User Authentication Using Face

M. Tech Project report submitted for partial fulfilment for the award of degree of

Master of Technology

In

Visual Information and Embedded Systems

by

GUGULOTHU PANDU
14EC65R28

Under the guidance of
Dr. Sudipta Mukhopadhyay

Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur
Spring Semester, 2015-16
April, 2016

CERTIFICATE

This is to certify that the project report entitled "User Authentication Using face" submitted by Gugulothu Pandu (Roll No. 14EC65R28) to Indian Institute of Technology, Kharagpur towards partial fulfillment of requirements for the award of degree of Master of Technology in Electronics and Electrical Communication Engineering is a record of bonafide work carried out by him under my supervision and guidance during academic session 2015-2016.

Date -
Place -

Dr. Sudipta Mukhopadhyay

Department of Electronics
& Electrical Communication Engineering,
Indian Institute of Technology, Kharagpur

DECLARATION

I certify that

1. The work contained in this report has been done by me under the guidance of my supervisor.

2. The work has not been submitted to any other Institute for any degree or diploma.

3. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

4. Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

Date: **GUGULOTHU PANDU**
Place: Kharagpur **14EC65R28**

Department of Electronics & Electrical Communication Engineering,
Indian Institute of Technology, Kharagpur

ACKNOWLEDGEMENT

I would like to take this opportunity to express my deepest sense of gratitude and sincere appreciation to my project guide **Dr. Sudipta Mukhopadhyay** for constantly supervising my progress. His continual guidance was very helpful to me throughout the process of this project. His encouraging attitude has been immensely valuable in keeping me motivated and helping me in overcoming all obstacles in my work. He has been a great teacher, a great guide and above all a great human being. It is always my pleasure to interact with him on any topic.I wish to thank all our professors in Department of Electronics & Electrical Communication Engineering for their support and motivation to complete my course successfully. I am indebted to each and every one of my classmates and friends, for helping me in all ways and made these five years the most memorable one of my life. Finally endless thanks to my family who have always stood by my side and boosted my confidence.

Date: Signature of the Student

Abstract

Our research involves development of Face Recognition based user authentication system that can be used for security of applications. We have developed a technique where the Facial Features of an individual can be used to authenticate from an intruder who may try to access the secure data. Face Recognition is a process of identifying or verifying a person from an image or video. This is done by comparing selected facial features from the image and a face database. There are many security applications which rely on password and ID badges which can be easily stolen or lost.

Biometrics like Face, Keyboard dynamics and mouse dynamics are more reliable, convenient and efficient as they don't suffer above defects. Sometimes, Authentication using behavioral biometrics like keystroke dynamics and mouse dynamics can be affected with variations of the keystrokes and mouse dynamics with emotional state of the user. Face Recognition is robust to these effects.

User Authentication using Face image involves two steps Face Detection and Face Recognition. Face Detection test is carried out using FDDB (Face Detection Data set and Benchmark) Database. A True positive rate of 0.84 is obtained for test of 300 images Face Recognition via Sparse Representation based classification is tested on Yale B database and Recognition rate of 0.92 is obtained for the feature dimension of 440 with down-sample images. In the case of SRC with Uniform local binary patterns histogram, recognition rate of 0.92 is obtained for feature dimension of 531.The whole Face Recognition Processing module is integrated with other Authentication system that uses Keyboard Dynamics to make User Authentication more accurate.

Keywords: Histogram of Oriented Gradients (HOG), Face Detection, Support Vector Machine (SVM), Eigenfaces, Face Recognition, and Sparse Representation based Classifier (SRC).

Chapter 1

Introduction

1.1 Introduction to Biometric Authentication

Authentication refers to the process of verifying a person's legitimate right prior to the release of secure resources. This can be achieved by counterchecking unique information provided by an individual. In the recent years, Biometric authentication has become popular in modern society. It has been used for many purposes such as personal identification, security system, computer interaction, smart card, surveillance and access control. The traditional authentication systems use passwords, ID cards and smart cards. A password or PIN can be forgotten or lost. Also, smart cards and ID badges devices can be easily misplaced, stolen or forgotten, but biometrics cannot be forgotten or lost and requires physical presence of the person to be authenticated. Thus personal authentication systems using biometrics like keystroke dynamics, mouse dynamics and face are more reliable, convenient and efficient than the traditional identification methods.

A biometric system which relies only on a single biometric identifier in making a personal identification is often unable to meet the desired performance requirements. A single biometric sometimes may fails to be accurate enough for the identification of a large user population. Another disadvantage of using only one biometric is that the physical characteristics of a person for the selected biometric might not be always available or readable. To cater for the problems and limitations of single biometric, the multimodal biometrics keystroke dynamics, mouse dynamics and face are used, leading to the improvement of the system's performance.

1.1.2 Types of Biometrics

The biometrics can be broadly classified into two types:

- Physiological biometrics
- Behavioural biometrics

Physiological biometrics involves physiological characteristics of a human being used as a biometric such as voice, DNA, fingerprint, IRIS pattern or hand geometry. These biometrics are more reliable and accurate [1]. They are not affected by any mental conditions such as stress or illness. We shall look at some of the physiological biometrics and their strengths and weakness.

1. **Iris pattern**: Iris is the annular region of the eye which controls the amount of light that enters it [2]. Iris recognition technology is used primarily in high security environments, where low error rates are essential. But the problem with Iris patterns is that its accuracy is affected by changes in lighting, its scanners are more expensive. Moreover, the recognition is difficult to perform at a distance longer than few meters.

2. **Fingerprint**: The analysis of fingerprints for matching purposes generally requires the comparison of several features of the print pattern. These patterns don't change much

over the time period and differ from human to human, hence if properly recognized the biometric can be used to a great level of accuracy. The sensor is cheap and its size is also small. But still there is a chance that at the high level of security the fingerprint recognition will fail since any intruder may access a legitimate user’s fingerprints and can use it to login to the system.

3. **Face**: The ability of distinguishing one individual from another is possessed by virtually every human. Facial metrics technology relies on the measurement of the specific facial features (the systems usually look for the positioning of the eyes, nose and mouth and the distances between these features) [1]. The accuracy of the face recognition systems improves with time, but it has not been very satisfying so far. It requires a person to sit at a proper distance from the camera. Also, the illumination factor affects the performance.

Behavioural biometrics involves the behavioural characteristics of a human being. These biometric characteristics are acquired over time by an individual, and are at least partly based on acquired behaviour. Thus, it is something known to an individual and thus can be exploited in authentication purposes. The best known behavioural biometrics is depicted below:

1. **Keyboard Dynamics**: This is a behavioural biometric which are characterized by the way a user presses a key on the keyboard or the pattern of typing, which differs from individual to individual. The authentication in keyboard can be carried out in two ways using fixed or free text [3]. In case of fixed text, all the participants are asked to type same kind of text data. Whereas, free text is carried out in a continuous manner where the analysis is done throughout the active login period. The biometric does not involve any special hardware for data collection. However, the only difficulty with it is that the behaviour of a user doesn’t remain constant throughout the active session.

2. **Mouse Dynamics**: This behavioural biometric is characterized by the way an individual moves the mouse or clicks on the screen of the desktop/laptop. Mouse actions like mouse movements, clicks, drag and drop etc. can be used as useful features. This behavioural biometric also has issue with variability of features over time. Even though related work on mouse dynamics has suggested that it can still be used as a biometric.

Our User Authentication System is based on three biometrics Keystroke dynamics, mouse dynamics and face. Within this thesis, we would like to concern about Face Biometric related Recognition system and its application in User Authentication. We also deal with integration of two authentication modules that uses Face and Keystroke dynamics.

1.2 Motivation

Now-a-days, Computer systems are widely used everywhere. They are used in the almost all aspects of our lives storage of personal information, net banking, buy and sell stocks , other online transactions like online shopping etc. This is a strong reason to protect them against illegal intrusions. So, we have chosen Biometrics like keystroke dynamics, mouse dynamics and face as they are more reliable and efficient. Also they don’t require extra hardware like in the case of retina and iris.

My motivation towards this project is due to necessity of real world application that achieves higher accuracy of user authentication and high level of interactions with the machine. The project can also be used to build a complete face recognition system which can monitor an employee continuously in real –time, which will give us understanding of employee’s emotion or state at work place. This system can also be used for understanding the state of the student while coming to classes and medical condition of patient.

1.3 Objective and Scope of Work:

- To do research on the current existing techniques of Face Detection and Face Recognition for user authentication.

- To integrate Face Recognition Processing module with other Authentication module that uses Keyboard Dynamics.

- The work can be extended to build face recognition system that can monitor a person continuously in real –time, which will give us understanding of person’s emotion or state at work.

1.4 Overview of Face Recognition:

Face recognition is a process of identifying or verifying a person from an image or a video source. One of the ways to do this is by comparing selected facial features from the image and a facial database. A face recognition system should be able to identify faces present in images and videos automatically. It can operate in either or both of two modes: (1) Face verification and (2) face identification.

Face verification involves a one to one match that compares a query face image against a template (model) face image whose identity is being claimed. Face identification involves a one to many matches that compares a query face image against all the template (model) images in the database to determine the identity of the query face. Face identification mode is discussed in this thesis due the fact that we will have large Database in general.

A face recognition system can split into *three* modules as shown in fig[1.1]. When the system finishes scanning a video or photo of a user’s face, the digitalized information will go through these following modules one after another. The first step involves acquiring a Face image or video and detection of face in the image if any. In second stage, important features will be extracted from face such that they will discriminate from other face images. The relevant features then compared with different classes already stored in Database. Suitable classifier need to be selected to obtain high recognition rates.

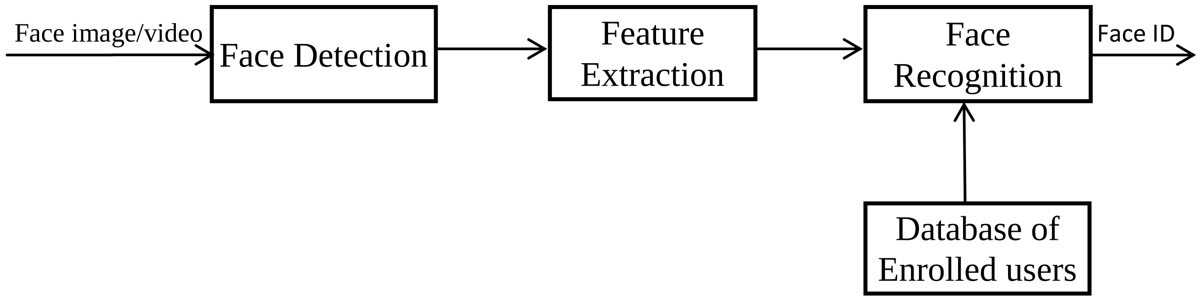

Fig 1.1 Flow of Face Recognition Processing

1.5 Organization of the Thesis

A brief overview of the work carried out in the thesis is organized as follows:

- Chapter 1: Introduction
  This chapter presents an introduction to the need of biometric authentication in PCs/laptops, types of biometrics and their performances and problem definition.

- Chapter 2: Literature Survey
  This chapter summarizes the previous work done on Face Detection and Face Recognition.

- Chapter 3: Face Detection
  This chapter describes the entire procedure followed to develop Face Detection module using HOG features and results are given discussed.

- Chapter 4: Face Recognition
  This chapter describes the entire procedure followed to develop Face Recognition module using Sparse Representation based Classifier and results are given discussed.

- Chapter 5: User Authentication Module
  This chapter describes about Face Recognition Processing module integration with Keyboard Dynamics module to develop User Authentication module.

- Chapter 6: Conclusion and Future work
  This chapter concludes the thesis and the work to be done in the future are discussed.

Chapter 2

Literature Survey

2.1 General Face Detection Techniques

Face detection is the foremost step in the process of Biometric authentication using Face. It is the step stone to all facial analysis algorithms including face alignment face modelling, face recognition, face authentication and facial expression recognition. Generally, face detection approaches can be divided into four main categories namely feature based, appearance based, knowledge-based, and template matching method.

2.1.1 Feature based methods:

These algorithms aim to find structural features that exist even exist for different pose, viewpoint and lighting conditions. Some features that can be used to detect face(s) include skin colour, nose, ears, eyes, and mouth etc. Furthermore, some studies have proved that colour of skin is an excellent feature for detecting faces among other objects due to different people have different skin colour[4]. Additionally, human faces have particular textures which can be used to differentiate between face and other objects. Moreover, edge of features can help to detect the objects from the face.

2.1.2 Appearance based methods

Appearance based method is also another type of face detection methods. In this type, face detection is considered as a 2-class pattern recognition issue. This method also uses classification and it uses the features in the search window. The features are calculated from the values of pixels. For example, MCT (Modified Census Transform), Haar-like features are some types which have been used for the method.

In addition, this method is a learning based method and the classifier required to be created by using a statistical learning between the enormous instances. For instance, adaptive boosting (AdaBoost) includes some weak classifiers to create a cascade classifier which is multi stage and effective [5].

Viola Jones Face detection

A real time face detection algorithm capable of achieving high detection rates[5]. Basic idea is to slide a window across image and evaluate a Feature value at every location. Haar like Features are used to detect the faces which share some similar properties with faces like the eyes region is darker than the upper-cheeks, the nose bridge region is brighter than the eyes. This method involves mainly four steps as described below.

a. Haar-like Features:

These rectangular features where the value of a *two-rectangle feature* is the difference between the sum of the pixels within two rectangular regions. The regions have the same size and shape and are horizontally or vertically adjacent (Fig 2.1). A *three-rectangle feature* computes the sum within two outside rectangles subtracted from the sum in a centre rectangle. Finally a *four-rectangle feature* computes the difference between diagonal pairs of rectangles.

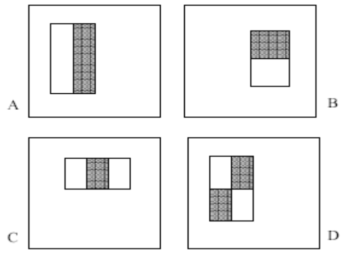

Fig 2.1Haar Like Features

b. Integral Image:

Integral Image representation decreases the computational complexity for calculating Haar Features value. The integral image at location (x,y), is the sum of the pixels above and to the left of (x,y), inclusive. The Integral image can be computed in a single pass and only once for each sub-window.

c. Adaboost:

In this Method, a variant of AdaBoost is used both to select the features and to train the classifier. In its original form, the AdaBoost learning algorithm is used to boost the classification performance of a simple learning algorithm (e.g., it might be used to boost the performance of a simple perceptron).It does this by combining a collection of weak classification functions to form a stronger classifier. In the language of boosting the simple learning algorithm is called a weak learner. So, for example the perceptron learning algorithm searches over the set of possible perceptrons and returns the perceptron with the lowest classification error.

$$F(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x) + \alpha_3 f_3(x) + ... + \alpha_n f_n(x) \qquad (2.1)$$

$$f_i(x) = \begin{cases} 1 & \text{if } g_i(x) > \theta_i \\ -1 & \text{otherwise} \end{cases} \qquad (2.2)$$

F(x) is the overall “strong classifier” which is the linear combination of weak classifiers.

d. Cascading:

This section describes an algorithm for constructing a cascade of classifiers which achieves increased detection performance while radically reducing computation time. The key insight is that smaller, and therefore more efficient, boosted classifiers can be constructed which reject many of the negative sub-windows while detecting almost all positive instances. Simpler classifiers are used to reject the majority of sub-windows before more complex classifiers are called upon to achieve low false positive rates.

2.1.3 Template based methods:

Template matching based methods are commonly used to obtain regions with the most possibilities to be human face. A template is an instance of the objects or features of a face. Template based methods use the relation between the pattern of the input image and the predefined pattern of the face or its features. Template based methods are simple in terms of implementation.

2.1.4 Knowledge based methods:

These methods are based on human knowledge of the typical human face geometry and arrangement of facial features. Facial features will be extracted first, and face candidates are identified, based on the coded rules. Several methods to detect faces based on facial appearance have been developed [6].

2.2 Face Recognition Approaches

The goal of face Recognition system is image understanding -the ability to detect recover image structure but also to know what it represents. A general statement of face recognition can be formulated as follows: given still or video images of a scene, identify or verify one or more persons in the scene using a stored database of faces. The solution to the problem involves segmentation of faces (face detection) from cluttered scenes, feature extraction from the face regions, recognition or verification.

Face recognition can be done in both a still image and video which has its origin in still image face recognition. Different approaches of face recognition can be categorized into tree main groups such as:

1. Feature based approach
2. Holistic approach
3. Hybrid approach

2.2.1 Feature based approach

Face recognition based on the geometric features of a face is probably the most intuitive approach to face recognition. One of the first automated face recognition systems was described in [7]: marker points (position of eyes, ears, nose) were used to build a feature vector (distance between the points, angle between them). The recognition was performed by calculating the Euclidean distance between feature vectors of a probe and reference image. Such a method is robust against changes in illumination by its nature, but has a huge drawback: the accurate registration of the marker points is complicated, even with state of the art algorithms. Some of the latest work on geometric face recognition was carried out in [8]. A 22-dimensional feature vector was used and experiments on large datasets have shown, that geometrical features alone my not carry enough information for face recognition.

2.2.2 Holistic approach

The Eigenfaces method described in [9] took a holistic approach to face recognition: A facial image is a point from a high-dimensional image space and a lower-dimensional representation is found, where classification becomes easy. The lower-dimensional subspace is found with Principal Component Analysis, which identifies the axes with maximum variance. While this kind of transformation is optimal from a reconstruction standpoint, it doesn't take any class labels into account. Imagine a situation where the variance is generated from external sources, let it be light. The axes with maximum variance do not necessarily contain any discriminative information at all, hence a classification becomes impossible. So a class-specific projection with a Linear Discriminant Analysis was applied to face recognition in [10]. The basic idea is to minimize the variance within a class, while maximizing the variance between the classes at the same time.

Recently various methods for a local feature extraction emerged. To avoid the high-dimensionality of the input data only local regions of an image are described, the extracted features are (hopefully) more robust against partial occlusion, illumination and small sample size. Algorithms used for a local feature extraction are Gabor Wavelets, Discrete Cosines Transform and Local Binary Patterns [11]. It's still an open research question what's the best way to preserve spatial information when applying a local feature extraction, because spatial information is potentially useful information.

2.2.3 Hybrid approach

The idea of this method comes from how human vision system perceives both holistic and local feature. The key factors that influence the performance of hybrid approach include how to determine which features should be combined and how to combine, so as to preserve their advantages and avert their disadvantages at the same time. There are modular Eigenfaces, hybrid local feature, shape normalized, component based methods in hybrid approach.

Chapter 3

Face detection

Face detection refers to the process of detecting faces in an image or video source if any. It is the foremost step in the process of Biometric authentication using Face. It is also first step to all facial analysis algorithms including face alignment, face modelling, face recognition, head pose tracking, facial expression recognition. So there is a necessity of real time face detection algorithm that capable of achieving high detection rates. Face Detection is a binary Classification problem that classifies a face and non-face. Generally, Face Detection mainly follows two approaches. First approach is based on the facial features like skin colour, eyes, mouth and nose. Other approach is to pass a Sliding window through the image and extract features over window.

3.1 Sliding window Face Detection

Sliding-window object detection is a popular technique for identifying objects in an image. The approach involves scanning the image with a fixed-size rectangular window and applying a classifier to the sub-image defined by the window. The classifier extracts image features from within the window (sub-image) and returns the probability that the window (tightly) bounds a particular object. The process is repeated on successively scaled copies of the image so that objects can be detected at any size. Usually non-maximal neighbourhood suppression is applied to the output to remove multiple detections of the same object. Features like Haar Cascades, HOG and SIFT that can be obtained from faces and non-faces over sliding window are used for classification. Among these, HOG is shown to be a robust feature set that allows object detection with high accuracy of detection even under difficult illumination conditions.

3.2 Histogram of Oriented gradients

HOG is a robust feature set that allows object detection, even in cluttered backgrounds under difficult illumination [12].The method is based on evaluating a dense grid of well-normalised local histograms of image gradient orientations over the image windows. The hypothesis is that local object appearance and shape can often be characterised rather well by the distribution of local intensity gradient or edge directions, even without precise knowledge of the corresponding gradient or edge positions. It can be used for any generic object detection method. The later section presents the complete processing chain of the feature extraction algorithm. The following sections present Extraction of HOG features and training them with SVM classifier.

3.2.1 HOG Feature Extraction Algorithm

The steps in the extraction of HOG features over a sliding window are describes as below as shown in fig 3.1:

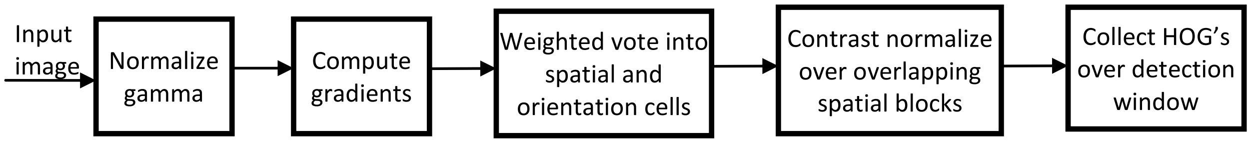

Fig 3.1 HOG Feature Extraction

- The first stage applies an optional global image normalisation equalisation that is designed to reduce the influence of illumination effects. In practice we use gamma (power law) compression, either computing the square root or the log of each colour channel.
- The second stage computes first order image gradients. These capture contour, silhouette and some texture information, while providing further resistance to illumination variations.
- The third stage aims to produce an encoding that is sensitive to local image content while remaining resistant to small changes in poses or appearance. The image window is divided into small spatial regions, called "cells". For each cell we accumulate a local 1-D histogram of gradient or edge orientations over all the pixels in the cell. This combined cell-level 1-D histogram forms the basic "orientation histogram" representation.
- The fourth stage computes normalisation, which takes local groups of cells and contrast normalises their overall responses before passing to next stage. Normalisation introduces better invariance to illumination, shadowing, and edge contrast.
- The final step collects the HOG descriptors from all blocks of a dense overlapping grid of blocks covering the detection window into a combined feature vector for use in the window classifier.
- Finally, concatenating of HOG features over each cell results in final HOG descriptor.

HOG Descriptor:

A window size of 40 X 40 pixels is considered for sliding window detection. It is divided into cells; each cell is of 4 X 4 pixels. Each Block is formed with 2 cell X 2 cell as shown in fig 3.2. Histogram of oriented gradients is divided into nine bins ($0 - 180^0$) and calculated over each cell. Contrast normalisation is performed over each block with 50% of overlap between blocks. Now, HOG histograms of all blocks are concatenated to form a vector called HOG descriptor.

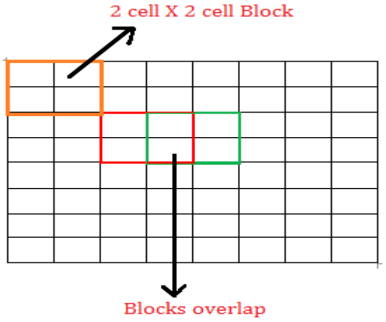

Fig 3.2 HOG Descriptor

Dimension of HOG Descriptor = no of bins*no of cells per block*total no of blocks.

3.3 Support Vector Machine

Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression and pattern recognition [13]. Given a set of training examples, each marked for belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possibleas shown in fig 3.3.

Given set of positive and negative samples, SVMs maximize the *margin* around the separating hyperplane. The decision function is fully specified by a subset of training samples called support vectors. Let x_i be the training sample belongs to class y_i. y_i will takes values +1 for class 1 and -1 for class 2.

For a given x_i,

$$w.x_i + b \geq 1 \; ; \quad \text{if} \;\; y_i = +1 \qquad (3.1)$$

$$w.x_i + b \leq 1 \; ; \quad \text{if} \;\; y_i = -1 \qquad (3.2)$$

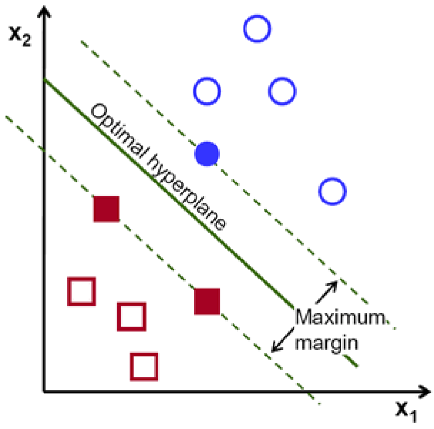

Fig 3.3 SVM

The formulation of the Optimization Problem with weight vector *w* and bias *b* is

$$\text{Minimize} \qquad \Phi(w) = \frac{1}{2} w^t w$$

$$\text{Subject to} \qquad y_i(wx_i + b) \geq 1 \quad \forall i \qquad (3.3)$$

This is a problem of Lagrangian optimization that can be solved using Lagrange multipliers to obtain the weight vector *w* and the bias *b* of the optimal hyperplane.

3.3.1 Soft Margin SVM

Generally, training data will not linearly separable due to the presence of noise in data. The standard approach is to allow the large decision margin to make a few mistakes (some points - outliers or noisy examples - are inside or on the wrong side of the margin). We then pay a cost for each misclassified example, which depends on how far it is from meeting the margin requirement. To implement this, we introduce *slack variables* ξ_i . A non-zero value for ξ_i allows x_i to not meeting the margin requirement at a cost proportional to the value of ξ_i.

The formulation of the SVM optimization problem with slack variables is:

$$\min \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$$

$$\text{Subject to} \qquad y_i(w \cdot x_i + b) \geq 1 - \xi_i, \; \forall x_i \qquad (3.4)$$

$$\xi_i \geq 0$$

The optimization problem is then trading off how fat it can make the margin versus how many points have to be moved around to allow this margin. The margin can be less than 1 for a point x_i by setting ξ_i,>0, but then one pays a penalty of $C\xi_i$ in the minimization for having done that. The sum of the ξ_i gives an upper bound on the number of training errors. Soft-margin SVMs minimize training error traded off against margin. The parameter C is a *regularization* term, which provides a way to control over fitting: as C becomes large, it is unattractive to not respect the data at the cost of reducing the geometric margin; when it is small, it is easy to account for some data points with the use of slack variables and to have a fat margin placed so it models the bulk of the data.

3.4 Classification Methodology:

The overall object detection architecture is built around a method for classifying individual image regions. This is divided into two phases. The *learning phase* creates a binary classifier that provides object/non-object decisions for fixed sized image regions *windows*, while the *detection phase* uses the classifier to perform a dense multi-scale scan reporting preliminary object decisions at each location of the test image. These preliminary decisions are then fused to obtain the final object detections. Both the *learning phase* and the *detection phase* contain three stages. Figure 3.1 depicts these. Overall this defines a fixed and relatively simple architecture for object detection. The final detector performance depends on the accuracy and reliability of the binary classifier and on how multiple detections are fused during the detection phase.

The first stage of learning is the creation of the training data. The positive training examples are fixed resolution *image windows* containing the centred object, and the negative examples are similar windows that are usually randomly subsampled and cropped from set of images not containing any instances of the object. The binary classifier is learned using these examples. This simple window architecture has various advantages. It allows a conventional classifier to be used for detection and relieves the classifier of the responsibility to be invariant to changes in position and scale (although invariance to other types of transformations, changes in pose and viewpoint, and illumination still has to be assured). It also means that the classifier works in relative coordinates (feature position relative to the centre of the current window) which allows relatively rigid template-like feature sets to be used. On the other hand it means that the classifier is run on a large number of windows, which can be computationally expensive and which makes the overall results very sensitive to the false positive rate of the classifier.

The image feature extraction process maps image windows to a fixed size feature space that robustly encodes visual form. Our Feature extraction process involves extraction of HOG descriptor over fixed image window. We use *linear* SVM as our baseline binary classifier as it proved to be the most accurate, reliable. Three properties of linear SVM make it valuable for comparative testing work: it converges reliably and repeatedly during training; it handles large data sets gracefully; and it has good robustness towards different choices of feature sets and parameters.

During detection, the input test image is scanned at all scales and locations. For each scale and location, the feature vector is computed over the detection window, just as in the learning phase, and the binary classifier (SVM) is run to produce object/non-object decision for the window. Image regions that contain objects typically produce multiple firings and it is necessary to fuse these overlapping detections into a single coherent one. To scan images at different scales, we form given image into Feature pyramid. Feature pyramid is a representation of images into multiple scales by smoothing and subsampling to extract features of interest and avoid noise. Sliding window size of 40X 40 pixels is used and those strides through entire image with stride of cell size along each row. Filter Score at a position p is obtained by dot product of learned SVM Filter 'F' and HOG descriptor.

Let H be a HOG pyramid and p = (x, y, l) be a rectangular window in the l-th level of the pyramid.

$$\text{Score at location } p = F.\emptyset(p,H) \qquad (3.5)$$

Here, $\emptyset(p,H)$ = HOG features in sub window specified by location p.

Detection phase typically produces multiple overlapping detections for each object instance. These detections need to be fused together. For this, we use a principled solution based on representing detections in a position scale pyramid. Each detection provides a weighted point in this 3-D space and the weights were the detection's confidence score. A non-parametric density estimator is run to estimate the corresponding density function and the resulting modes (peaks) of the density function constitute the final detections, with positions, scales and detection scores given by value of the peaks. We will call this process as *non-maximum suppression*.

3.5 Results and Discussions:

We have collected 1000 images from Face Detection Data Set and Benchmark (FDDB) database. The whole data is divided into 80:20 ratios. 600 images are chosen for training and 200 are for validation. Some images are with more than one face. Total no of faces labelled for training are 652. Sliding window will scan at all possible scales and positions. Non-face patches are the windows those don't over with the labelled face patches. All the images are trained with soft margin SVM with six level of Feature pyramid. Model is trained with different C values to obtain optimum operating point.

Choosing proper Regularization parameter C value avoids over fitting thus performs best on new real world test data. So, data is with trained with different C values and is tested with validation set of 200 face images. C value increased by one in each step up to 10 and they by value of 10 after that to obtain True positive rate (TPR) and False positives (FP). The Receiver Operating Characteristics (ROC) curve for different values of is as shown in fig 3.4. ROC is drawn for True positive rate vs. false positives. Optimal threshold for operating is obtained as C = 60.

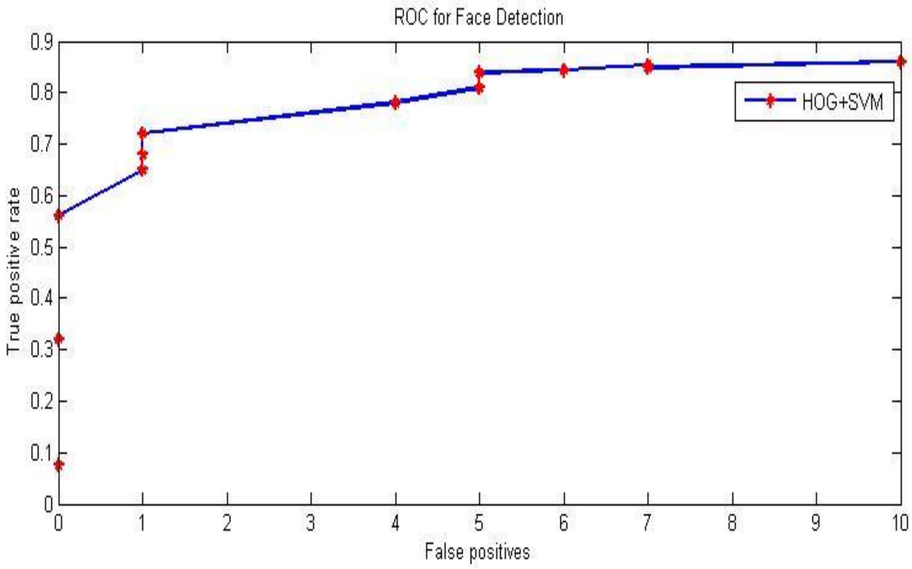

Fig 3.4 ROC curve for Face detection

We have tested the trained SVM classifier with HOG features with FDDB database. 200 images are considered for testing. Face Detection results are shown image no.3.1 and image no. 3.2 and image no 3.3. One of the false positive detected by the model is shown in image no. 3.4.

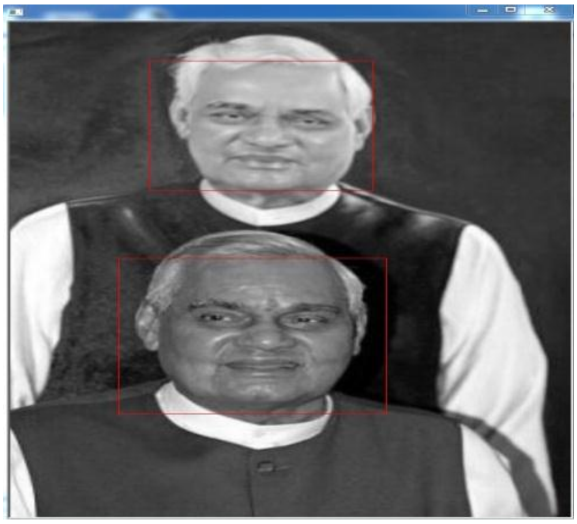

Image 3.1

Image 3.2

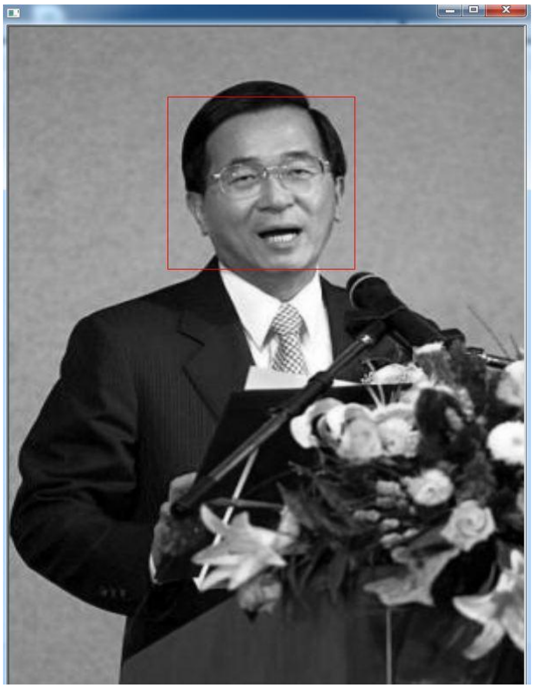

Image 3.3

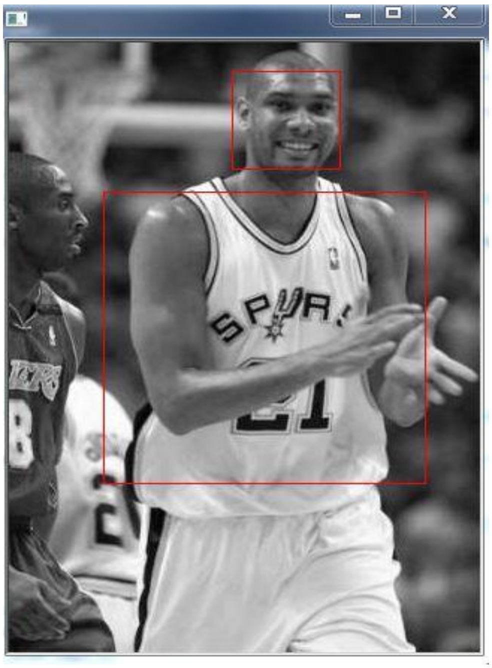

Image 3.4

Implementation of Face detection using Sliding window method is discussed in this chapter. The HOG descriptors extracted over sliding windows over face and non-face patches are used for train SVM classifier. True positive rate (Recall) of 0.84 is obtained on test images of FDDB database with 5 false positives. Optimal threshold value for Regularization parameter C is obtained as 60. Detection rate using HOG descriptor is 5 frames per second. True positive rate for the detection can be increased with high training data and Detector rate can be increased using Dimensionality reduction but it could affect the detection accuracy.

Chapter 4

Face Recognition

Face recognition is a process of identifying or verifying a person from an image or a video source. This task is generally performed by comparing selected facial features from the image and a facial database. Face recognition systems were known to be critically dependent on discriminative feature extraction methods, such as eigenfaces, Fisherfaces and Laplacianfaces. In recent years, Wright et al. have demonstrated that, once the test image can be approximated by a sparse linear combination of the training images, the choice of feature space is no longer critical [14]. Sparse Representation-Based Classification (SRC) is a face recognition breakthrough which has successfully addressed the recognition problem with sufficient training images of each gallery subject. In the later section, we discussed about sparse representation and classification depends on such a representation.

4.1 Face Recognition via sparse Representation

Researches have recently revealed that human vision [15], many neurons in the visual pathway are selective for a variety of specific stimuli, such as colour, texture, orientation, scale, and even view-tuned object images. Considering these neurons to form an overcomplete dictionary of base signal elements at each visual stage. Similarly, if sufficient training samples are available from each class, it will be possible to represent the test samples as a linear combination of just those training samples from the same class. This representation is naturally sparse, involving only a small fraction of the overall training database.

In this section, we first illustrate the core idea of SRC in a slightly artificial scenario in which the training and test images are very well-aligned, but do contain significant variations in illumination. We will see how in this setting, face recognition can be naturally cast as a sparse representation problem, and solved via l_1-minimization. In subsequent sections, we will see how this core formulation extends naturally to handle other variabilities in real-world face images.

A basic problem in object recognition is to use labelled training samples from k distinct object classes to correctly determine the class to which a new test sample belongs. We arrange the given n_i training samples from the i^{th} class as columns of a matrix $A_i = [v_{i,1}, v_{i,2}, \ldots\ldots\ldots, v_{i,ni}] \in \mathbb{R}^{m \times n_i}$. In the context of face recognition, we will identify a $w \times h$ grayscale image with the vector v $\in \mathbb{R}^m$(m = w*h) given by stacking its columns; the columns of A_i are then the training face images of the i^{th} subject.

An immense variety of statistical, generative or discriminative, models have been proposed for exploiting the structure of the *Ai* for recognition. One particularly simple and effective approach models the samples from a single class as lying on a linear subspace. Subspace models are flexible enough to capture much of the variation in real datasets, and are especially well-motivated in the context of face recognition, where it has been observed that the images of faces under varying lighting and expression lie on a special low-dimensional subspace [24] often called a *face subspace.*

4.2 **Sparse Representation Methodology:**

Given sufficient training samples of the *i*-th object class, $[v_{i,1}, v_{i,2}, \ldots\ldots\ldots, v_{i,ni}] \in \mathbb{R}^{m \times n_i}$, any new (test) sample $\boldsymbol{y} \in \mathbb{R}^m$(from the same class will approximately lie in the linear span of the training samples associated with object *i*:

$$y = \alpha_{i,1} v_{i,1} + \alpha_{i,2} v_{i,2} + \ldots\ldots\ldots\ldots + \alpha_{i,n} v_{i,n} \quad (4.1)$$

for some scalars $\alpha_{i,j} \in \mathbb{R}$, $j = 1, 2, \ldots, n_i$.

Since the membership *i* of the test sample is initially unknown, we define a new matrix *A* for the entire training set as the concatenation of the *n* training samples of all *k* object classes:

$$A = [A_1, A_2, A_3, \ldots, A_k] = [v_{1,1}, v_{1,2}, \ldots, v_{k,n_k}] \quad (4.2)$$

Then the linear representation of *y* can be rewritten in terms of all training samples as

$$y = Ax_0 \in \mathbb{R}^m \quad (4.3)$$

where $x_0 = [0, \cdots, 0, v_{i,1}, v_{i,2}, \ldots\ldots\ldots, v_{i,ni}, 0, \ldots, 0]^T \in \mathbb{R}^n$ is a coefficient vector whose entries are zero except those associated with the *i*-th class. As the entries of the vector ***x***0 encode the identity of the test sample ***y***, it is tempting to attempt to obtain it by solving the linear system of equations ***y*** = *A**x***

Obviously, if *m* > *n*, the system of equations ***y*** = *A**x*** is over determined and the correct x_0 can usually be found as its unique solution. We will see in Section III, however, that in robust face recognition, the system ***y*** = *A**x*** is typically underdetermined, and so its solution is not unique. Conventionally, this difficulty is resolved by choosing the minimum *l*2-norm solution

$$(l_2): \quad \hat{x}_2 = \arg\min ||x||_2 \text{ subject to } Ax = y. \quad (4.4)$$

While this optimization problem can be easily solved (via the pseudoinverse of A), the solution $\hat{x}_2$ is not especially informative for recognizing the test sample y. is generally dense, with large nonzero entries corresponding to training samples from many different classes. To resolve this difficulty, we instead exploit the following simple observation: A valid test sample y can be sufficiently represented using only the training samples from the same class. This representation is naturally sparse if the number of object classes k is reasonably large. For instance, if k = 20, only 5% of the entries of the desired x0 should be nonzero. The more sparse the recovered x0 is, the easier will it be to accurately determine the identity of the test sample y.

$$(l_0): \quad \hat{x}_0 = \arg\min ||x||_0 \text{ subject to } Ax = y. \quad (4.5)$$

where ||·||0 denotes the *l*0-norm, which counts the number of nonzero entries in a vector. In fact, if the columns of *A* are in general position, then whenever ***y*** = *A**x*** for some ***x*** with less than *m/*2 nonzero, ***x*** is the unique sparsest solution: $\boldsymbol{\hat{x}_0} = \boldsymbol{x}$. However, the problem of finding the sparsest solution of an underdetermined system of linear equations is NP-hard, and difficult even to approximate.

4.2.1 Sparse Solution via l1-Minimization:

Recent development in the emerging theory of *sparse representation and compressed sensing* reveals that if the solution x_0 sought is *sparse enough*, the solution of the *l*0-minimization problem (4.5) is equal to the solution to the following *l*1-minimization problem:

$$(l_1): \quad \hat{x}_1 = \arg\min ||x||_1 \text{ subject to } Ax = y. \tag{4.6}$$

This problem can be solved in polynomial time by standard linear programming methods [16]. Even more efficient methods are available when the solution is known to be very sparse. For example, homotopy algorithms recover solutions with *t* nonzero in *O*(*t*3 + *n*) time, linear in the size of the training set[17].

Dealing with Small, Dense Noise:

So far, it has been assumed that equation (3) holds exactly. Since real data are noisy, it may not be possible to express the test sample exactly as a sparse superposition of the training samples. The model (3) can be modified to explicitly account for small, possibly dense noise, by writing

$$y = Ax_0 + z \tag{4.7}$$

where $z \in \mathbb{R}^m$ is a noise term with bounded energy $||z||_2 < E$. The sparse solution ***x***0 can still be approximately recovered by solving the following *stable* l_1-minimization problem:

$$(l_{1s}): \quad \hat{x}_1 = \arg\min ||x||_1 \text{ subject to } ||Ax - y||_2 \leq \varepsilon \tag{4.8}$$

Algorithm 1 : Sparse Representation-based Classification(SRC):

1: Input: a matrix of training samples $A = [A_1, A_2, A_3, \ldots, A_k] \in \mathbb{R}^{m \times n_i}$ for *k* classes, a test sample $y \in \mathbb{R}^m$, (and an optional error tolerance $\varepsilon > 0$.)

2: Normalize the columns of *A* to have unit l_2-norm.

3: Solve the l_1-minimization problem: $\hat{x}_1 = \arg\min ||x||_1$ subject to $Ax = y$. (Or alternatively, solve $\hat{x}_1 = \arg\min ||x||_1$ subject to $||Ax - y||_2 \leq \varepsilon$.

4: Compute the residuals $r_i(y) = ||y - A\,\delta_i(\hat{x}_1)||$
For *i* = 1,2,……, *k*.

5: Output: identity(*y*) = arg min*i* $r_i(y)$.

4.2.2 Validation Based on Sparse Representation:

Before classifying a given test sample, we must first decide if it is a valid sample from one of the classes in the dataset. The ability to detect and then reject invalid test samples, or “outliers,” is crucial for recognition systems to work in real-world situations. A face recognition system, for example, could be given a face image of a subject that is not in the database, or an image that is not a face at all. A valid test image should have a sparse representation whose nonzero entries concentrate mostly on one subject, whereas an invalid image has sparse coefficients spread widely among multiple subjects. To quantify this observation, we define the following measure of how concentrated the coefficients are on a single class in the dataset.

Sparsity Concentration Index*:* The *sparsity concentration index* (SCI) of a coefficient vector ***x*** $\in$ R*n* is

$$\mathrm{SCI}(x) = \frac{k.max_i||\delta_i(\hat{x}_1)||/||\mathrm{x}||-1}{k-1} \qquad (4.9)$$

For a solution $\hat{x}$ found by Algorithm 1, if SCI($\hat{x}$) = 1, the test image is represented using only images from a single object, and if SCI($\hat{x}$) = 0, the sparse coefficients are spread evenly over all classes. We choose a threshold $\tau \in (0,1)$ and accept a test image as valid if

$$\mathrm{SCI}(\hat{x}) \geq \tau,$$

and otherwise reject as invalid. One may choose to output the identity of ***y*** only if it passes this criterion.

4.3 Feature Extraction:

In the computer vision literature, numerous feature extraction schemes have been investigated for finding projections that better separate the classes in lower-dimensional spaces, which are often referred to as *feature spaces*. One class of methods extracts holistic face features, such as Eigenfaces, Fisherfaces , and Laplacianfaces. Another class of methods tries to extract meaningful partial facial features (e.g., patches around eyes or nose).

Recently various methods for a local feature extraction emerged. To avoid the high-dimensionality of the input data only local regions of an image are described, the extracted features are (hopefully) more robust against partial occlusion, illumination and small sample size. Algorithms used for a local feature extraction are Gabor Wavelets, Discrete Cosine Transform and Local Binary Patterns.

4.3.1 Local Binary Patterns:

The idea is to not look at the whole image as a high-dimensional vector, but describe only local features of an object. The features you extract this way will have a low-dimensionality implicitly. We will observe the image representation we are given doesn’t only suffer from illumination variations. Think of things like scale, translation or rotation in images - your local description has to be at least a bit robust against those things. Just like SIFT, the

Local Binary Patterns methodology has its roots in 2D texture analysis. The basic idea of Local Binary Patterns is to summarize the local structure in an image by comparing each pixel with its neighbourhood. Take a pixel as centre and threshold its neighbours against. If the intensity of the centre pixel is greater-equal its neighbour, then denote it with 1 and 0 if not. You'll end up with a binary number for each pixel, just like 11001111. So with 8 surrounding pixels you'll end up with 2^8 possible combinations, called *Local Binary Patterns* or sometimes referred to as *LBP codes*. The first LBP operator described in literature actually used a fixed 3 x 3 neighbourhood just like this:

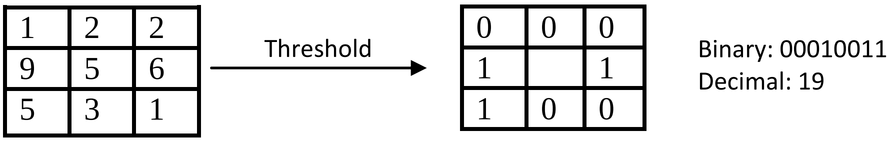

Fig 4.1 Local Binary Patterns

By definition the LBP operator is robust against monotonic gray scale transformations. We can easily verify this by looking at the LBP image of an artificially modified image as shown in fig [4.2]. To incorporate the spatial information in the face recognition model, the representation proposed by Ahonen et. al [16]is to divide the LBP image into m local regions and extract a histogram from each. The spatially enhanced feature vector is then obtained by concatenating the local histograms. These histograms are called *Local Binary Patterns Histograms*.

Another extension to the original operator is the definition of so-called *uniform patterns*, which can be used to reduce the length of the feature vector and implement a simple rotation-invariant descriptor. A local binary pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is traversed circularly. For example, the patterns 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010010 (6 transitions) are not. After the LBP labelled image $f_l(x,y)$ has been obtained, the uniform LBP histogram can be obtained by concatenation of histogram m local regions.

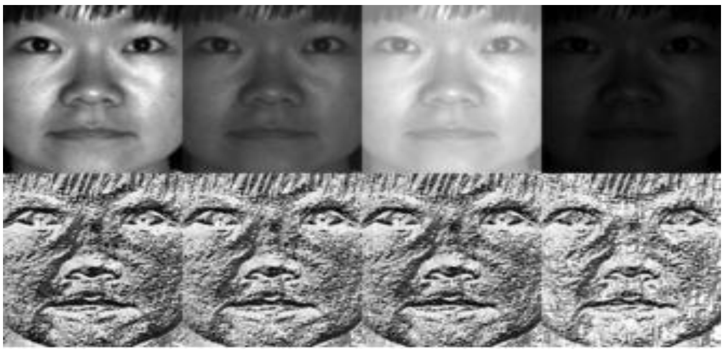

Fig 4.2 LBP Images under different Illuminations

4.4 Results and Conclusion:

Face Recognition Experiment is carried on Extended Yale B Database. The Extended Yale B database consists of frontal-face images of 38 individuals. The cropped and normalized 192 × 168 face images were captured under various laboratory-controlled lighting conditions [59]. For each subject, we randomly select half of the images for training (i.e., about 32 images per subject), and the other half for testing. SRC using down-sample images and Uniform LBP histograms is implemented and recognition rates are obtained as shown in Fig 4.3.

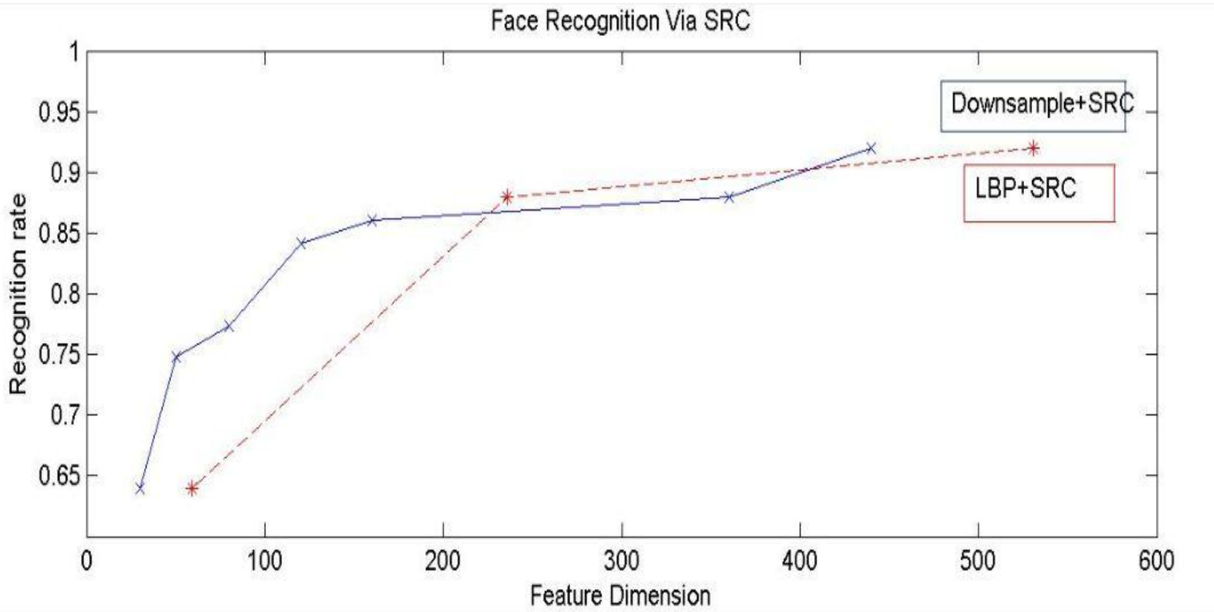

Fig 4.3 Face Recognition Via SRC

We have tested SRC using uniform local binary patterns (LBP) histograms. We compared our results with SRC using down-sample images. Recognition rate of 0.92 is obtained for the feature dimension of 440 with down-sample images. In the case of SRC with Uniform Local Binary patterns histogram, recognition rate of 0.92 is obtained for feature dimension of 531. We observed that linear structure due to illumination is not critical with Uniform LBP histograms. We can have linear structure with other aspects like facial expressions if we use Uniform LBP histograms.

Chapter 5

User Authentication Module

5.1 Multi-Biometric Authentication

In recent times, most of the computer systems authenticate users using mainly passwords, which can be easily hacked by any hacker and can cause exposure to confidential or secure data. Most of the identity thefts are committed by employees working in office or insiders. These kinds of attacks are known as insider attacks. For more secure systems using physiological biometrics like fingerprints or iris pattern or face. However, these kinds of systems involving physiological biometric verification require high cost hardware devices.

In case of behavioural biometric these hardware devices are not required. However, the only problem with them is that they are highly variable with time. However, the behaviour of a person can be accurately identified by the behavioural biometric system and they are very hard to imitate or copy by any hacker. The trade-off between using physiological and behavioural biometrics can be overcome by using a system with multi-modal biometrics involving both the type of biometrics. This idea suggested us to use an innovative method of secure login system which can detect an intruder based on three biometrics: face, keyboard and mouse dynamics.

5.2 User Authentication Module Design:

Our User Authentication module is based on two biometrics; Face and Keyboard dynamics. Authentication module using Keyboard dynamics is developed by one of our senior student [19].We have integrated our Face Recognition module with that module as shown in Fig 5.1.

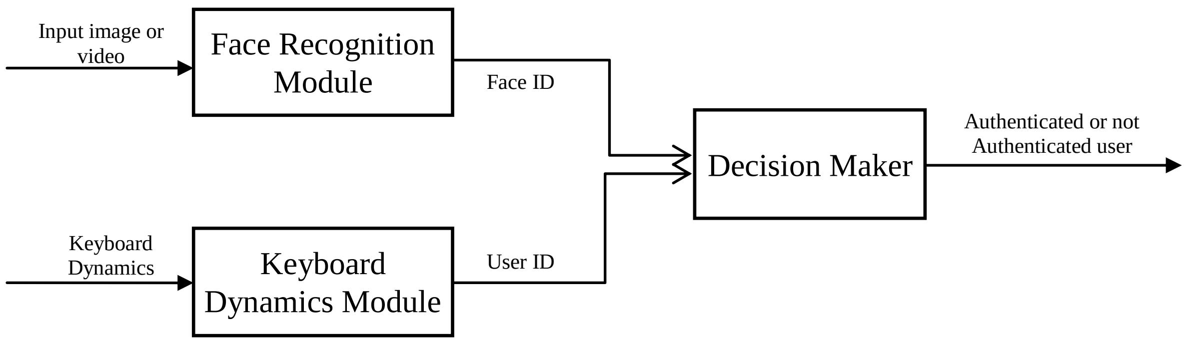

Fig 5.1 User Authentication Flow.

Overall system consists of two modules, authentication module using face recognition authentication module using keyboard dynamics. Face recognition module returns Face ID if he/she is an authenticated user otherwise returns ‘unknown’. Similarly, Keyboard dynamics authentication module returns User ID if he/she is an authenticated user otherwise returns ‘unknown’. Decision maker block will authenticate user if one of the (User ID of Face ID) decision is correct, otherwise it will allow user to login. The application developed based on these two authentication modules is as shown in fig 5.2.Overall authentication accuracy is depends on the decision made by each of the authentication modules.

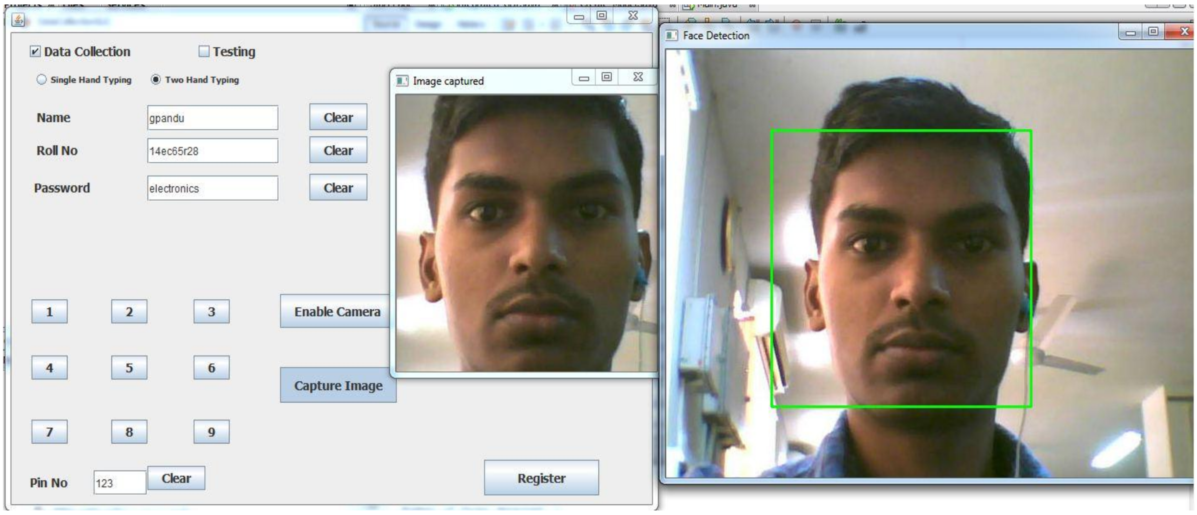

Fig 5.2 Application for User Authentication.

This application operates in two modes:

- Data collection
- Testing

In the initial stage of the development, the application would be working in Data collection mode, to acquire data for training and authentication. In the process of data collection, user needs to provide data at least three times in order get sufficient training data. Testing is accomplished by one step process of entering his/her details and capture of ten images. Images should contain frontal faces for both training and testing as our developed Face Recognition module works only for frontal faces. We have tested the application with standard database of Face Recognition. This application can be used to for taking attendance of students in a college.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, User Authentication module using Face Recognition is developed. The Face Recognition module is integrated with the other Authentication module (created using Keyboard Dynamics). The overall module can be used as student attendance system.

Face detection is implemented using Sliding window method. The HOG descriptors extracted over sliding windows over face and non-face patches are used for train SVM classifier. True positive rate (Recall) of 0.84 is obtained on test images of FDDB database with 5 false positives. Optimal threshold value for Regularization parameter C is obtained as 60. Detection rate using HOG descriptor is 5 frames per second. True positive rate for the detection can be increased with high training data. Detector rate can be increased using Dimensionality reduction but it could affect the detection accuracy.

We have implemented Face Recognition via Sparse Representation based Classification (SRC). Linear structure of samples of a subject is important in the process of sparse representation. This linear structure can be due to illumination or expressions. We have tested SRC using uniform local binary patterns (LBP) histograms. We compared our results with SRC using down-sample images. The results are as good SRC using down-sample images. We observed that linear structure due to illumination is not critical with Uniform LBP histograms. We can have linear structure with other aspects like facial expressions.

Sometimes, variation of keystroke typing behaviour with varying emotions and other factors may lead to the rejection of already registered user. As we have integrated both biometrics, Face Recognition can alone be used for authentication in such cases.

The limitation of Face Recognition module is that it works only for frontal faces. Also, linear structure of samples needs be retained with one of the aspects like illumination or facial expressions although it is not much critical with SRC.

6.2 Future Work

The User Authentication module can be used to collect details of students for authentication in colleges. Emotion detection using Face expression Recognition can be implemented in future which will help in understanding of employee's emotion or state at work place. This will also help in understanding the emotional state of the student while attending classes. Emotion detection using facial expressions can be used to understand medical condition of a patient as well. Future work can be extended to study the keyboard dynamics pattern of user with emotion if both biometrics are obtained together.

Bibliography

[1] Vaclav Matyas Zdenek Riha. Biometric authentication systems. 2000.

[2] RK Das, S Mukhopadhyay, P Bhattacharya, et al. Continuous multimodal biometric authentication for pc and handheld devices. IETE Journal of Education,52(2):59, 2011.

[3] Kenneth Revett. Behavioral biometrics: a remote access approach. Wiley, 2008.

[4] Zhengming Li; Lijie Xue; Fei Tan, "Face detection in complex background based on skin color features and improved AdaBoost algorithms", Progress in Informatics and Computing (PIC), 2010 IEEE International Conference on , vol.2, no., pp.723,727, 10-12 Dec. 2010.

[5] P. Viola, and M. J. Jones, “Robust Real-Time Face Detection”, Int. Journal of Computer Vision,Vol.57, No.2, 2004, pp.137-154.

[6] A. Z. Kouzain, F. He, and K. Sammut, “Commonsense Knowledge-Based Face Detection”, Proc. ofIEEE INSE, 1997, pp.215-220.

[7] *Kanade, T.* Picture processing system by computer complex and recognition of human *faces.* PhD thesis, Kyoto University, November 1973.

[8] Brunelli, R., Poggio, T. *Face Recognition through Geometrical Features.* European Conference on Computer Vision (ECCV) 1992, S. 792–800.

[9] Turk, M., and Pentland, A. *Eigenfaces for recognition.* Journal of Cognitive Neuroscience 3 (1991), 71–86.

[10] Belhumeur, P. N., Hespanha, J., and Kriegman, D. *Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection.* IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 7 (1997), 711–720.

[11] Ahonen, T., Hadid, A., and Pietikainen, M. *Face Recognition with Local Binary Patterns.* Computer Vision - ECCV 2004 (2004), 469–481.

[12] N. Dalal and B. Triggs. “Histograms of oriented gradients for human detection”In Computer Vision and Pattern Recognition(CVPR), pages 886–893, 2005.

[13] Cortes, Corinna, and Vladimir Vapnik. "Support vector machine." *Machine learning* 20.3 (1995): 273-297.

[14] T. Serre, “Learning a dictionary of shape-components in visual cortex: Comparison with neurons, humans and machines,” Ph.D. dissertation, MIT, 2006.

[15] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma.Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):210–227, 2009.

[16] S. Chen, D. Donoho, and M. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[17] D. Donoho and Y. Tsaig, “Fast solution of `1-norm minimization problems when the solution may be sparse,” preprint, *http://www.stanford.edu/ tsaig/research.html*, 2006.

[18]Ahonen, T., Hadid, A., and Pietikainen, M. *Face Recognition with Local Binary Patterns.* Computer Vision - ECCV 2004 (2004), 469–481.

[19] Rahul Arora, Sudipta Mukhopadhyay, “*User Authentication Using Multimodal Biometrics: Keystroke Dynamics*”, M.Tech Thesis, Dept E&ECE, IIT Kharagpur, Kharagpur, West Bengal,2014.

[20] M. R. Mohammadi, E. Fatemizadeh, and M. H. Mahoor, “Simultaneous recognition of facial expression and identity via sparse representation,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Steamboat Springs, CO, USA, 2014, pp. 1066–1073.