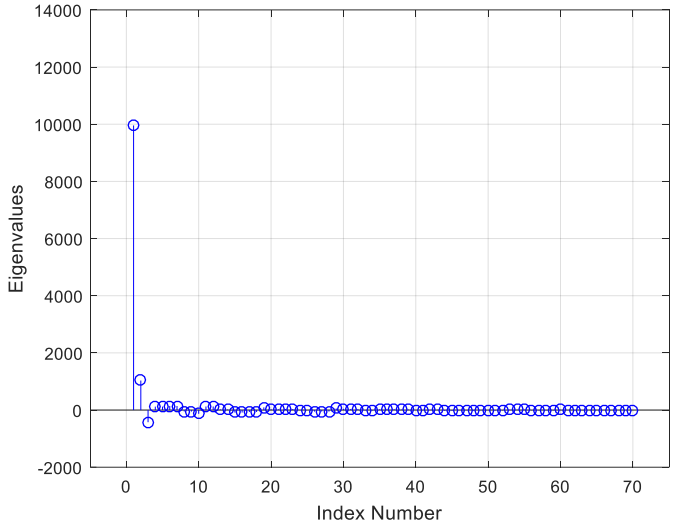
Implementation of the suggested model has been performed using C++ with MATLAB 2022a software on a system with an 11th Generation Intel Core i7-1165G7 processor @ 2.8 GHz, 16.00 GB RAM, 64- bit. ORL and Yale datastes have been used to validate the efficiency of the proposed algorithm in recognizing the face. ORL and Yale databases are used to verify the effectiveness of given algorithm. ORL dataset consists of 400 face images of 40 people with distinct face orientations and facial expressions. Yale database consists of 165 face images of 15 people with distinct illumination conditions and face behavior. First of all images of every individual are divided into training and testing pairs. Both training and testing face images are first filtered out to achieve maximum noise filtration. Unfiltered and filtered sample images are shown in 5.1 (a) and 5.1 (b) respectively. To speed up the process time, to achieve better recognition accuracy and to lessen memory consumption alongwith using only three useful SVD values, preprocessing stage has also been improved.

(a) (b)

Fig 5.1: Operation of CLAHE technique on sample images (a) Without filtering (b) With filtering

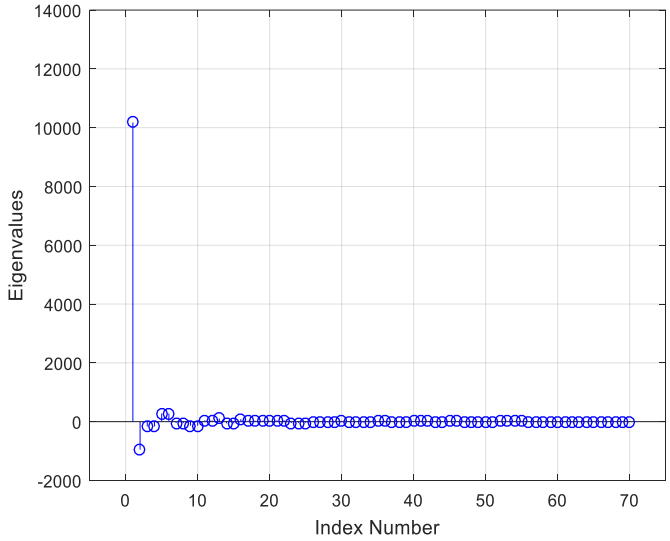
Recognition accuracy of the suggested model is calculated for each of these pairs. Few larger eigenvalues are used as image features. This is because larger eigenvalues contain most of the information about image. Some sample image and their corresponding eigenvalues are shown in Fig. 5.2(a) to 5.2(d). It is clear from the eigenvalues of this image that only two larger eigenvalues have most of the information while left over eigenvalues are insignificant.

(a) Sample Image 1



(b) Eigen Values

(c) Sample Image 2



(d) Eigen Values

Fig 5.2: Eigen values for test images for ORL and Yale databasets respectively

Parameters used for proposed model on Yale and ORL database are specified in Table 5.1. The ability of proposed model to successfully reject 85% and 91% unknown images is advantageous for trained ORL and Yale databases respectively. The proposed algorithm is effective in classifying face images captured at varying illumination conditions, different facial expressions, and behavior.

Table 5.1: Parameters design for ORL and Yale database.

Parameters	ORL Dataset	Yale Dataset
Block Height	5	5
Block overlap	75%	75%
Quantization level 1	18	18
Quantization level 2	10	10
Quantization level 3	7	7
Number of states	3	3
Index of training images	[5 6 7 8 9]	[5 6 7 8 9]
Index of testing images	[1 2 3 4 10]	[1 2 3 4 10 11]
Selected features	$\sigma_{11}, \sigma_{22}, U_{11}$	$\sigma_{11}, \sigma_{22}, U_{11}$
Image size	53x44	53x44
Remarks	Accuracy: 98.24% Total of 200 testing images for 40 individuals	Accuracy: 95.53% Total of 90 testing images for 15 individuals

The number of features are determined by number of states that are used to characterize the face besides computational complexity of the system. In two states HMM [32], the computational complexity also decreases with decrease in recognition rate to 82.76%. On the contrary, in 7-state [31] by using five training image (ORL and Yale Database) the predicted HMM model gives recognition rate of 97.78 % however, its computational complexity increases considerably. While in 5-state [30] MIT database 9 images were used for training set and achieved 90 % recognition. Although its recognition rate reduced as compared to seven state, but its computational complexity is less than seven state HMM as it takes less sequential information. Less number of states leads to minimum memory occupation is evident but on the same time recognition rate compromises.

Thus, it is obvious that it is an interchange between computational cost and recognition accuracy. 3-state HMM is improved as we are attaining satisfactory recognition rate using CLAHE technique in preprocessing and with minimized computational complexity using only three state HMM.

Table 5.2: Recognition time comparison for ORL Dataset

Image size	64×64 (7-state HMM)	32×32 (7-state HMM)	53x44 (4-state HMM)	53x44 (Proposed 3-state HMM)
Number of training image(s)	5	5	5	5
Recognition time per image(second)	0.28	0.15	0.05	0.045
Number of Symbols	1260	960	700	1260

Table 5.3 and 5.4 demonstrate the evaluation of the presented model with advanced methodologies on datasets of ORL and Yale respectively. HMM using three state has improved accurracy rate than other methods proposed in [33] to [42] except [36] for ORL and [34] for Yale in which 100% and 96.67% respectively, recognition rate was professed. Though computational complexity of [36] and [34] is considerably greater than our proposed HMM for FR.

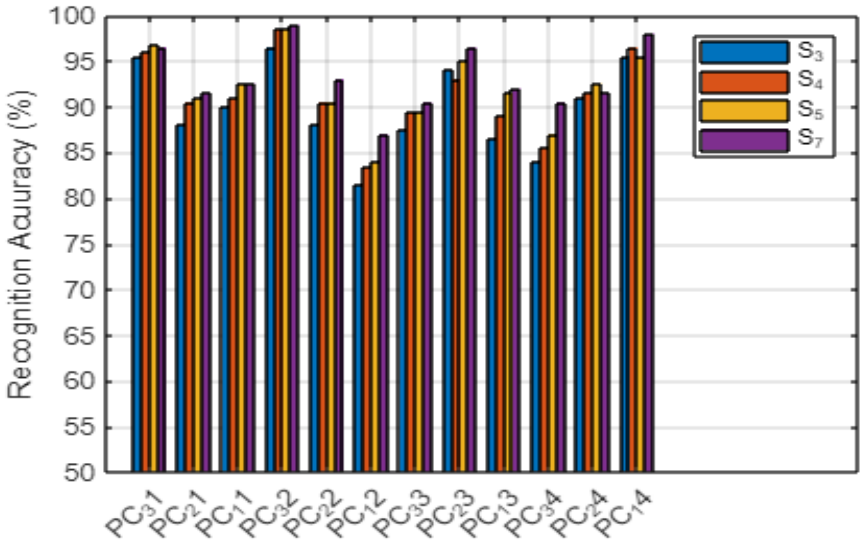
Table 5.3: Evaluation of the suggested technique with state-of-the-art methodologies on Yale database

Recognition Method	No. of training images/person	Recognition rate (%)
DIWTLBP [33]	5	97.00
LDA [37]	5	94.50
PCA + SVM [40]	5	92.00
ICA + SVM [41]	5	91.50
CFLDA [37]	5	97.85
DWT (SVD/LR + RWLDA/QR) [38]	5	97.75
Gabor & SRC [39]	5	97.50
LDA & SRC [39]	5	95.50
LDA +MDC [42]	7	90.62
ESGK [42]	6	97.50
Multi-class SVM [34]	5	96.50
2D-DMWT [35]	8	97.50
SVM-HMM [36]	6	100
Proposed Method	5	**98.24**

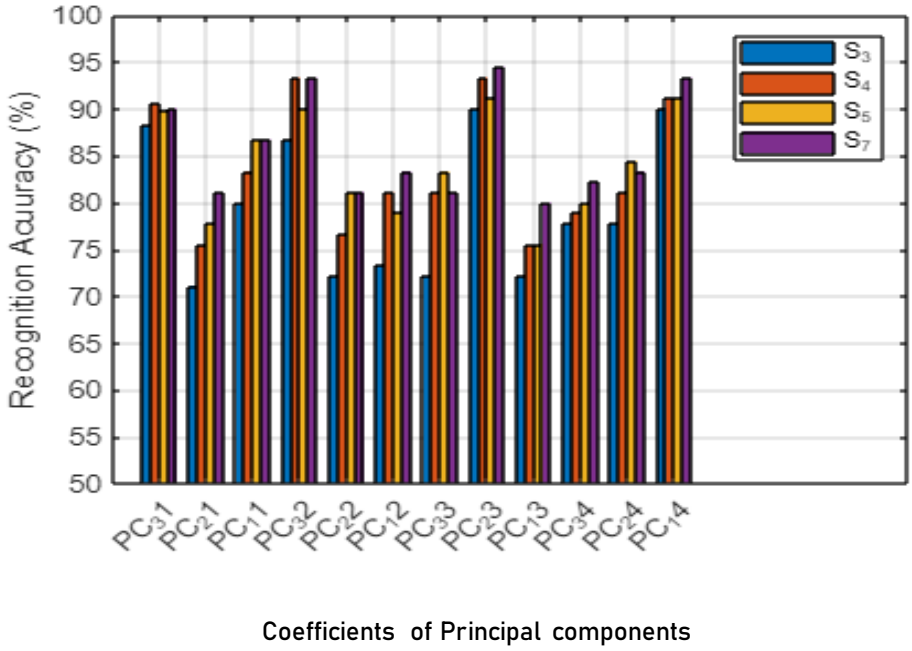
Table 5.4: Evaluation of the suggested technique with state-of-the-art methodologies on Yale database

Recognition Method	No. training images/person	Recognition rate (%)
LDA [37]	5	91.80
CFLDA [37]	5	94.77
PCA & SVM [40]	7	90.00
ICA & SVM [41]	6	92.00
Gabor & SRC [39]	5	92.22
LDA & SRC [39]	5	83.33
Multi-class SVM [34]	5	96.67
Proposed Method	5	**95.53**

Highest variation occurs in two foremost eigen values of image blocks while left over eigen values are insignificant. Therefore, these two principal eigen values have been used along changing the eigenvectors coefficients and computed accuracy rate for all eigenvectors coefficients. Fig 5.3 (a) and 5.3 (b) exhibits the accuracy of suggested algorithm for 4 distinct HMM states (S_3, S_4,S_5,S_7) on databases of ORL and Yale respectively. For all image blocks, PC_{ij} (i, j=1, 2, 3,) in these diagrams signify the j^{th} coefficient of i^{th} principal component. It is obvious from the graphs that satisfactory accuracy rate is attained for principal component coefficients with less computational complexity.



(a)



(b)

Fig 5.3: Accuracy of suggested algorithm using distinct states of HMM (a) ORL face dataset (b) Yale face dataset

Confusion matrix is used to show result of classification of face images i-e test image is properly identified or not. Correctly classified (TPR) and misclassified (TNR) rate is calculated for every class via the confusion matrix coefficients as presented in Fig. 5.4. The ROC curve intersect for values greater than false positive i-e 0.6, the ROC curve performs better than the average ROC curve. When associated with the accuracy of [32] using 2-state HMM, the suggested HMM for 3-state (three different face regions) achieved more recognition accuracy.

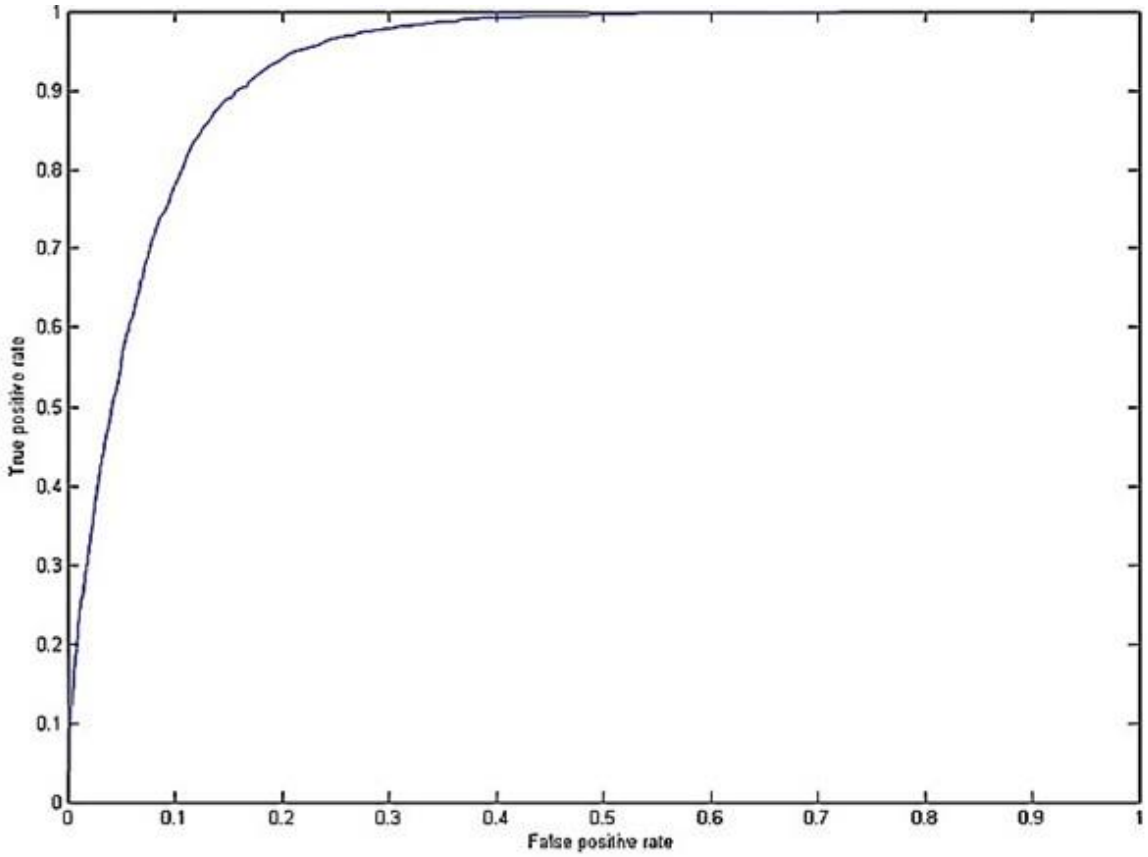


Fig 5.4: ROC curve between TPR and FPR

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

A three state efficient HMM model has been proposed for face recognition. PCA is used to find local features/observation sequence of face image blocks. A separate HMM is trained for each person using observation sequence of feature coefficients. Eigenvalues are computed for all persons in the database and largest eigenvalues of each person's training images are stored in a database. The model based on their eigenvalues rejects face images that do not belong to the trained database. Computational complexity and recognition error rate are computed for different states of HMM. Minimum value of complexity is obtained for three states HMM which represents the efficacy of suggested model. Experiments were conducted on face directories of ORL and Yale. Experimental outcomes verify that accuracy of suggested model is comparative to the existing methods for trained directories along with 85% and 91% unknown test images are successfully rejected for trained directories ORL and Yale respectively. Computational complexity of the suggested model has also been reduced as compared to higher order HMMs. Satisfactory recognition accuracy has been achieved even with less number of HMM states and achieved maximum noise filtration in preprocessing stage.

6.2 Future Work

The research work can be improved by including facial images with low intensity and pose variant. The use of 2-D HMM may improve the system performance more. Fusion of HMM with other existing techniques would increase the accuracy of FR system.

REFERENCES

[1] Srivastava, M., Kumar, A., Dixit, A., & Kumar, A. (2020, February). Real time attendance system using face recognition technique. *In 2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)* (pp. 370-373). IEEE.

[2] Tan, S. F., & Isa, N. A. M. (2019). Exposure based multi-histogram equalization contrast enhancement for non-uniform illumination images. *IEEE* Access, 7, 70842-70861.

[3] Zohaib, M., Shan, A., Rahman, A. U., & ALi, H. (2018). Image enhancement by using histogram equalization technique in Matlab. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, 7(2), 150-154.

[4] Kumar, A., Kaur, A., & Kumar, M. (2019). Face detection techniques: a review. *Artificial Intelligence* Review, 52(2), 927-948.

[5] Sharma, S., Bhatt, M., & Sharma, P. (2020, June). Face recognition system using machine learning algorithm. *In 2020 5th International Conference on Communication and Electronics Systems (ICCES)* (pp. 1162-1168). IEEE.

[6] Kanagaraj, H., & Muneeswaran, V. (2020, March). Image compression using HAAR discrete wavelet transform. *In 2020 5th International Conference on Devices, Circuits and Systems (ICDCS)* (pp. 271-274). IEEE.

[7] Ahmed, A., Guo, J., Ali, F., Deeba, F., & Ahmed, A. (2018, May). LBPH based improved face recognition at low resolution. *In 2018 international conference on Artificial Intelligence and big data (ICAIBD)* (pp. 144-147). IEEE.

[8] Hossain, M. T., Teng, S. W., Zhang, D., Lim, S., & Lu, G. (2019, September). Distortion robust image classification using deep convolutional neural network with discrete cosine transform. *In 2019 IEEE International Conference on Image Processing (ICIP)* (pp. 659-663). IEEE.

[9] Liu, S., Wang, Y., Peng, Y., Hou, S., Zhang, K., & Wu, X. (2020). Singular value decomposition-based virtual representation for face recognition. Machine Vision and *Applications, 31(3), 1-9.*

[10] Oloyede, M. O., Hancke, G. P., & Myburgh, H. C. (2020). A review on face recognition systems: recent approaches and challenges. Multimedia Tools and Applications, 79(37), 27891-27922.

[11] Crosswhite, N., Byrne, J., Stauffer, C., Parkhi, O., Cao, Q., & Zisserman, A. (2018). Template adaptation for face verification and identification. *Image and Vision Computing*, 79, 35-48.

[12] Bankar, R. T., & Salankar, S. S. (2019). Design of Eye Template Matching Method for Head Gesture Recognition System. *In Smart Innovations in Communication and Computational Sciences* (pp. 3-10). Springer, Singapore.

[13] M. K. Halidu, P. Bagheri-Zadeh, A. Sheikh-Akbari, and R. Behringer, "PCA in the context of Face Recognition with the Image Enlargement Techniques," *2019 8th Mediterranean Conference on Embedded Computing (MECO),* June. 2019.

[14] Y. Ren, Z. Song, S. Sun, J. Liu, and G. Feng, "Outsourcing LDA Based Face Recognition to an Untrusted Cloud," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2022.

[15] R. Naik, D. Pratap Singh, and J. Chaudhary, "A Survey on Comparative Analysis of Different ICA based Face Recognition Technologies," 2018 *Second International Conference on Electronics, Communication and Aerospace Technology (ICECA),* Mar. 2018.

[16] Mahmood, M., Jalal, A., & Evans, H. A. (2018, September). Facial expression recognition in image sequences using 1D transform and gabor wavelet transform. *In 2018 international* conference on Applied and Engineering Mathematics (ICAEM) (pp. 1-6). IEEE.

[17] Lal, M., Kumar, K., Arain, R. H., Maitlo, A., Ruk, S. A., & Shaikh, H. Study of face recognition techniques: A survey. *International Journal of Advanced Computer Science and* Applications, 9(6). 2018.

[18] Angadi, S. A., & Hatture, S. M. Face recognition through symbolic modeling of face graphs and texture. *International Journal of Pattern Recognition and Artificial Intelligence*, 33(12), 1956008. 2019.

[19] Kaur, P., Krishan, K., Sharma, S. K., & Kanchan, T. (2020). Facial-recognition algorithms: A literature review. *Medicine, Science and the Law,* 60(2), 131-139.

[20] Taunk, K., De, S., Verma, S., & Swetapadma, A. (2019, May). A brief review of nearest neighbor algorithm for learning and classification. *In 2019 International Conference on Intelligent Computing and Control Systems (ICCS)* (pp. 1255-1260). IEEE.

[21] Khan, M. Z., Harous, S., Hassan, S. U., Khan, M. U. G., Iqbal, R., & Mumtaz, S. (2019). Deep unified model for face recognition based on convolution neural network and edge computing. IEEE Access, 7, 72622-72633.

[22] Khudaybergenov, K. (2020). Choosing the structure of convolutional neural networks for face recognition. Bulletin of National University of Uzbekistan: *Mathematics and Natural Sciences*, 3(1), 20-32.

[23] Agarwal, V., & Bhanot, S. (2018). Radial basis function neural network-based face recognition using firefly algorithm. *Neural Computing and Applications*, 30(8), 2643-2660.

[24] Wechsler, H., Phillips, J. P., Bruce, V., Soulie, F. F., & Huang, T. S. (Eds.). (2012). Face recognition: *From theory to applications* (Vol. 163). Springer Science & Business Media.

[25] Lahaw, Z. B., Essaidani, D., & Seddik, H. (2018, July). Robust face recognition approaches using PCA, ICA, LDA based on DWT, and SVM algorithms. *In 2018 41st International Conference on Telecommunications and Signal Processing (TSP)* (pp. 1-5). IEEE.

[26] Kiani, K., & Rezaeirad, S. (2019). A new ergodic HMM-based face recognition using DWT and half of the face. *In 2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI)* (pp. 531-536). IEEE.

[27] O. R. L. Cambridge, "The orl database of faces," https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[28] T.Y.V.Group, "Yale faces," https://vismod.media.mit.edu/vismod/classes/mas622-00/datasets/.

[29] Bacci, G., Ingólfsdóttir, A., Larsen, K. G., & Reynouard, R. (2021, December). Active Learning of Markov Decision Processes using Baum-Welch algorithm. *In 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)* (pp. 1203-1208). IEEE.

[30] A. V. Nefian, and M.H. Hayes, "Hidden Markov models for face recognition," *in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2721–2724, 1998, doi: 10.1109/ICASSP.1998.678085.

[31] Dinkova, P., Georgieva, P., Manolova, A., & Milanova, M. (2016, June). Face recognition based on subject dependent Hidden Markov Models. In *2016 IEEE international Black Sea conference on communications and networking (BlackSeaCom)* (pp. 1-5). IEEE.

[32] Farhan, H. R., Al-Muifraje, M. H., & Saeed, T. R. (2016, May). Using only two states of discrete HMM for high-speed face recognition. In *2016 Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA)* (pp. 1-5). IEEE.

[33] M. A. Muqeet and R. S. Holambe, "Local binary patterns based on directional wavelet transform for expression and pose-invariant face recognition," *Applied Computing and Informatics*, vol. 15, no. 2, pp. 163–171, 2019.

[34] Pk, A. M. N., Ding, X., & Page, T. (2020, April). An Integrated Approach for Face Recognition Using Multi-class SVM. In *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)* (pp. 398-402). IEEE.

[35] A. Aldhahab and W. B. Mikhael, "A facial recognition method based on DMW transformed partitioned images," *in Midwest Symp. on Circuits and Systems, MWSCAS*, Boston, MA, pp. 1352– 1355, 2017.

[36] S. Nebti, and B. Fadila, "Combining classifiers for enhanced face recognition," *in Int. Conf. Advances in Information Science and Computer Engineering*, Dordrecht, The Netherlands, 2015.

[37] Y. Wei, "Face recognition method based on improved," *in Proc.-9th Int. Conf. on Intelligent HumanMachine Systems and Cybernetics*, Hangzhou, pp. 456–459, 2017.

[38] M. Ayyad and C. Khalid, "New fusion of SVD and relevance weighted LDA for face recognition," *Procedia Computer Science*, vol. 148, no. 4, pp. 380–388, 2019.

[39] G. Song, D. He, P. Chen, J. Tian, B. Zhou et al., "Fusion of global and local Gaussian-Hermite moments for face recognition," *Chinese Conf. on Image and Graphics Technologies and Applications, Communications in Computer and Information Science Book Series*, vol. 1043, pp. 172–183, 2019.

[40] X. Chen, L. Song, C. Qiu, "Face Recognition by Feature Extraction and Classification," *In: 2018 12th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID),* pp. 43-46, November 2018.

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

[42] L. Dora, S. Agrawal, R. Panda and A. Abraham, "An evolutionary single Gabor kernel based filter approach to face recognition," *Engineering Applications of Artificial Intelligence*, vol. 62, no. 14, pp. 286– 301, 2017.