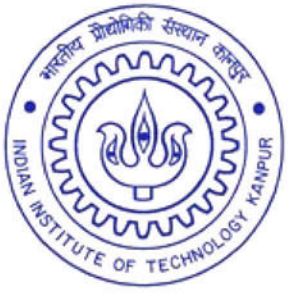
Integration of Offline and Online Signature Verification Systems

by

Deepthi Uppalapati



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

July 2007

Integration of Offline and Online Signature Verification Systems

A Thesis Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Master of Technology

by

Deepthi Uppalapati

to the

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

July 2007

CERTIFICATE

It is certified that the work contained in the thesis entitled "*Integration of Offline and Online Signature Verification Systems*" by *Deepthi Uppalapati* has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

3rd July, 2007

Phalguni Gupta
Department of Computer Science & Engineering,
Indian Institute of Technology Kanpur,
Kanpur-208016.

Abstract

Biometrics are automated methods of recognizing a person based on a physiological or behavioral characteristics. The features measured can be anyone or a combination of face, fingerprints, hand geometry, iris, retina, handwriting, signature and voice. Biometric technologies are becoming the foundation of highly secure identification and personal verification solutions. The need for biometrics can be found in central, state and local governments, in the military, and in commercial applications.

Signature is a behavioral trait of an individual and forms a special class of handwriting in which legible letters or words may not be exhibited. Depending on the hardware front-end, a signature verification system can be classified as either offline or online. Offline signature recognition is performed after the writing is complete. The data is captured at a later time by using an optical scanner to convert the image into a bit pattern. The features thus extracted are said to be static. Online signature recognition, in contrast, means that the machine recognizes the handwriting as the user writes. It requires a transducer that captures the signature as it is written and hence the features are dynamic in nature.

In this thesis, an integrated signature verification system has been proposed in which the feature vector comprises of both static and dynamic features. The integrated signature verification system incorporates database management, noise removal and pre-processing, feature extraction, learning and verification modules. An integrated version not only provides a way to compare and match an offline signature against an online one and vice-versa, but also improves the system performance in those cases where both static and dynamic features are available.

Contents

1 Introduction **1**

1.1 Introduction to Signature based Verification System 1

1.2 Motivation and Problem Statement 2

1.3 Thesis Organization 3

2 Background and Related Work **4**

2.1 Offline Signature Verification System 4

2.2 Online Signature Verification System 5

3 The Proposed Approach **7**

3.1 Database Management 8

3.2 Noise Removal and Pre-processing 9

3.2.1 Noise Removal 10

3.2.2 Image Extraction 10

3.3 Feature Extraction and Parameter Calculations 13

3.3.1 Parameter Calculation 13

3.3.2 Global Feature Extraction 14

3.3.3 Local Feature Extraction 16

3.3.4 Dynamic Feature Extraction 17

3.4 Learning 18

3.5 Verification 18

3.5.1 Matching 20

3.5.2 Decision Making 23

4 Results And Discussion 24

5 Conclusions and Future Scope 32

List of Tables

3.1 Weights used in Offline to Offline Matching 21
3.2 Weights used in Online to Online Matching 21
3.3 Weights used in Online to Offline Matching 22
3.4 Weights used in Offline to Online Matching 22

4.1 Feature Vectors of Offline and Online images for id = 8 26

List of Figures

3.1 Modular structure of an integrated signature verification system . . 9

3.2 Examples of Cross (C1, C2, C3, C4) and Edge (E1, E2, E3, E4) points 15

3.3 Connected Components of a Signature 16

3.4 Masks for obtaining slant features from thinned image 16

4.1 Offline Signature and Online Signature Images for id = 8 25

4.2 Offline to Offline Verification Curves 28

4.3 Online to Online Verification Curves 29

4.4 Online to Offline Verification Curves 30

4.5 Offline to Online Verification Curves 31

Chapter 1

Introduction

Biometric automatic identification or authentication of individuals is essentially a pattern recognition system based on the physical or behavioral characteristics of individuals. The characteristics that are captured essentially need to be:

(a) **Universal**. Every person must possess the characteristic. It must be one that is seldom lost to accident or disease.

(b) **Invariant**. It should be constant over a long period of time.

(c) **Singular**. It must be unique to the individual

(d) **Inimitable**. It must be irreproducible by other means.

(e) **Reducible and Comparable**. It should be capable of being reduced to a format that is easy to handle and digitally comparable to others.

(f) **Reliable and Tamper-resistant**. It should be impractical to mask or manipulate.

The various physiological characteristics that satisfy the above requirements are face, iris, fingerprints, palm prints, hand geometry and the behavioral characteristics that include signature, voice and keystroke patterns.

1.1 Introduction to Signature based Verification System

A signature is treated as an image carrying a certain pattern of pixels that pertains to a specific individual. Signature verification problem, therefore, is con-

cerned with examining and determining whether a particular signature truly belongs to a person or not. Signatures are a special case of handwriting in which special characters and flourishes are viable. Signature verification is a different pattern recognition problem as no two genuine signatures of a person are precisely the same. The difficulty also stems from the fact that skilled forgeries follow the genuine pattern unlike fingerprints or irises which vary widely for two different persons. Ideally, interpersonal variation should be much more than intrapersonal variation. Therefore, it is important to identify those features which minimize the intrapersonal variation and maximize interpersonal variations. The key factor is to differentiate between the parts of the signature that are habitual and those that vary with almost every signature. Disadvantages include problems of long-term reliability, lack of accuracy and cost.

Signature verification system can be classified as:

- Offline Signature Verification System
- Online Signature Verification System

In offline verification systems, only static features are considered, whereas in case of online systems, dynamic features are taken into consideration. Dynamic features include the time that the stylus is in and out of contact with the paper, the total time taken to make the signature and the position where the stylus is raised from and lowered onto the paper, number of breakpoints, maximum/minimum pressure of stylus contact, speed, direction at crucial points.

1.2 Motivation and Problem Statement

Due to the relative ease of use of an offline system, a number of applications worldwide prefer to use this system (e.g. cheque verification in a bank). Online signature recognition systems are more reliable (closer to 99%) as compared to offline systems in terms of accuracy. However, online methods require some special hardware like digitizers, pressure sensitive tablets to capture the dynamic features, which the offline methods do not require.

There may be a case where the type of signature verification system used for training differs from that used for testing purpose. Though the test sample is of a genuine person, it might not be possible to prove with either of these systems alone. Hence, development of an integrated version of offline and online signature verification systems would be useful, which is the main motivation for this thesis. At the time of data acquisition, either or both the offline and the online signature templates of the person being registered are recorded and an identification number is generated for that person. During testing, the test sample recorded is matched against the information available for that identification number in the database. A reasonably high accuracy of verification has been achieved when the static features are also taken into consideration for the online recognition systems.

1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 provides an overview of the existing offline and online signature recognition systems. Chapter 3 discusses the details of the proposed approach. In this chapter, the major modules of the integrated signature recognition system: database management, noise removal and pre-processing, feature extraction and parameter calculations, learning and verification are presented and detailed. The subsequent chapter discusses the details of measuring the performance of a biometric system, the evaluation of the proposed approach and the results achieved. Chapter 5 concludes the thesis and presents the scope for future work.

Chapter 2

Background and Related Work

Signatures form a special class of handwriting in which legible letters or words may not be exhibited. They provide secure means for authentication, attestation and authorization in legal, banking or other high security environments. Signature verification problem pertains to determining where a particular signature is verily written by a person so that forgeries can be detected. Based on the hardware front-end, a signature verification system can be classified as either offline or online. A comprehensive survey on online and offline handwriting recognition has been done by Plamondon [16].

2.1 Offline Signature Verification System

In this system, a user writes his signature on a piece paper, it is then digitized through an optical scanner or a camera, and the biometric system recognizes the signature analyzing its shape. The static information derived from such signatures is used for matching and decision making.

A lot of work has been done in developing offline signature verification and recognition systems with variations in feature extraction and matching. Geometric feature extraction with neural network classification has been used [8]. Ramesh and Murty have implemented four different types of pattern representation schemes, viz., geometric features, moment-based representations, envelope characteristics and tree-structured wavelet features [17]. In their approach, the

individual feature components have been weighed by their pattern characterization capability using Genetic Algorithms. Ismail and Gad have developed a system of two separate phases for signature recognition and verification [9]. For recognition, a multistage classifier and a combination of global and local features have been used. And for verification, algorithms based on fuzzy concepts are used. An offline signature verification system based on fusion of local and global information has been developed [6].

2.2 Online Signature Verification System

This system uses the digitized signature of an individual. This signature is acquired in real time [19]. Another possibility is the acquisition by means of stylus-operated PDAs. These tablets capture dynamic information like position in x-axis, position in y-axis, pressure applied by the pen, angle of the pen with respect to the tablet. Using this set of dynamic data, further information can be inferred, such as acceleration, velocity, instantaneous trajectory angle, instantaneous displacement, tangential acceleration etc.

The use of dynamic features makes it harder to forge. Even if a skilled forger is able to copy the shape of the signature, it is very unlikely that he can simultaneously reproduce all the dynamic features as well. Online signature verification scheme extracts features from the signature, which characterize the signature. The feature statistics of a training set of genuine signatures are used to generate a model, which is further used for testing. Selecting a good feature set to represent the model is a very important part for a verification scheme. There are mainly three direct approaches for the selection of features. The first one is based on point-to-point local feature relating position, velocity, acceleration, pressure etc. The second one deals with the global features like writing time, pen up time, number of breakpoints, maximum/minimum pressure and speed, pen direction at crucial points like starting and ending points. The third one deals with the shape of the signature. There are several methods using local features in signature verification. The most commonly used strategies are matching by Dynamic Warping [13]

and by using Hidden Markov Model (HMM) [20]. Dynamic Warping approaches give a flexible matching of the local features. A HMM performs stochastic matching of a model and a signature using a sequence of probability distributions of the features along the signature. The Dynamic Warping approach was incorporated to match the local features, mainly because learning techniques like HMM requires many test data, which does not suit the condition. In case of global feature verification, a number of global features are extracted and then compared with reference signature features. The point-to-point local feature comparison is more sensitive to handwriting variations than other approaches but is also more resource intensive. Selection of features depends on their variation tolerance quality. For example, the direction, speed, acceleration features are rotation variant. Hence, if these are used for verification, the signature sample has to be rotated to a fixed slant before extracting these features. Similarly, the displacement, area features are size dependent and would require the signature to be transformed to a fixed size before extracting them.

There has been a lot of work involved in developing online signature verification systems. The comparison algorithms used in online systems are mostly dynamic time warping and regional correlation. Those systems have reported success in signature verification [12, 15]. Shafiei and Rabiee have applied the HMM approach [18], where one segments a signature based on its perceptually important points and then computes for each segment a number of features that are scale and displacement invariant. The resulted sequence is then used for training the HMM to achieve signature verification. They achieved a false acceptance rate (FAR) of 4% and a false rejection rate (FRR) of 12% on there database, which includes 622 genuine signatures and 1010 forgery signatures collected from 69 individuals. Other techniques like Fourier transform [11], extreme points warping [5] have also been applied for signature verification. Recently, a combination of vector quantization and dynamic time warping has been proposed [4] for online signature recognition.

Chapter 3

The Proposed Approach

The proposed integrated signature verification system takes a gray scale query image and identification number as input. It verifies whether the input image matches with the training signature images in the database against the given identification number. The system can be broadly categorized on the basis of the type of test sample (offline or online) and the one in the training database. For instance, if an online test sample is being matched against the offline training database, then it is called online to offline verification. The algorithm used for the implementation of integrated signature verification system consists of the following five major modules:

- **Database Management**. This module handles the process of data acquisition and the maintenance of the signature images and the learned feature vector for each identification number.
- **Noise Removal and Pre-processing**. This module involves removing noise like spurious pixels in case of offline signatures or signals in case of online signatures, smoothening, space standardization and normalization, converting a gray scale image to a binary image, extraction of the high pressure region in an image, etc.
- **Feature Extraction and Parameter Calculations**. Features can be broadly classified as static features and dynamic features. Static features can be further classified into two types-global and local features, where global features

are 'characteristics which identify or describe the signature as a whole' [9] and local features are confined to a limited portion (eg. a grid) of the signature. The width and height of individual signature components, width to height ratio, total area of black pixels in the binary and high pressure region (HPR) images, horizontal and vertical projections of signature images, baseline, baseline shift, relative position of global baseline and centre of gravity with respect to width of the signature, number of cross and edge points, slant, connected components of the signature are examples of global features. On the other hand, examples of local features are the same as that of the global features except that they are calculated for each of the number of grids that the signature has been divided into, say, three equal parts. The center of gravity, global base line and also the number of black pixels of each part are extracted separately. The speed, pressure, the pen inclination, the time taken to sign are some examples of dynamic features.

- **Learning**. This module uses the extracted features to calculate the mean and standard deviation for each of the feature. These values are placed as a vector and stored in the database against the entered identification number. The higher the number of training samples, the higher would be the accuracy.
- **Verification**. This module compares the different features obtained from the test image given to signature verification system with the features stored in the database against the given identification number. Based on this comparison, it either accepts the signature instance as being genuine or rejects it.

This chapter discusses each of the above modules in detail. The modular structure of an integrated signature verification system is shown in the Figure 3.1.

3.1 Database Management

This module handles the various aspects of database management like creation, modification, deletion and training for a signature instance. Data acquisition of the static features is carried out using high resolution scanners. And the dynamic

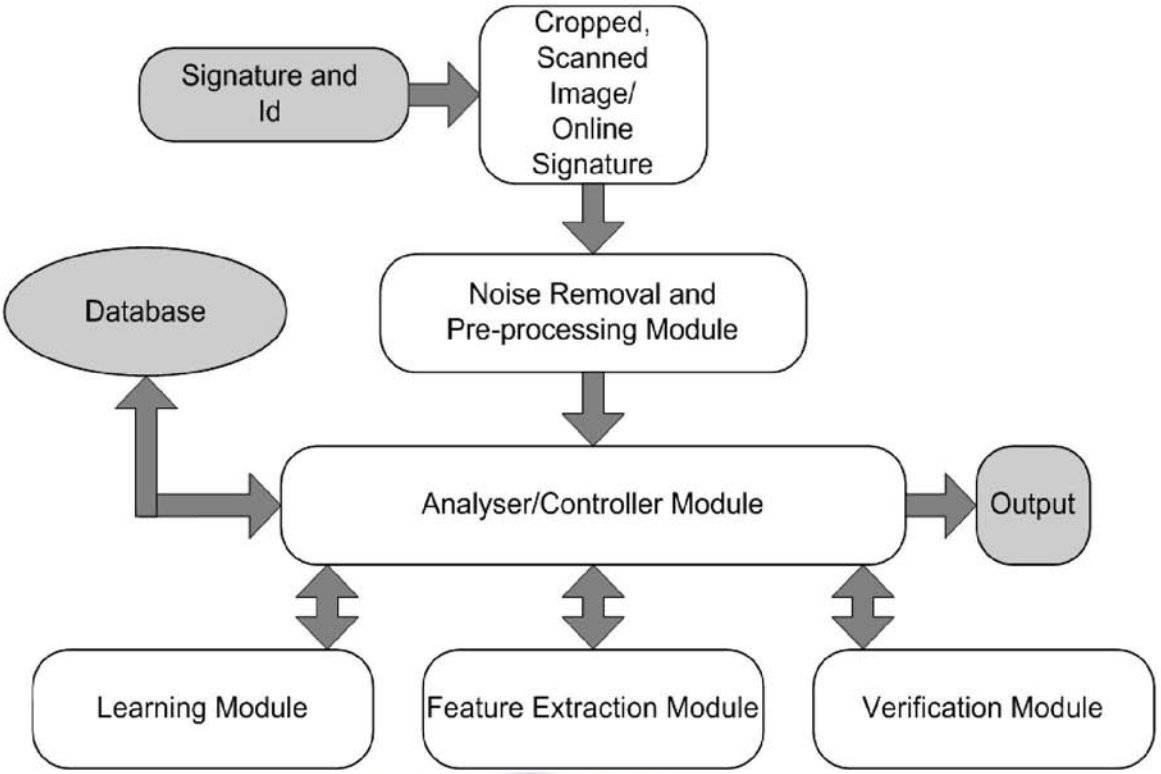


Figure 3.1. Modular structure of an integrated signature verification system

features are acquired using special devices called transducers or digitizers. The information regarding a particular signature is stored in the database as a feature vector where all the static features are stored against the identification number. For signatures captured by the digital signature pad, the feature vector comprises of both static and dynamic features. The feature vectors comprising of mean, standard deviation or median for a given identification number is stored in the database after learning.

3.2 Noise Removal and Pre-processing

This module is divided into two sub-modules: noise removal and image extraction. The query image is first passed to the noise removal module. The noise removal module returns a noise free image of the same size as the input image.

The image extraction module then extracts the binary, high pressure region (HPR), and thinned images which are used for feature extraction by the feature extraction module. After the binary image extraction, the horizontal and vertical

projections of the image are used to extract the area of interest from the image. The area of interest is the region where maximum and pertinent information is present. The area of interest is extracted from the HPR and thinned images as well.

3.2.1 Noise Removal

For noise removal, a median filter is used followed by the mean filter, each of which is described below:

- *Median filter* replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel, and is given by the expression:

$$f(x,y) = median\{g(s,t)\} \tag{3.1}$$

  where $g(s,t) \in S_{xy}$, *f(x,y)* is the noise removed image, $\{g(s,t)\}$ are the immediate eight neighbors of *I(x,y)* and S_{xy} is the sliding window over the image pixels.

- *Mean filter* replaces the value of a pixel by the average or mean value of eight neighbor pixel values and is given by the expression:

$$f(x,y) = \sum_{i=0}^{8} (g(s_i,t_i))/9 \tag{3.2}$$

  where *f(x,y)* is the noise removed image, $(g(s_i,t_i))$ are the immediate eight neighbors of *I(x,y)*.

3.2.2 Image Extraction

The image extraction module is used for extraction of the following three images:

- Binary image
- High pressure region (HPR) image
- Thinned image

The area of interest is extracted in this module using the projections of the binary image. The image extraction module removes the extra white spaces in the image to reduce its size which inturn reduces the time required by the feature extraction module. The algorithm for this module is described in **Algorithm 1**.

Algorithm 1

Step 1: Let the input noise-free image be called I1.

Step 2: Obtain the binary image B1 from I1.

Step 3: Obtain the horizontal and vertical projections of the images I1 and B1.

Step 4: Isolate the regions of interest from the binary image B1 (call it B2) and the input image I1 (call it I2) using the horizontal and vertical projections obtained in Step 3.

Step 5: Obtain the thinned image T1 from B2.

Step 6: Obtain the HPR image HPR1 from I2.

Binary Image Extraction. The binary image is obtained using a threshold which varies from signature to signature. The algorithm for binary image extraction is described in **Algorithm 2**.

Algorithm 2

Step 1: Get the gray level value for each pixel of image.

Step 2: Find the number of pixels having the same gray level value and sort the gray levels based on the count.

Step 3: Find the average of the first twenty most occurring intensity values and set this as threshold.

For each of the pixels, if the pixel value is less than the threshold, set the value of that pixel to black, otherwise set it to white.

Region of Interest. Region of Interest is obtained from the input image by removing the blank areas without any sufficient data. The final image so extracted is rich in signal to noise ratio. The algorithm to extract the region of interest for an input image is described in **Algorithm 3**.

Algorithm 3

Step 1: Calculate the horizontal and vertical projections of the image.

Step 2: Flag the index positions from the right and the left side of the signature image where the horizontal projection has just crossed a threshold. Call them indexH1 and indexH2.

Step 3: Repeat the process from top and bottom. Let the index be indexV1 and indexV2.

Step 4: Obtain the image segment between (indexH1, indexV1), (indexH2, indexV1), (indexH1, indexV2) and (indexH2, indexV2). This is the required region of interest.

High Pressure Region Extraction. There are certain regions in the signature image where the signer gives extra stress. This occurs usually at the corners and at the beginning of a stroke. Since the more emphatic regions will have a darker area in the scan, a threshold T_{hpr} is set up. The value of T_{hpr} is set to be 80% mark between the max and min gray scale intensities. Pixels with gray scale values larger than this are considered to belong to high pressure regions.

$$T_{hpr} = g_{min} + 0.8 * (g_{max} - g_{min}) \tag{3.3}$$

and $P \in HPR$ if $I_p \geq T_{hpr}$

Thinning. Thinning is performed over the binary image to obtain a single pixel thick skeleton of the signature instance. This image is used to obtain a number of features later. The standard thinning algorithm using morphological operations [7] is used for this purpose.

Horizontal and Vertical Projections. *Horizontal projection* is the projection of the image along the horizontal axis and is given by:

$$P_h[y] = \sum_{x=1}^{n} black * pixel(x, y) \tag{3.4}$$

where n is height of the image, *black* is set to 1 for those pixels whose grayscale value is greater than threshold and zero otherwise, and *pixel(x,y)* is 1 for all x, y. *Vertical projection* is the projection of the image along the vertical axis and is

given by:

$$P_v[x] = \sum_{y=1}^{m} black * pixel(x, y) \tag{3.5}$$

where, m is width of the image, and *black*, *pixel(x,y)* are same as defined above for horizontal projection.

3.3 Feature Extraction and Parameter Calculations

This module extracts various features from three images passed to it by the noise removal and pre-processing module. This module performs the following operations:

(a) Parameter Calculations

(b) Global Feature Extraction

(c) Local Feature Extraction

(d) Dynamic Feature Extraction (only if the digital signature pad is used)

3.3.1 Parameter Calculation

The various parameters that are computed are as follows:

- Horizontal Projection. Horizontal projection is the projection of the image along the horizontal axis and is given by (3.4).
- Vertical Projection. Vertical projection is the projection of the image along the vertical axis and is given by (3.5).
- Centre of Gravity (CG). The centre of gravity of the image is calculated as follows:

  If X, Y be the x and y co-ordinates of the centre of gravity of the image, then,

$$X = \sum_{y=1}^{m} (y.P_h[y]) \,/\, \sum_{y=1}^{m} P_h[y] \tag{3.6}$$

  and

$$Y = \sum_{x=1}^{n} (x.P_v[x]) \,/\, \sum_{x=1}^{n} P_v[x] \tag{3.7}$$

where $P_h[y]$ and $P_v[x]$ are the horizontal and vertical projections as defined in (3.4) and (3.5) respectively.

- Global Baseline (GB). The global baseline parameter can be taken to be the median of the pixel distribution. Usually, the vertical projection of the binary image is over smoothened and then, if the projection has one maximum point (peak), GB corresponds to this peak, otherwise it is taken as halfway between the two outermost maximum points.
- Upper Extension (UEX). This is the maximum difference between the smoothened curve of the vertical projection and the approximated curve of the same projection above the baseline.
- Lower Extension (LEX). This is the maximum difference between the smoothened curve of the vertical projection and the approximated curve of the same projection below the baseline.
- Middle Zone (MZ). This is a critical region which will have the maximum amount of feature data.
- Width (W). It is defined as the distance between two points from either ends in the horizontal projection which contain more than three pixels of the binary image.
- Height (H). It is defined as the distance between two points from either ends in the vertical projection which contain more than three pixels of the binary image.
- Area (A). It is the total number of pixels in the thinned image.

3.3.2 Global Feature Extraction

The signature has a large number of features related to length like the length of the vertical projection or the length of the trace etc. All these parameters may have different measurements for different instances of the same signature, but, it has been observed that they have a constant relative value. The Height (H) of the signature is taken as the base and all the parameters are converted to their relative values. The following parameters are calculated in this manner:

- Width of the signature. Width to height ratio = W/H
- Global Baseline (GB). Relative position of the Global Baseline = GB/H
- Length of the trace. Length of trace to height ratio = L/H
- Area Normalization. Normalized Area = Area of black pixels in the image / (W * H)
- Number of Edge Points. An edge point is defined as a point having only one eight-neighbor in the thinned image. In Figure 3.2, E1, E2, E3, E4 are edge points.
- Number of Cross Points. A cross point is defined as a point having at least three eight-neighbors in the thinned image. In Figure 3.2, C1, C2, C3, C4 are corner points.

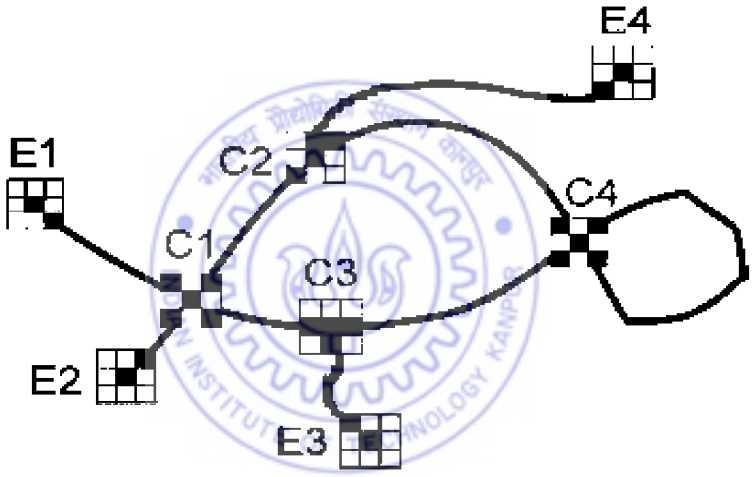


Figure 3.2. Examples of Cross (C1, C2, C3, C4) and Edge (E1, E2, E3, E4) points

- Number of Connected Components. This feature indicates the number of independent components in the signature. For example, there are six connected components in Figure 3.3.
- Global Slant Feature. The thinned image can be used for extracting this feature. Overall for a given non zero pixel $p(i, j)$ in the image, the non-zero $p(i-1, j+1)$, $p(i, j+1)$, $p(i+1, j+1)$ and $p(i+1, j)$ are the negatively, vertically, positively and horizontally slanted pixels. These pixels can be obtained using the following masks over the image:

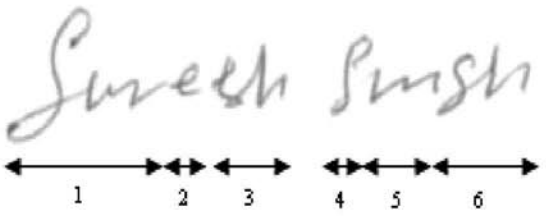


Figure 3.3. Connected Components of a Signature

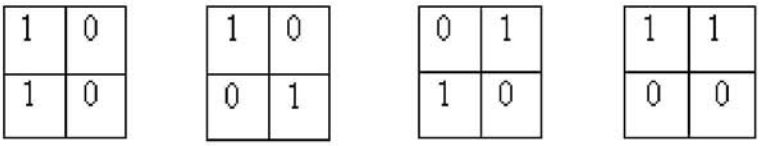


Figure 3.4. Masks for obtaining slant features from thinned image

3.3.3 Local Feature Extraction

The signature is divided into three equal horizontal zones. The local features extracted are baseline for each segment, CG in each segment of the binary and the HPR images and the pixel distribution for each segment.

The static feature values computed in the above three sub-sections can be organized as a total of 27 features which were extracted. They are:

- 1 - Width to height ratio
- 2, 3 - Center of gravity (both X and Y coordinates) to height ratio
- 4 - Global Baseline to height ratio
- 5 - Normalized area of black pixels
- 6 - Total number of components of the signature
- 7 - Upper extension to height ratio
- 8 - Lower extension to height ratio
- 9 - Number of cross points to area of black pixels in the thinned image
- 10 - Number of edge points to area of black pixels in the thinned image
- 11 - Slope of the thinned image
- 12 - Trace to area of black pixels in the thinned image

- 13, 14 - Center of gravity (both X and Y coordinates) of the HPR image to height ratio
- 15 - Area of black pixels in the HPR image to total area of black pixels in the image.
- 16 to 27 - Ratio of baseline position to height of the image, Ratio of centre of gravity co-ordinates to height and Ratio of pixel count of individual sections to total pixel count of the image in three horizontal sections.

3.3.4 Dynamic Feature Extraction

System Description of Digital Signature Pad.

The online recognition system uses a graphics tablet from Wacom as data acquisition device. It is Intuos2 (model XD-0405-R) with ps2 interface. The device captures about 100 points per second. The device provides the relative time value, X and Y coordinates, pressure and pen inclination information of each point captured by it.

The dynamic features are more difficult to imitate. The mostly widely used ones are: position, pressure, force, velocity, absolute and relative speed between two crucial points, acceleration.

The dynamic features include global features and local features. The first seven of those listed below are global features; the next three are local ones and the last one is a derived feature.

- Total time is calculated by subtracting the first contact instance of the pen and the pad from the last contact instance gives the total time taken during the signature.
- Pen down time
- Maximum speed
- Minimum speed
- Maximum pressure
- Minimum pressure

- Number of connected components
- Peak and valley variation Of X co-ordinates and Y co-ordinates
- Pen inclination (X tilt)
- Pen inclination (Y tilt)
- Down time ratio that is calculated by dividing the pen down time by total time.

3.4 Learning

In this module, the mean and standard deviation values for different static features are calculated. These values are later used in the verification module. The mean and standard deviation values are calculated for all the 27 features except for the 6^{th} and the 11^{th} features. In case of the 6^{th} feature (the number of connected components in the signature image as shown in Figure 3.3), the value for which there are maximum number of learning image instances is chosen instead of mean of the total number of components. The slant of the image which is the 11^{th} feature is calculated by taking the majority of the number of individual slant values (vertical, horizontal, positive and negative).

The mean and standard deviation values are stored as a feature vector in the database against the entered identification number. For verification purpose, only these values are sufficient and the learning set is not required. When a very large number of records have to be maintained with constraint on the memory size, the feature vector value is enough for storage.

3.5 Verification

Forgeries for handwritten signature can be classified into four classes- random, simple, skilled and traced. Random forgery is one in which the forged signature has a totally different semantic meaning and overall shape in comparison to the genuine signature. In simple forgery, the semantics of the signature are the same as that of the genuine signature, but the overall shape differs to a great extent as the

forger has no idea about how the signature is done. In skilled forgery, the forger has a prior knowledge about how the signature is written and practices it well before the final attempt of duplicating it. For traced forgery, a signature instance or its photocopy is used as a reference and is tried to be forged. The integrated signature verification is good in detecting the first two types of the forgeries but the results are not that encouraging in the case of latter two. If the dynamic features are available for both testing and training instances, then the accuracy is high, as much as 100% for the first two classes of forgeries and 99% in case of the latter.

The feature vectors containing the values for different features are given as input to this module for the signature instance which has to be verified. A total of 38 features are used which include 27 static features and 11 dynamic features for recognition purpose. The matching is performed using weighted Euclidean distance measure.

A person can have variations in his signature. And it is this variation in the signature images of the genuine signer and the forger which provides enough information to differentiate between the two signatures. If a larger variance is allowed during matching, the chances of a forger being labeled as a genuine person is increased (in biometric terminology, False Acceptance Rate (FAR) is increased). And, if this variance is decreased, then there are chances that even a genuine person may not get access and be labeled as a forger (which means increase in False Rejection Rate (FRR)). So, a balance should be maintained and an optimum threshold must be present such that both FAR and FRR can be kept low, though it is obvious from the situation that the two terms are inverse of each other . In most of the signature recognition approaches, a compromise is reached between the two values, that is, the variance threshold is decided for the complete database.

The verification process comprises of 2 modules, namely, matching and decision making. A detailed description of these modules is given below:

3.5.1 Matching

This module matches the feature vector extracted from the signature image to be tested with the corresponding feature vector retrieved from the database. The error is computed using the weighted Euclidean distance between the feature matrices. The distance measure for the unknown sample is defined as:

$$Dist = \left(\sum_{i=1}^{n} C_i \left((X_i - M_i)^2 / \sigma_i^2 \right) \right)^{\frac{1}{2}} \tag{3.8}$$

where n is the total number of features used for a particular signature instance, C_i is the weight associated with each feature, the nature of which is discussed in more detail below, X_i is the i^{th} feature for the unknown signature, M_i is the mean of the i^{th} feature calculated over the authentic sample instances, and σ_i is the standard deviation calculated for the i^{th} feature on the same set.

The optimal value of C_i in the above formula has been a subject of a great number of research studies. M Ammar had taken all the constants as one [1, 2, 3]. Jain and Griess have suggested two approaches [10]. In one, all the constants are global in nature i.e. all the signatures share a single value for each of the C_i's. The other approach suggested is to have a global base value plus an offset value for each of the signatures independently, i.e.

$$C_{i,p} = G_i + \alpha_{i,p} \tag{3.9}$$

where $C_{i,p}$ is the constant value for i^{th} feature of signature of identification number $= p$, G_i is the global constant shared by all the signatures, and $\alpha_{i,p}$ is the offset for i^{th} feature of signature of identification number $= p$.

In [17], Ramesh and Murty have used genetic algorithms for determining the values of these constants for optimal results. In this case, a global feature constant with a signature specific offset calculated from the authentic test instances was used.

Global weights are used where the weights are chosen in a manner which increases inter signature distance but reduces intra signature distance. Currently

an empirical method is used for the weight allocation depending on what the researcher presumes to be an important feature and its corresponding weight.

Table 3.1, table 3.2, table 3.3, table 3.4 show the weights assigned to various static features in offline to offline, online to online, online to offline, and offline to online verification systems.

Feature No.	Feature Weight	Feature No.	Feature Weight	Feature No.	Feature Weight
1	1	10	1	19	2
2	1	11	2	20	2
3	1	12	1	21	1
4	2	13	1	22	1
5	1	14	1	23	2
6	2	15	2	24	2
7	1	16	2	25	1
8	1	17	1	26	1
9	1	18	1	27	2

Table 3.1. Weights used in Offline to Offline Matching

Feature No.	Feature Weight	Feature No.	Feature Weight	Feature No.	Feature Weight
1	1	10	0	19	1
2	1	11	1	20	1
3	1	12	1	21	1
4	1	13	1	22	1
5	1	14	1	23	1
6	0	15	1	24	1
7	1	16	1	25	1
8	1	17	1	26	1
9	0	18	1	27	1

Table 3.2. Weights used in Online to Online Matching

If both the testing and the training images are taken using the digital signature pad, then *Dynamic Warping* is employed.

Dynamic Time Warping Algorithm: This algorithm takes two signals and returns a measure of dissimilarity between them. It tries to elongate or squeeze time line at various steps so that the two signals can be matched optimally. It places one signal on x-axis and the other on y-axis, there by forming a grid on a two dimensional plane. Inside each grid cell $cell_{i,j}$, distance between i^{th} value of

Feature No.	Feature Weight	Feature No.	Feature Weight	Feature No.	Feature Weight
1	0	10	0	19	1
2	0	11	0	20	1
3	1	12	0	21	0
4	1	13	0	22	0
5	1	14	0	23	1
6	0	15	0	24	1
7	1	16	1	25	0
8	0	17	0	26	0
9	0	18	0	27	1

Table 3.3. Weights used in Online to Offline Matching

Feature No.	Feature Weight	Feature No.	Feature Weight	Feature No.	Feature Weight
1	0	10	0	19	1
2	0	11	1	20	1
3	1	12	0	21	1
4	1	13	0	22	0
5	1	14	0	23	1
6	0	15	0	24	1
7	1	16	1	25	1
8	0	17	1	26	0
9	0	18	0	27	1

Table 3.4. Weights used in Offline to Online Matching

the first signal and j^{th} value of the second signal is placed. The best match will be the path through the grid that minimizes the total distance between the two sequences. The total distance is given as the sum of the distances between the individual elements on the path. For a reasonable grid size, the number of possible paths can be very large. To reduce the search domain, optimization is done by restricting the domain to a smaller one, which satisfy certain basic conditions of a 'good path'. The following conditions were used in this work:

- Monotonic condition: The path will not turn back.
- Continuity condition: The path advances one step at a time.
- Boundary condition: The path starts at the bottom left and ends at the top right.

3.5.2 Decision Making

For the decision making part, either a global threshold or a local threshold for each signature can be used. Due to high variability in the signatures of individuals, a local threshold gives better results although it may increase data storage necessities. Finally, the system gives the output to the calling module/user whether the signature is authentic or forged based on the threshold.

To classify unknown signatures, a global threshold Θ_{tresh} [14] is taken. If the measured distance for a given unknown signature is greater than Θ_{tresh}, the signature is labeled a forgery; otherwise, it is decided as genuine. Θ_{tresh} is chosen to correspond to a given value of z for the normal distribution. If $((X_i\text{-}M_i)/\sigma_i)$ in Equation 3.8 is constrained to be atmost z for each i for a signature to be authentic, *Dist* must be atmost

$$(\sum_{i=1}^{n} z^2)^{\frac{1}{2}} = (n * z^2)^{\frac{1}{2}} = \sqrt{n} * z \tag{3.10}$$

so that the corresponding value of Θ_{tresh} is given by:

$$\Theta_{tresh} = z * \sqrt{n} \tag{3.11}$$

where, Θ_{tresh} is the threshold for the system, n is the number of static features used, and z is a constant.

The database has been tested with a number of values of z. A satisfactory result was obtained with a value of z in the range 1.6 to 2.7. Value of $n = 27$. A detailed discussion of the results is covered in the subsequent chapter.

Chapter 4

Results And Discussion

The biometric characteristics of a sample signature can never be exactly the same as those provided during the registration procedure for the same id. This requires the matching algorithm to return results which are near matches to the characteristic given, and hence the need for a threshold arises. In the integrated signature verification system, an upper threshold is used i.e all the matches which are below this specified threshold are said to be valid and the others are rejected.

The performance of a biometric system is measured in certain standard terms. These are *False Acceptance Rate* (FAR), *False Rejection Rate* (FRR) and *Equal Error Rate* (EER). FAR is the ratio of the number of unauthorized users accepted by the biometric system to the total number of verification attempts made. FRR is the ratio of the number of authorized users rejected by the biometric system to the total number of verification attempts made. EER is a point where FRR and FAR are same.

As the thresholds are system dependent, they cannot be used to effectively compare different biometric systems. The *Receiver Operating Characteristic* (ROC) is used instead of thresholds for this purpose. The ROC is a plot depicting the FAR along the X-axis and the genuine acceptance rate along the Y-axis.

The proposed system has been tested on a database of 100 individuals, which comprises of six offline signatures and two online signatures for each identification number. The test data for each individual consists of three offline and one online genuine signatures and one forged signature. Initially, an arbitrary threshold was

chosen. After testing, the threshold is set to the value corresponding to the EER point, which aids in improving the system performance.

The feature vectors of offline and online images (shown in Figure 4.1) for an identification number is shown in Table 4.1.

Figure 4.1. Offline Signature and Online Signature Images for id = 8

Depending on the types of the test sample and the signature in the training database, the system can be classified into offline to offline, online to online, online to offline, and offline to online. Here, the feature vector extracted for the test sample is compared with the mean and standard deviation values stored against the identification number in the database using the weighted Euclidean distance. The test sample is accepted as being a genuine one if the *Dist* thus computed is less than the threshold and is rejected otherwise.

Offline to Offline Verification. This case corresponds to both the test image and the training image taken offline. So, the feature vector size here is limited to 27 as the dynamic features are not available in either case. The performance curve is shown in Figure 4.2(a). The curve indicates best performance of the system for a threshold value of 1.9 where the two curves intersect, but at a cost of 32% error. Figure 4.2(b) shows the ROC curve for offline to offline signature verification system.

Online to Online Verification. This case corresponds to both the test image and the training image taken online. So, the feature vector size here is 38 as

Feature No.	Feature Value of Offline Image	Feature Value of Online Image
1	1.128787878787879	1.2151898734177216
2	0.45454545454545453	0.5569620253164557
3	0.5151515151515151	0.46835443037974683
4	0.5606060606060606	0.12658227848101267
5	0.1947325605043726	0.1371637658227848
6	1.0	1.0
7	0.5378787878787878	0.0
8	0.5909090909090909	0.13291139240506328
9	13.844647519582246	27.282864695986543
10	0.2584856396866841	5.4609468877673635
11	1.0	1.0
12	0.22506527415143604	0.2641192021148762
13	0.44507575757575757	0.4737086287860264
14	0.12263257575757576	0.556532560924383
15	0.004177545691906005	0.4852198990627253
16	0.7424242424242424	0.0379746835443038
17	0.6184388528138528	0.3470100392841554
18	0.23446800595238093	0.2194612458689281
19	0.23394255874673628	0.04878634943523191
20	0.6439393939393939	0.5379746835443038
21	0.857489838405869	0.992344786015672
22	0.13021711113314166	0.2086062554416985
23	0.23942558746736292	0.22710886806056235
24	0.5681818181818182	0.08227848101265822
25	1.0946026097271648	1.0163177155198273
26	0.09492612962363851	0.0888475613414122
27	0.07336814621409922	0.050708964191300165

Table 4.1. Feature Vectors of Offline and Online images for id = 8

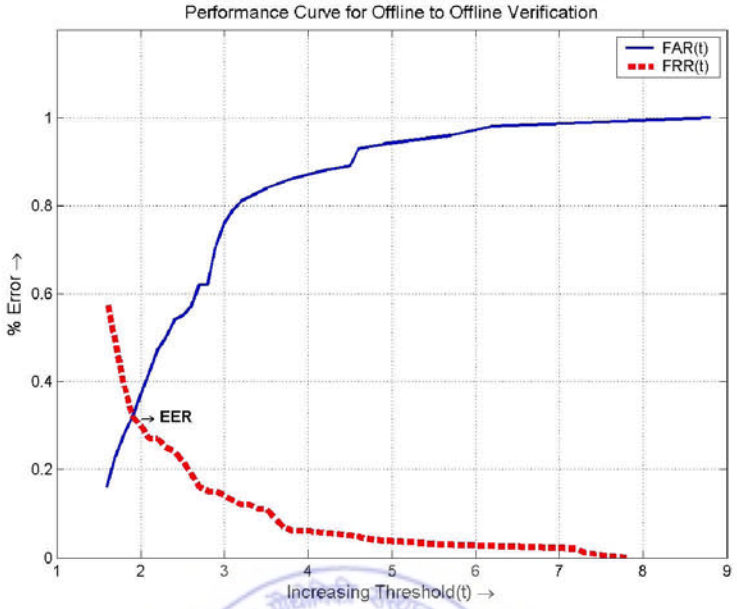
both the static and dynamic features are available in either case. Of all the 4 categories, online to online verification gives the best results as all the 38 features are taken into consideration. The performance curve is shown in Figure 4.3(a). Here, for a threshold value as low as 1.1, both FAR and FRR are zero, giving 100% accuracy. Figure 4.3(b) shows the ROC curve for online to online signature verification system. The test results also show that the system attains only 91% accuracy in this case if only dynamic features are considered.

Online to Offline Verification. This case corresponds to the online test sample being matched against the offline training images. So, the feature vector size

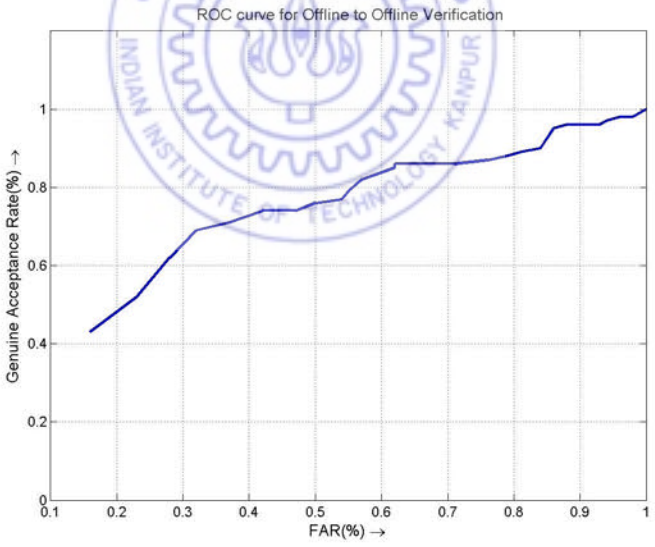
here is limited to 27 as the dynamic features are not available in the database for training images. The performance curve is shown in Figure 4.4(a). The curve indicates best performance of the system for a threshold value of 1.3 where the two curves intersect, but at a cost of 15% error. Figure 4.4(b) shows the ROC curve for online to offline signature verification system.

Offline to Online Verification. This case corresponds to the offline test sample being matched against the online training images. So, the feature vector size here is limited to 27 as dynamic feature set is not captured for the test image. The performance curve is shown in Figure 4.5(a). The curve indicates best performance of the system for a threshold value of 2.1 where the two curves intersect, but at a cost of 30% error. Figure 4.5(b) shows the ROC curve for offline to online signature verification system.

On the whole, the integrated signature verification system attains an accuracy of 85%. The performance is excellent when all static and dynamic features are available for both the test and the training samples.

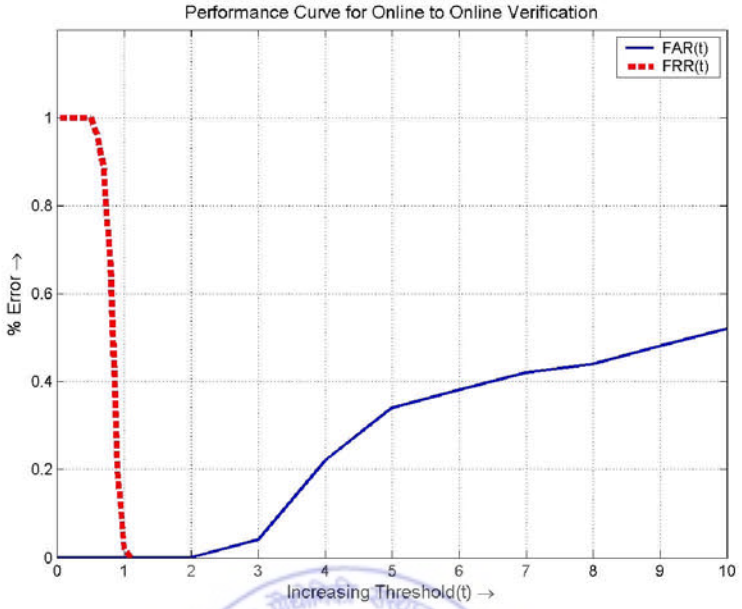


(a) Performance Curve

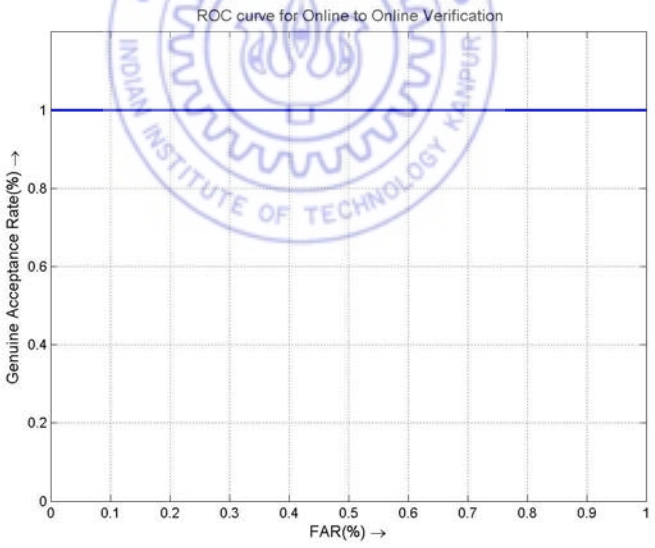


(b) ROC Curve

Figure 4.2. Offline to Offline Verification Curves

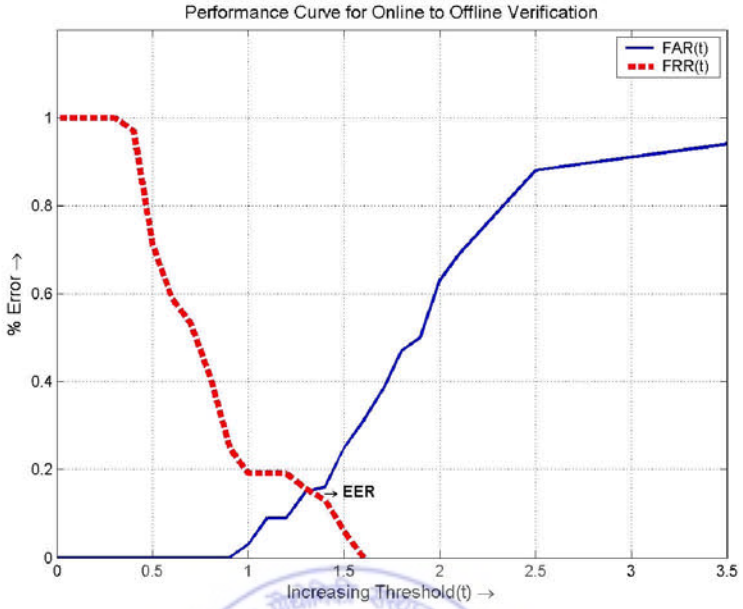


(a) Performance Curve

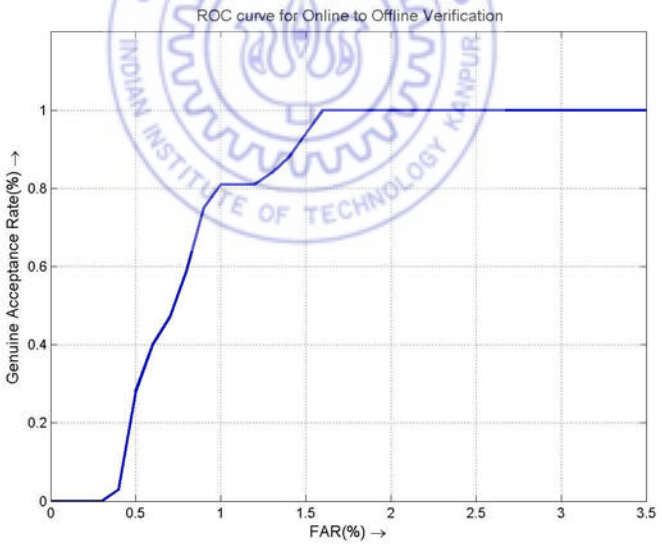


(b) ROC Curve

Figure 4.3. Online to Online Verification Curves

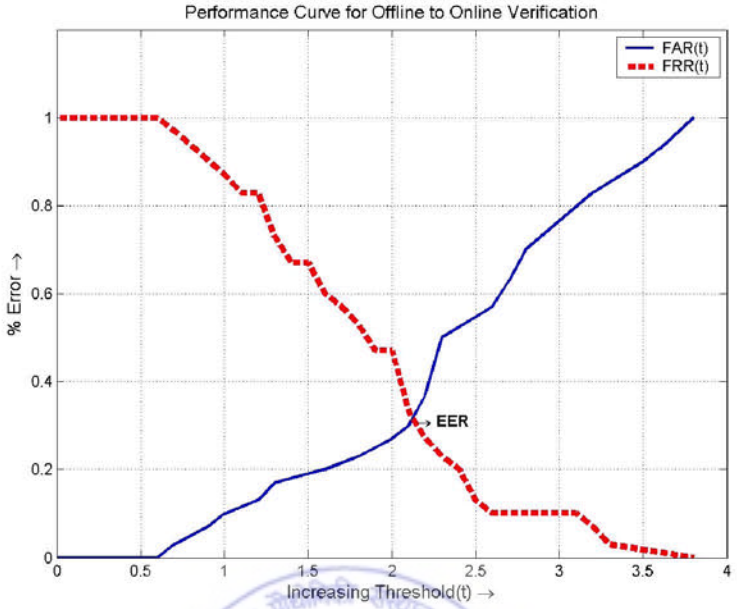


(a) Performance Curve

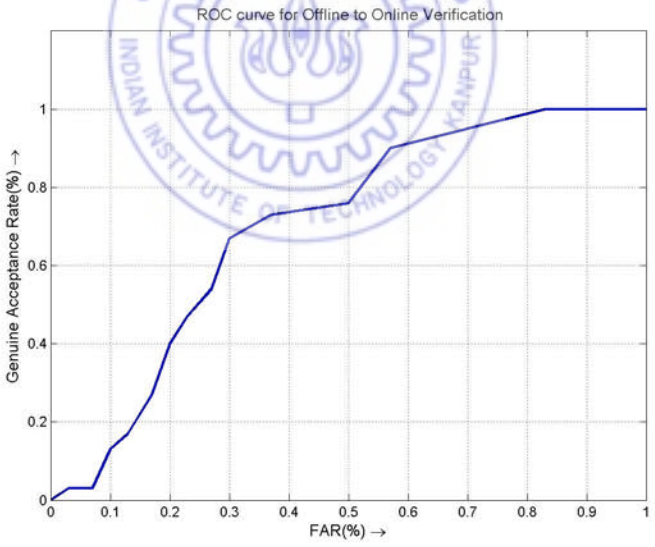


(b) ROC Curve

Figure 4.4. Online to Offline Verification Curves



(a) Performance Curve



(b) ROC Curve

Figure 4.5. Offline to Online Verification Curves

Chapter 5

Conclusions and Future Scope

A reliable signature verification system is an important part of law enforcement, security control and many business processes. It can be used in many applications like cheques, certificates, contracts etc. Generally, the verification can be done in two ways: offline and online depending on the way the input data is captured. An integrated version of the system not only provides a way to compare and match an offline signature against an online one and vice-versa, but also improves the system performance in those cases where both static and dynamic feature information is available.

The integrated signature verification system incorporates database management, noise removal and pre-processing, feature extraction (static, dynamic, local and global), learning and verification modules. The matching is done using weighted Euclidean distance and decision making is based on threshold based technique that gives near accurate results. The system showed promising results in case of online to online matching with almost 100% accuracy and satisfactory results in the remaining cases with an average accuracy of 85%. Different threshold values are used for matching depending on testing and training feature vectors (offline or online or both), thereby boosting the overall performance of the system.

Defining a new effective feature vector which results in minimum deviation for any signature instance may aid to further improvement of the system accuracy. An increase in number of partitions of each signature sample to extract local feature vectors from 3 to a higher value and increase in number of training samples would

help boost the system performance. An extension to the approach would be implementation of more accurate distance measurement techniques like Mahalanobis distance to verify the signature sample instead of Euclidean distance measure.

Bibliography

[1] Ammar, M. Performance of parametric and reference pattern based features in static signature verification: a comparative study. In *Proceedings of 10th International Conference on Pattern Recognition vol.1* (1990), IEEE Computer Society Press, pp. 646–648.

[2] Ammar, M., Yosheda, Y., and Fukumura, T. Off-line preprocessing and verification of signatures. Int. J. Pattern Recognition Arti. Intell. 2, pp. 589–902.

[3] Ammar, M., Yoshida, Y., and Fukumura, T. Structural description and classification of signature images. *Pattern Recognition 23*, 7 (1990), 697–710.

[4] Faundez-Zanuy, M. On-line signature recognition based on vq-dtw. *Pattern Recogn. 40*, 3 (2007), 981–992.

[5] Feng, H., and Wah, C. C. Online signature verification using a new extreme points warping technique. *Pattern Recogn. Lett. 24*, 16 (2003), 2943–2951.

[6] Fiérrez-Aguilar, J., Alonso-Hermira, N., Moreno-Marquez, G., and Ortega-Garcia, J. An off-line signature verification system based on fusion of local and global information. In *ECCV Workshop BioAW* (2004), pp. 295–306.

[7] Gonzalez, R. C., and Woods, R. E. *Digital Image Processing.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001.

[8] Huang, K., and Yan, H. Off-line signature verification based on geometric feature extraction and neural network classification. *Pattern Recognition 30*,

1 (1997), 9–17.

[9] Ismail, M. A., and Gad, S. Off-line arabic signature recognition and verification. *Pattern Recognition 33*, 10 (2000), 1727–1740.

[10] Jain, A., Griess, F., and Connell, S. On-line signature verification. *Pattern Recognition 35*, 12 (2002), 2963–2972.

[11] Lam, C. F., and Kamins, D. Signature recognition through spectral analysis. *Pattern Recognition 22*, 1 (1989), 39–44.

[12] Leclerc, F., and Plamondon, R. Automatic signature verification: The state of the art - 1989-1993. *IJPRAI 8*, 3 (1994), 643–660.

[13] Martens, R., and Claesen, L. On-line signature verification by dynamic time-warping. In *ICPR '96: Proceedings of the International Conference on Pattern Recognition (ICPR '96) Volume III-Volume 7276* (Washington, DC, USA, 1996), IEEE Computer Society, p. 38.

[14] Nagel, R. N., and Rosenfeld, A. Computer detection of freehand forgeries. *IEEE Trans. Computers 26*, 9 (1977), 895–905.

[15] Parizeau, M., and Plamondon, R. A comparative analysis of regional correlation, dynamic time warping, and skeletal tree matching for signature verification. *IEEE Trans. Pattern Anal. Mach. Intell. 12*, 7 (1990), 710–717.

[16] Plamondon, R., and Srihari, S. N. On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Trans. Pattern Anal. Mach. Intell. 22*, 1 (2000), 63–84.

[17] Ramesh, V. E., and Murty, M. N. Off-line signature verification using genetically optimized weighted features. *Pattern Recognition 32*, 2 (1999), 217–233.

[18] Shafiei, M. M., and Rabiee, H. R. A new on-line signature verification algorithm using variable length segmentation and hidden markov models. In *ICDAR '03: Proceedings of the Seventh International Conference on Document Analysis and Recognition* (Washington, DC, USA, 2003), IEEE Computer Society, p. 443.

[19] Tappert, C. C., Suen, C. Y., and Wakahara, T. The state of the art in online handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell. 12*, 8 (1990), 787–808.

[20] Yang, L., Widjaja, B. K., and Prasad, R. Application of hidden markov models for signature verification. *Pattern Recognition 28*, 2 (1995), 161–170.