DYNAMIC TIME WARPING BASED ONLINE SIGNATURE VERIFICATION USING ADAPTABLE GLOBAL CONSTRAINTS

by

ASHUTOSH PATHAK

A thesis submitted in partial fulfillment of the
requirements for the degree of
Bachelor-Master of Technology(Dual Degree)

to the

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur

CERTIFICATE

This is to certify that the work contained in the thesis entitled **Online Signature Verification** by **Ashutosh Pathak(Y4177108)** has been carried out under my supervision. This work has not been submitted elsewhere for a degree.

Dr. Phalguni Gupta

Professor

Department of Computer Science and Engineering

Indian Institute of Technology Kanpur

Kanpur, India

June, 2009

Abstract

Online Signture Verification is fast emerging as an active research area in the field of biometrics. This is attributed to the increase in the number of identity theft cases and the future prospects that it offers in automatic clearing of electronic bank cheques. The aim of this work is to do improvements in Dynamic Time Warping (DTW) based Online Signature Verification system. The developed system uses DTW with Ratanamahatana-Keogh(R-K) band based global constraints. Global constraint restricts the warping path globally and speed up the calculation of DTW distance. Many global constraint models have been proposed including Sakoe-Chiba (S-C) band, Itakura Parallelogram, and Ratanamahatana-Keogh (R-K) band in speech community. The R-K band is a general global constraint model that can represent any global constraints with arbitrary shape and size effectively. However, the current R-K band does not support time series sequences with different length, therefore, we propose some changes to make it applicable in the field of signature biometric. Experiments are conducted on IITK databse and Signature Verification Competition database (SVC2004) for basic DTW and the modified DTW. The system has acheived an improvement of 40% and 20% in EER on IITK and SVC2004 database respectively in case of random forgery. The results obtained show an amelioration of the classical use of DTW.

Acknowledgments

I would like to express my sincere gratitude to my advisor Dr. Phalguni Gupta for giving me the opportunity to work on this interesting topic . I appreciate his valuable advice and support during the course of study.

I would also like to thank Pradeep and other members of biometric lab for their immense help in developing the IITK on-line signature database.

My colleges and friends receive my heartfelt thank for their valuable friendship and discussion which facilitated my writing .

Lastly , I would like to thank my parents for their enormous encouragement and assistance, for without them, this work would not have been possible .

Ashutosh Pathak

Dedicated to my Family and my Country

Contents

List of Figures

Chapter 1

INTRODUCTION

Due to the prospering of electronic commerce, the demand of automatically verifying some ones identity is growing in daily routine this has inspired the development of a wide range of Biometric systems. Biometric system is a pattern recognition system based on Behavioural and Physiological traits of human beings. Behavioural characteristics are usually the actions or reactions of human beings in relation to the environment. Examples include signature, gait, and voice. On the other hand, physiological traits are related to the inherent property (shape/odour) of the body. Examples include fingerprint, face recognition and DNA. Biometric systems differ from system based on possession of something (Key, card etc.) or the knowledge of something (password etc.) as they rely on human characteristics. While looking for the proper biometric to be used in an environment, one has to understand the availability of the following characteristics in the biometric system [14]:-

- **Universal:** It must be such that every human being should have it.
- **Invariant:** It should not change very rapidly over time.
- **Singular:** It is the measure which shows how well the biometric separates individuals from another.
- **Inimitable:** Exact replica of biometric feature should not be producible from other means.
- **Reducible And Comparable :** Ease of acquisition for measurement and comparison.
- **Reliable And Temper Resistant:** It should be robust and free from manipulation.

Signature belongs to the set of biometrics based on behavioural characteristic. It is not nearly unique or difficult to forge as compared to other biometrics and also it changes over time. However it is widely accepted by the public for low security needs.

1.1 Handwritten Signatures

Handwritten signature based system for authentication, authorization and writer identification is one of the most socially or legally accepted biometric system. Hence most of the documents like bank cheques need signature which can be automated offline for verification. Figure1.1 shows the evidence of documents containing signature which are older than 15th century. In today's era with the increasing cheque frauds and losses, signature biometric is moving towards on line signature based system which have lower processing time and higher accuracy.

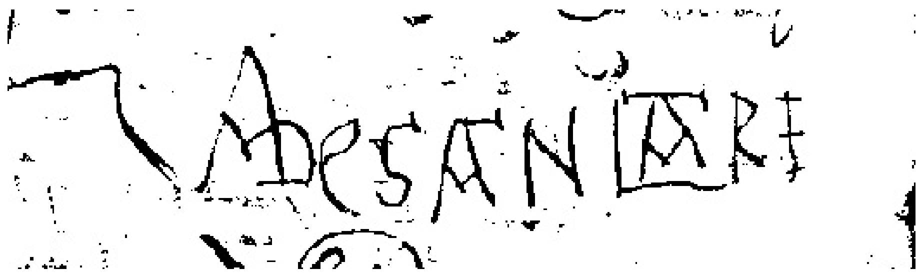


Figure 1.1: The oldest document that bears the signature of a probable ancestor Bzanaire-Besner, read "Bsaniare" and date of 1585. [20]

1.2 Identification Vs Verification

A biometric system can be used for *identification*, *verification* and *recognition*. Identification and recognition, are synonyms terms. The selection and implementation of biometric technology are closely dependent on these objectives. The proper distinction between identification and verification is that

- Verification is basically a two class based classifier which tells whether the entity belongs to that class or not. During verification, user provides the biometric sample from which the query template is extracted and a matching strategy is needed to match it with the templates of the same user in the database. It generates a matching score. This matching score is compared with a threshold value. Thus the verification system generally tries to give the answer "Am I who, claim to be ?". Since it is one to one matching system, verification is relatively fast.

- Identification is a many class based classifier which tells given a certain number of classes to which class a query entity belong to. During identification, template extracted from user provided biometric sample is compared to the all the available templates stored in a database with some matching algorithm. Due to the above complexity of identification system, it usually needs large database and is costly to deploy. Identification system generally tries to answer the question "Who am I?". As query template is compared to all the stored templates, identification system is slower than verification system.

This thesis focuses on on-line signature verification. In the context of signature verification, it is a classifier having two classes genuine signatures and forgery signatures. Signature verification differs from identification in the sense that it utilises only the templates of single user during matching process rather than the templates of all the users.

1.3 Signature Based Verification System

Signature is basically a static image made on documents involving the dynamic process of pen position, pressure, tilt and angle as a function of time. Signature can be either *on-line* or *off-line*. *On-line* signature is captured using an electronic device. For example, it is a digitizing tablet which is capable of capturing pressure, coordinate, tilts of a specially designed pen as it moves over it. *Off-line* signature is captured by scanning the documents containing the signature. Although *offline* signature uses static information whereas *online* signature uses dynamic information, thus *online* signature provide more information such as pressure of the pen tip, pen down time and pen up time etc. but the off-line systems in many case are, economically more viable and sufficiently accurate.

1.4 Quality Performance Measures

The important factors in evaluation of the performance of Biometric System are *false acceptance rate* ***(FAR)***, *false rejection rate* ***(FRR)***, *equal error rate* ***(EER)*** and *accuracy*. Various terms like FAR, FRR and Accuracy are calculated from confusion matrix as shown in Figure 1.2. Confusion matrix is a visualization tool for experimental results where each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The entries in the confusion matrix have the following meaning in the context of our study:

- True Positive: Those genuine test signatures which are predicted as non forgery(genuine) attempts.

- True Negative: Those imposter test signatures which are predicted as forgery attempts.

- False Positive: Those imposter test signatures which are predicted as non forgery (genuine) attempts.

- False Negative: Those genuine test signatures which are predicted as forgery attempts.

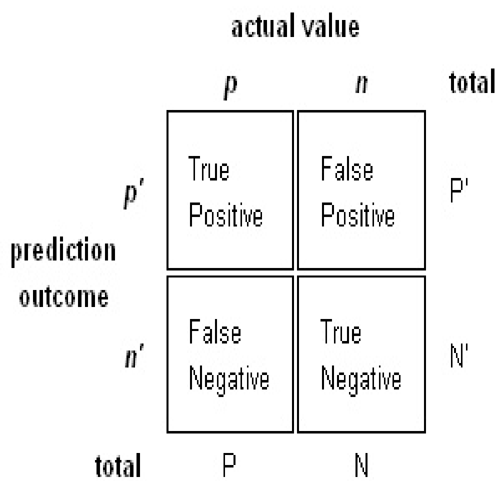


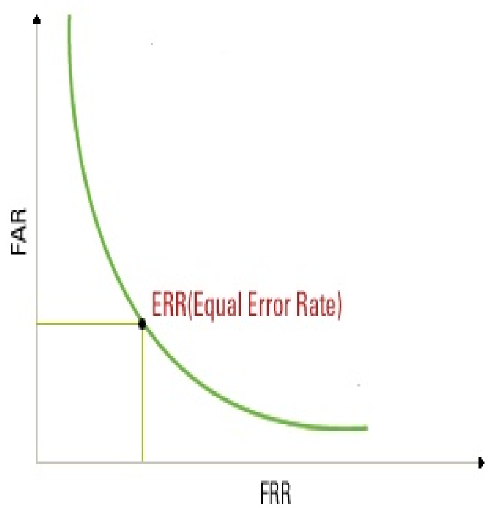
Figure 1.2: Confusion Matrix

FRR is the ratio of number of genuine test signatures rejected to the total number of genuine test signatures submitted to the system and FAR is the ratio of total number of imposters to the total number of imposter signatures submitted to the system. It can be shown that FAR and FRR are inversely related. Lowering one, results in the increase of another and vice versa. Hence, one can use another measure such as EER which is the point at which FAR becomes equal to FRR and the corresponding threshold is known as equal error rate threshold. Other widely used performance plots are Receiver Operating Characteristic curve (Roc Curve) and Detection error trade-off Curve Figure 1.3a. ROC Curve is plotted between FAR and (1-FRR) while the graph plotted between FAR against FRR is called DET-Curve. From DET curve shown in Figure 1.3a, it is clear that it is not possible to minimise FAR and FRR together and there is a point at which both the errors become equal and that point is known as ERR. The quality of system is better when the DET curve is closer to the origin . Various terms like FAR, FRR and Accuracy are calculated from confusion matrix as follows :-

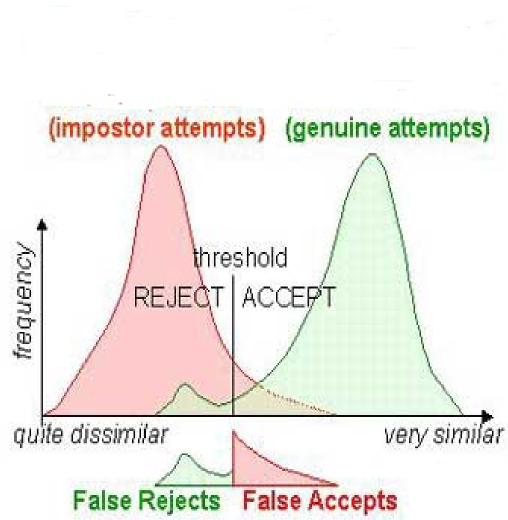
$$\begin{aligned} \text{FRR} &= \frac{\text{number of genuine test signatures rejected}}{\text{number of genuine test signatures submitted}} \\ &= \frac{FalseNegative}{FalseNegative + TruePositive} \end{aligned} \tag{1.1}$$

$$\begin{aligned} \text{FAR} &= \frac{\text{number of forgery test signatures accepted}}{\text{number of imposter test signatures submitted}} \\ &= \frac{FalsePositive}{FalsePositive + TrueNegative} \end{aligned} \tag{1.2}$$

$$Accuracy = \frac{TruePositive + TrueNegative}{FalsePositive + TrueNegative + FalseNegative + TruePositive} \tag{1.3}$$



(a) Changes in FRR and FAR according to changes in the critical value



(b) Concept Map on FAR and FRR

Figure 1.3: Performance Curves

1.5 Forgeries

There are mainly two types of forgeries in the field of signature verification namely skill forgery and random forgery. Skill forgery is done by a forger who has access to the actual signature samples of the user. Random and zero effort forgery is produced by the forger who has no information about the name and signature of the user. The signature used can be any random signature of another user or the forgers own signature. Skilled forgeries can be further subdivided into *amateur* and *professional* forgeries. Professional forgeries are obtained from person who has expertise in handwriting analysis. In the field of on-line signature amateur forgeries can further be divided into *"home-improved"* and *"over-the-shoulder "* [5]. *"Home-improved"* forgeries are done by the forger who has copies of static images of the signature and he does ample practice to do the forgery with help of signature copies. On the other hand, *"over-the-shoulder"* are produced when the forger witnesses the signing process of genuine signature, so he not only has the knowledge of static image but also that of the dynamics of the signature. An example of a skill forgery signature and the individual plots of different dynamics captured from the two signatures like pressure etc., from SVC on-line signature database is shown in Figure 1.4 and 1.5.

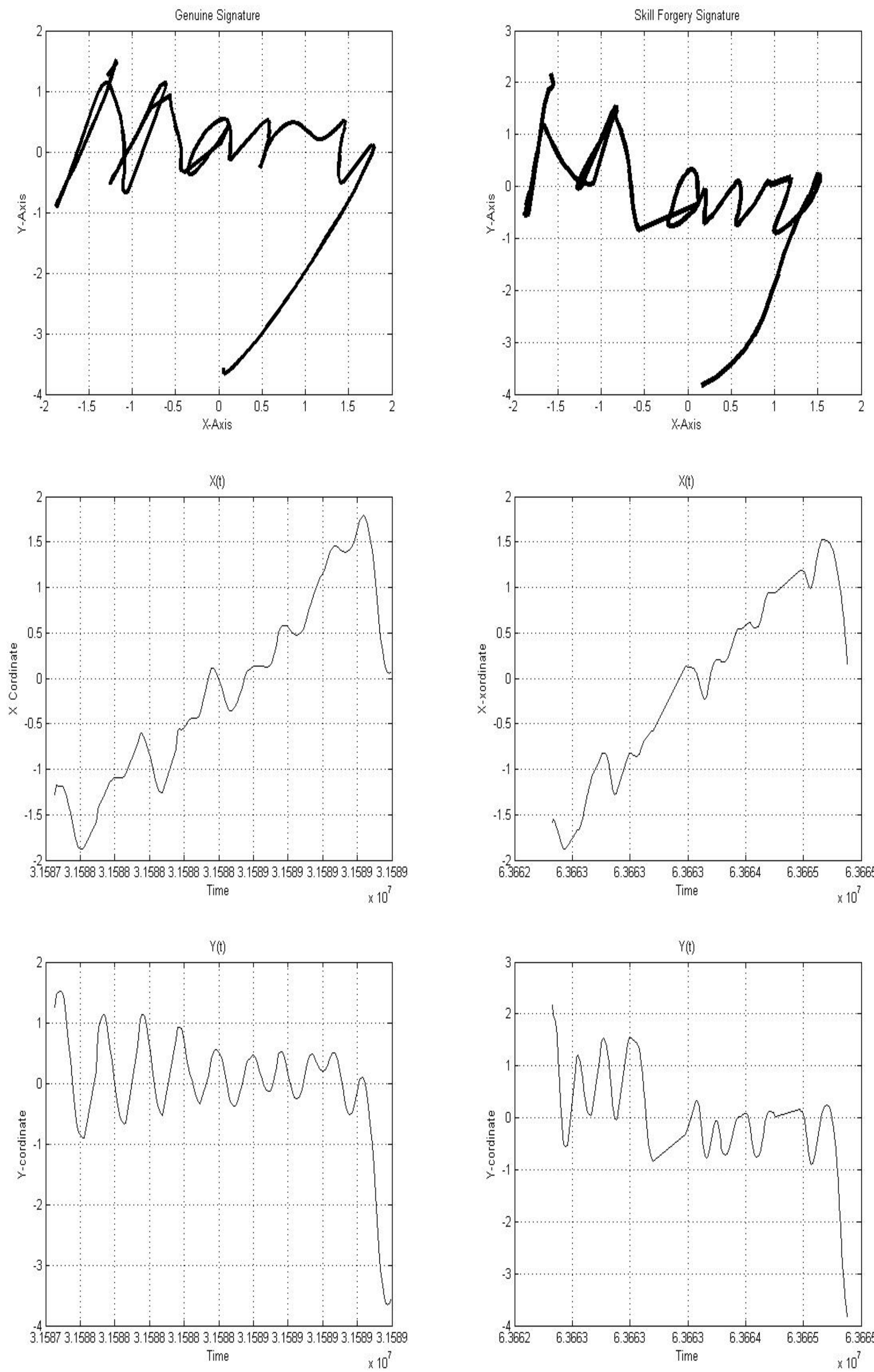


Figure 1.4: Example of Skill Forgery plots of Shape, X-Coordinates and Y-Coordinates with Time From SVC20004 Database

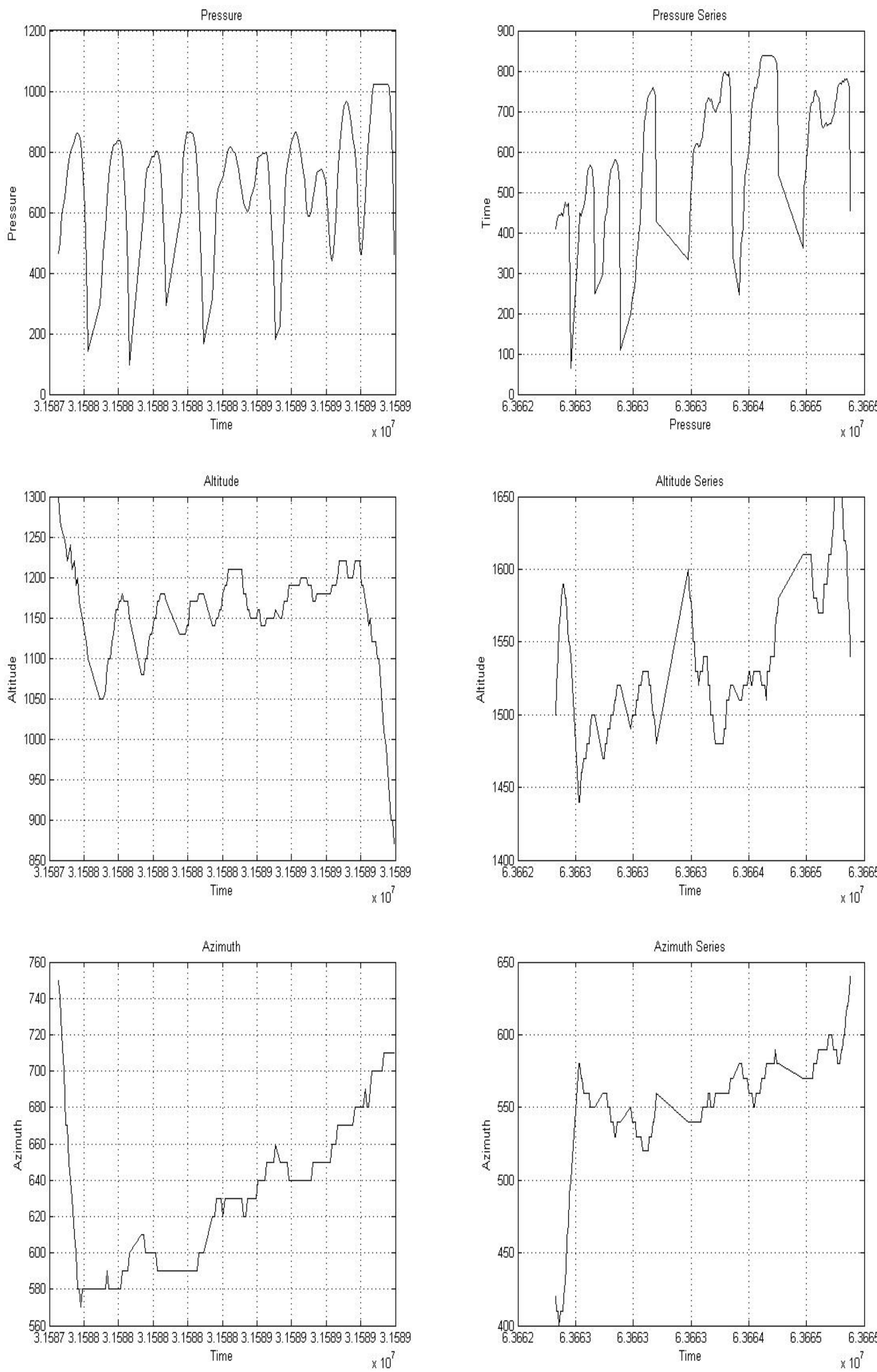


Figure 1.5: Example of Skill Forgery plots of Pressure, Altitude and Azimuth From SVC20004 Database

1.6 Problem Statement

Signature verification is widely used in the field of bank cheques verification. As the online verification is more reliable than offline systems, there is a rapid advancement in the field of online signature verification to make them better in terms of performance and response. Due to the better performance of on-line verification systems, specifically based on DTW. This thesis tries to make the existing DTW based online signature verification system better in terms of accuracy and speed up using learned global constraints.

1.7 Motivation

In this thesis we have discussed a Dynamic Time warping Based Signature Verification. The DTW-algorithm originates from the area of speech recognition [17], and has been applied successfully in many signature verification techniques. DTW-algorithm is used to align non linearly feature vectors (or observation sequences), which are then matched or compared. Sometime due to too much alignment done by DTW, it does not give intended score to satisfactory classify the the data from two different classes. So people started working on implementing constraints in warping path as explained in Section 2.7.4. Recently, few adaptable global constraints are suggested by authors [21] which can give extremely good results by making the alignment within the constraints and also speed up the DTW as explained in Section 2.7.5. This thesis is motivated by R-K Band based global constraint learning algorithm[21] because they can represent any global constraints with arbitrary shape and size effectively.

1.8 Outline of the Dissertation

This thesis is divided into five chapters. Brief outline of each chapter is given below.

- **Chapter 2:**[*Background and Literature Review*] It discusses some current research in the field of On-line signature verification and gives some historical perspective. It discusses the various online signature verification systems and the performance measure obtained by them. As the proposed approach is based on the modification of DTW, this chapter explains the DTW based system and the various constraints applicable to it.

- **Chapter 3:**[*Proposed System*] It proposes a new online signature verification system and discusses the various steps involved like feature extraction, verification and threshold selection criteria used in this thesis .

- **Chapter 4:**[*Results and Experiments*] This chapter discusses the data set and experimental set-up used in this dissertation. In this chapter results and performance of the proposed system are compared to the some existing system.

- **Chapter 5:**[*Conclusion and Future Work*] This chapter informs some outstanding issues of the system developed in this thesis and shows some ways to continue this work further. This chapter also talks about the possibility to improve the quality of performance measures of the proposed approach.

Chapter 2

Background And Literature Review

2.1 Introduction

In this chapter, some existing work along with various technology used in the field of online signature verification is discussed. The various verification systems are described in terms of feature extraction techniques, matching techniques, size and type of the database used for the target forgeries and the obtained results. The major techniques used in on-line signature verification can be broadly categorized into four categories. These are Template Matching Techniques (section 2.1), Bayesian Classifier (section 2.2), Neural Networks(section 2.3) and Support vector Machines (section 2.4). In section 2.5, a Template Based Matching technique called Dynamic

Time Warping(DTW) is explained in detail, as it is the fundamental basis of our proposed algorithm .

2.2 Bayesian Classifier

Bayesian Classifier is based on Bayesian networks which represents the casual probability of the random variables. Theses probabilities are calculated from Bayes theorem. The classifier gives the decision based on probability obtained of the test user. For more detail read the tutorial by Heckermann [11]. The main disadvantage with Bayesian network is that they require large training data. Xiao et. al. [27] implemented the modified Bayesian based Signature Verification system. They have used 60% data for training and 40% data for testing and have achieved the minimum 20%FRR and 14% FAR. Hairong Lv et al. [16] have proposed the first Dynamic Bayesian Networks(DBN) based online signature verification system. They have defined the topology of DBNs as simple HMM having two chains, called them as Coupled HMM (CHMM). They have used dynamic features of trajectory, tangent angles, pressure and velocities for their system. Their database consist of 3500 signatures divided into 100 sets, each having 25 genuine signature and 10 skill forgery signature. They have obtained an EERs of 1.5%.

2.3 Hidden Markov Models

Hidden Markov Models(HMM) are stochastic method of modelling a sequence of observations and their relationships with each other. HMM's are used to model the signals whose properties changes over time, That's why they have been extensively used in signature and speech recognition areas. Rigol et. al. [22] compared the off-line and on-line signature verification system using HMM's. They have used the Viterbi algorithm to compute the likelyhood probability of the test signature, belonging to the claimed writer. They have tested their system on a very small data set containing 14 users each having 20 signatures of which he used 16 signatures for training. For the purpose of testing he tested 60 forgeries out of which 40 were skilled forgeries. He claimed to obtain ERR of 1.9% with velocity, pressure and Fourier transform features in online case. Fierrez et al. [8] have implemented a HMM based online signature verification system having 2 HMM states and 32 Gaussian mixtures per state. Their proposed HMM system was ranked second for skilled forgeries and first for random forgeries in SVC2004. In addition to the basic feature captured during data acquisition their proposed system has following features path tangent, velocity magnitude, log curvature and total acceleration magnitude. They have obtained the performance results of 0.74% and 0.05% EER for skilled and random forgeries respectively on a database of 145 subjects comprising 3625 client signatures, 3625 skilled forgeries and 41,760 random impostor attempts.

2.4 Neural Networks

The Neural Network(NN) are parallel computing system consist of large number of perceptrons. The main characteristic of NN is their ability to learn non-linear relationship through sequential training and adapt themselves to the data. NN attempts to use the organisational principle in a network where nodes are perceptrons and directed weighted edges come and go out of the nodes. NN based system are good because of less preprocessing and also due to the fact that training data has large variance. Maryam et. al. in 2008 developed a new Spatial Temporal NN Based online signature verification [6]. They claimed to have obtained the FAR of 7.5% and FRR of 12.5%, without any preprocessing and normalisation on the data having 100% size difference. Baltzakis et al. [1] have implemented a two stage RBF neural network based offline signature verification system. They achieved 80% correct classification. They collected the signatures of signers by asking them to have as much variation in their signature as they have in real life. Hence they claimed to have very robust signature verification system .

2.5 Support vector Machines

Support Vector Machines (SVM) are recently developed machine learning algorithms which are advanced through the field of statistical learning theory. They were first proposed by Vapinik(1998) [24]. In SVM, input data is mapped to a high dimensional space using kernel functions to better classify the data with help of separating planes that maximises the separation between the classes. Kholmatov et. al. [15] have proposed the combine DTW and SVM based system and have achieved the lowest equal error rate of 1.4%(skilled forgery) in SVC(2004). Fuentes et al. [9] have proposed a fusion of HMM and NN based online signature verification via SVM. They have implemented 8 dynamic and 9 static features per signature point. They applied the SVM with various types of kernels linear, quadric and cubic on the final fused feature vector. They have achieved a TER of 5.32%. Gruber et al. [10] have proposed two new kernels "LCSS-global" and "LCSS-local" for time series classification with SVM. They claimed to have an TER of 0.93% using "LCSS-local kernel" on a database of 12 users each having 20 signatures out of which 7 genuine and 35 random forgery signature are used for training and 13 genuine and 30 random forgeries are used for testing.

2.6 Template Matching Techniques

A template is a pattern class that serves or determine as a model or pattern for matching. Template matching techniques are some of the oldest methods used in the field of signature verification. DTW is the most popular template based matching technique. We will discuss the DTW more detail in subsequent section. Jain et al. [13] have used a pressure sensitive tablet for capturing on-line signatures and after the re sampling preprocessing captured some of the features like x,y coordinate difference, absolute, relative speed and curvature and used the dynamic time warping in which they have incorporated the stroke difference cost to the final distance measure. They have investigated the verification using minimum average, maximum average and average to the reference (or template) signature. They have suggested that by using the average of minimum distance measures to the reference sets give the lowest error rates than simple average of all the distance measures to the reference set. They used the test data set of 1232 signatures including 60 skilled forgeries from 102 users. They have reported a 2.8% false accept rate and a 1.6% false reject rate for writer dependent threshold and a 3.3% false accept rate and a 2.7% false reject rate for common thresholds, using only random forgery signatures. Tanabe et al. [25] have investigated the DTW using only pressure feature after the signature normalisation using Discrete Fourier Transform (DFT). They have claimed to obtain a 6% error rate of Type 1 and Type 2 errors. Feng et al. have proposed a new warping technique called as Extreme Point Warping (EPW). They have proved it to be more adaptive as instead of warping the whole signature as DTW does, EPW warps a set of selected points on the signature. They have tested their system on a database of 25 users each having 30 genuine signatures and 10 forgery signatures consisting of 5 feature signals mainly x series, y series, torque series, segmented x and y series based on Gaussian window. They have claimed that EER is improved by a factor of 1.3 and computation time is reduce by a factor of 11 as compared to DTW. Faundez et al. [7] have suggested a new vector quantisation (VQ) based Dynamic Time Warping algorithm (VQ-DTW). They have tested their system on basic dynamic features captured with the digital tablet. The main aspect of their approach is that they concatenated the features of all the training sample into a single vector and quantised them using LBG algorithm. They have achieved an identification rate of 99% and EER of 6.7% with skilled forgeries on a database of 280 users.

2.7 Dynamic Time Warping

Dynamic Time Warping (DTW),has been introduced to the data mining community by Berndt and Clifford [2]. The Main Objective of DTW is to time warp the observation sequences non linearly with the reference sequences to align them before they are compared (or matched). It involves the matching of test sequences with one or more reference (or template) sequences, therefore it is better categorised as a *template matching* technique. DTW based series alignment algorithm is used to obtain a DTW distance between two series under some constraint. The system developed in this thesis uses this distance measure to align the two series and calculate the distance between two signatures.

2.7.1 Algorithm

Consider x and y be the two time sequences represented by vectors. In order to compute the optimal alignment between x and y, we first need to construct a cost grid (matrix) like the one shown in Figure 2.1. Each node in the grid relates the specific element of x series with specific element of y series. Node(i,j), for example relates the i^{th} component of X-series with the j^{th} component of Y-series ie. $x(i)$ with $y(j)$. Each of the node in the grid is filled with node based cost. For each node, the distance between the related X and Y series components are calculated as follows,

$$D_{Node}(i, j) = \sqrt{(x(i) - y(j))^2} \tag{2.1}$$

The main Objective of DTW is to find the Optimal Warping path (W)

$$W = w_1, w_2, ...w_K \tag{2.2}$$

where $w_l = (i(l), j(l))$ through the grid for which the sum of the node based cost

$$D^W_{Node}(X, Y) = \sum_{l=0}^{l=K} D_{Node}(i(l), j(l)) \tag{2.3}$$

is minimum, subject to the conditions

$$(i_0, j_0) = (1, 1), \;\; (i_k, j_k) = (m, n) \tag{2.4}$$

$$i_k > i_{k-1} \quad k = 1, 2, ...K \tag{2.5}$$

$$j_k > j_{k-1} \quad k = 1, 2, ...K \tag{2.6}$$

$$|j_k - j_{k-1}| \leq 1 \quad k = 1, 2, ...K \tag{2.7}$$

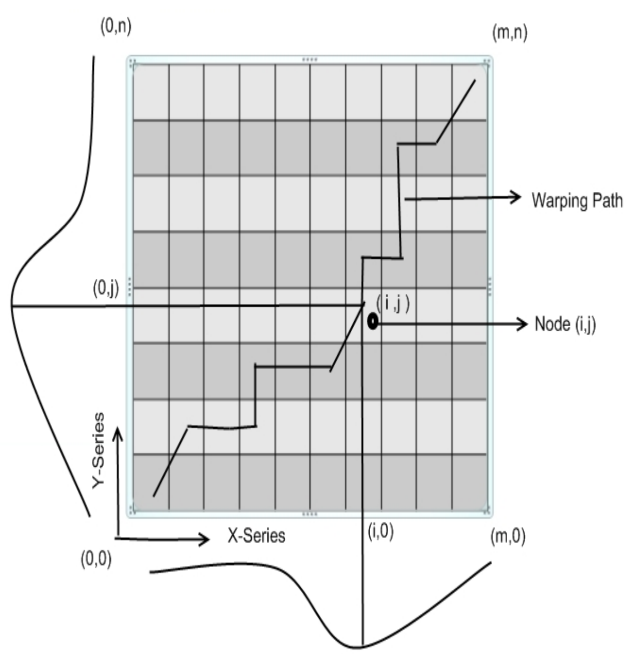


Figure 2.1: Illustration of DTW grid used to align two series.

$$|j_k - j_{k-1}| \leq 1 \quad k = 1, 2, ...K \tag{2.8}$$

Constraint 2.4 ensures that the warping path should be complete path. This implies that the first element (and last element) of two series are always mapped to each other. The starting node in warping path is (1,1) and the terminal node is (m,n). The detail explanation of this constraint is given in Section 2.7.4.1.

Constraint 2.5 and 2.6 ensure that the warping path is monotonically increasing. This implies that every node in the warping path can only have three possible preceding nodes, ie. (i,j-1), (i-1,j) and (i-1,j-1). The detail explanation and benefits of this constraints are given in Section 2.7.4.3.

Constraint 2.7 and 2.8 ensure that warping path should be continuous ie. there should be no breakup in warping path. This implies that there must not be any unaligned parts between the two series. The detail is given in Section 2.7.4.2.

The above constraints are called Local constraints. Optimal path on the grid is chosen under some constraints which are explained in section local 2.7.4 and global constraints 2.7.5. We now explain the DTW algorithm and how it is used to calculate the optimal alignment cost for the complete path:-

Given: Let two series between whom we need to calculate the Optimal cost be $X = \{x_1, x_2, x_3, ..., x_i, ..., x_m\}$ and $Y = \{y_1, y_2, y_3, ...y_j, ..., y_n\}$

Initialisation: First the node cost is calculated for all the nodes in the grid using some distance measure like euclidean distance or city block etc.

$$D_{Node}(i,j) = \sqrt{(x_(i) - y_(j))^2} \tag{2.9}$$

Recurrence: Let $D_{commu}(i,j)$ denotes the distance associated with the partial optimal path end at $Node(i,j)$. For each of the node in the grid the cumulative cost is calculated using Equation 2.10, in left to right and top to bottom order.

$$D_{Commu}(i,j) = D_{Node}(i,j) + min \left\{ \begin{array}{c} \alpha * D_{Commu}(i-1,j-1) + D_{trans}[(i,j)|(i-1,j-1)] \\ \beta * D_{Commu}(i-1,j) + D_{trans}[(i,j)|(i-1,j)] \\ \gamma * D_{Commu}(i,j-1) + D_{trans}[(i,j)|(i,j-1)] \end{array} \right\} \tag{2.10}$$

where D_{trans} is the cost associated with forward transition from one node to another. $D_{trans}[(i_k,j_k)|(i_{k-1},j_{k-1})]$ is the transition cost from node (i_{k-1},j_{k-1}) to node (i_k,j_k). α, β and γ represent the weight associated with different transition. Note that when the D_{commu} is same for two or three of the preceding nodes, by setting $\alpha = 0.5$, $\beta = 1$ and $\gamma = 1$ the order of preference of transition can be make as $(i-1,j-1)$, then $(i-1,j)$, or $(i,j-1)$.

Backtracking: The complete optimal path is found through backtracking. We start with the last node (m,n), then we find among its three neighbour $[D_{Cumu}(m-1,n-1), D_{Cumu}(m-1,n), D_{Cumu}(m,n-1)]$ whose cumulative cost is minimum and add that neighbour to the path. Continue the same operation from that neighbour. The process is continued till we reach the first node$(1,1)$in the grid.

Termination: After finding the complete path

$$W = [w_1(1,1), .., w_k(i,j), .., w_P(m,n)] \tag{2.11}$$

where $max(m,n) < P < m+n-1$ and $P = |W|$. Distance is calculated by adding the cumulative cost of all nodes in the warping path and normalise it with the number of nodes in the warping path. DTW distance is calculated below

$$DTW(X,Y) = \sqrt{\sum_{k=1}^{k=P} \frac{w_p}{P}} \tag{2.12}$$

2.7.2 Example

Following section illustrate the above algorithm with the help of an example. Let $X = [1,3,6,8]$ and $Y = [1,9,7,4,2]$ and be the two series we want to align. We first need to create the local cost grid and use any suitable distance measure to fill each node in the grid. The local cost grid associated with each node is shown in Figure 2.2(a) where each node cost is calculated using simple city block distance. For example $Node(1,1) = |1-1| = 0$ and $Node(2,3) = |3-7| = 4$. After initialising the grid, we calculate the cumulative cost at each node in left to right and bottom to top using Equation 2.10 with $\alpha = \beta = \gamma = 1$ and $D_{trans}[(i,j)|(i-1,j-1)] = D_{trans}[(i,j)|(i-1,j)] = D_{trans}[(i,j)|(i,j-1)] = 0$. The Final grid with cumulative cost is shown in Figure 2.2(b). For example, the cumulative cost of $Node(2,2) = 6 + \min(8,0,2) = 6+0 = 6$ and $Node(3,4) = 2 + \min(13,12,6) = 2+6 = 8$. The complete optimal path is found through backtracking. After backtracking the optimal path is found as shown in Figure 2.2(c). Final Warping path obtained is

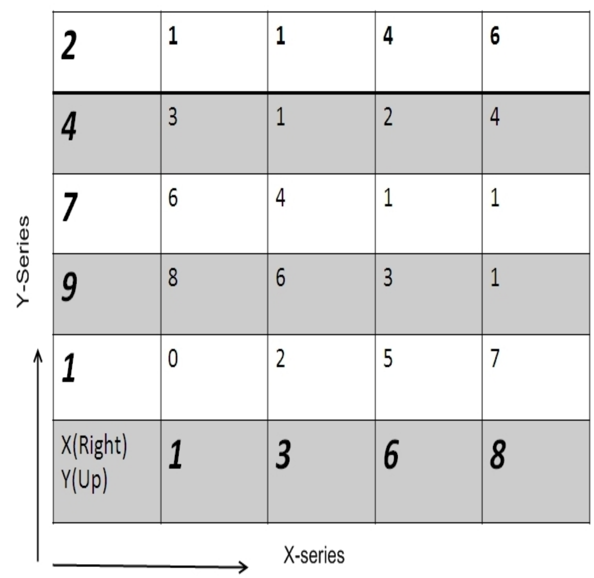
$$W = \{(4,5)(3,4)(3,3)(3,2)(2,1)(1,1)\}$$

and distance between X and Y series obtained is given below.

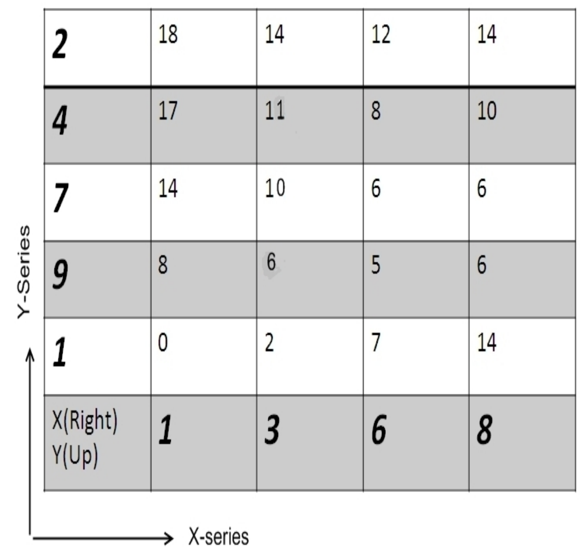
$$DTW(X,Y) = \sqrt[2]{\frac{14+8+6+5+2+0}{6}} = 2.415$$

2.7.3 Need of Local and Global Constraints

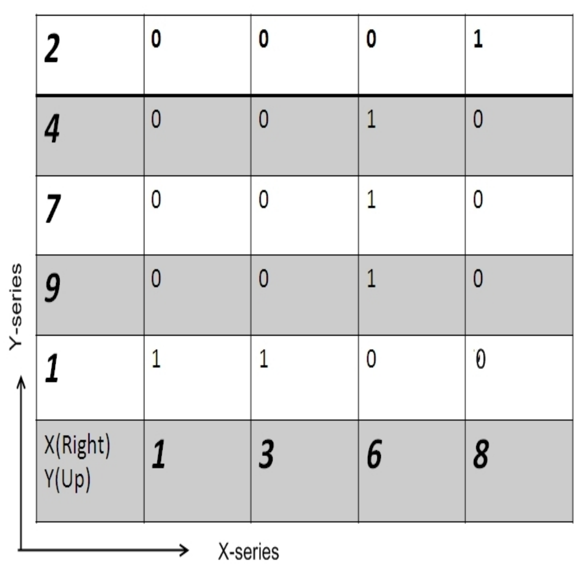
When there is no difference between the two series then the warping path coincides with the diagonal, but as differences between time axis increase, the warping path deviates more from the diagonal by matching similar time axis fluctuations. While DTW finds the optimal alignment of the time series, sometimes it creates an unrealistic correspondence between time series features such as by aligning very short features from the one of the series to the long features on the second time-series, leaving some features in one or both series unaligned and many to one alignment of some of the features. In order to avoid such phenomenons warping path is subject to constraints at each step. This constraints implemented as the possible relations between several consecutive points on the warping path. The constraints which check the local properties of warping path like continuity and monotonicity etc. are called Local constraints. Other common constraints impose on warping path are called Global constraints. These constraints not only speed up the DTW but also prevent



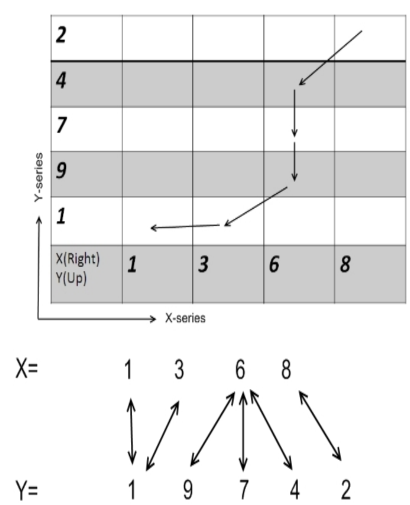
(a) Grid with Node distance



(b) Grid with cumulative distance



(c) Complete path through the grid



(d) Optimal Warping Path and Mapping

Figure 2.2: Example showing grid after Initialising , Recursion and Backtracking step

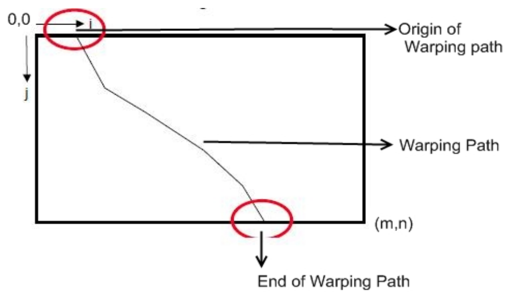
pathological warping by globally controlling the warping path. The Local constraints are discussed in Section 2.7.4 and Global constraint in Section 2.7.5.

2.7.4 Local Constraints

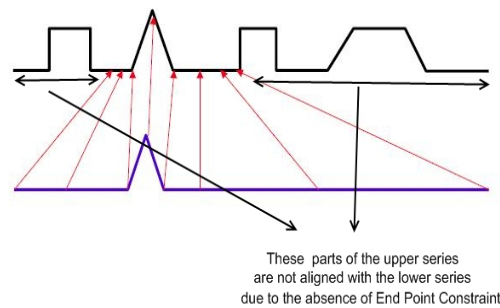
There are several local constraints which are implemented during recursion and backtracking step. These constraints make sure that the feature alignment between the two series is such that both the series should be aligned thoroughly, there should not be a miss or left of any feature in one of the series and there should not be the case where more than one feature is matched with the same feature of other series. Each of these constraints are discussed below.

2.7.4.1 End Point Constraints

This restriction makes the warping algorithm to have all the paths start at $(1, 1)$ (the first point of both the sequence) and end at the (n, m) (final point of both the sequence). In terms of path notation, having the complete path as $W = [w_1, w_2, .., w_K]$ then $w_1 = (1, 1)$ and $w_K = (N, M)$ means the warping path should start at the first node of grid and end at the other corner of the grid. In absence of this constraints there may be possibility of matching partial features between the two series. As shown in Figure 2.3(b).



(a) End Point Constraints

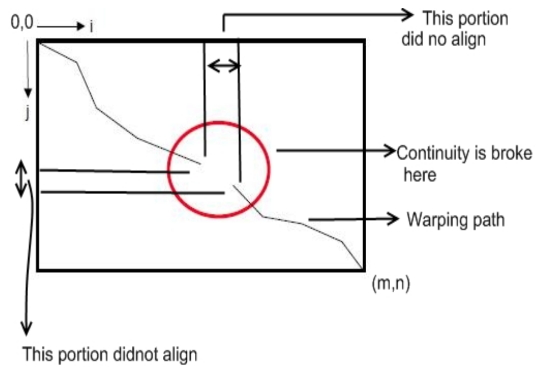


(b) Absence of End Point Constraints

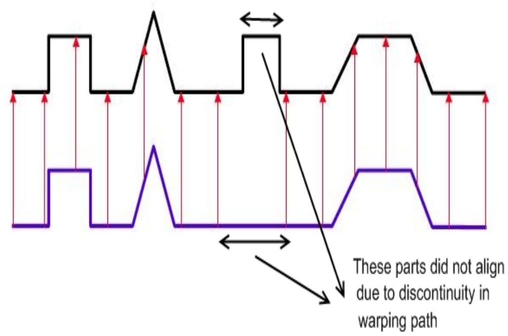
Figure 2.3: Importance of End point constraints

2.7.4.2 Local Continuity

This restriction ensures that there should not be any compression or expansion in one of the series due to the jump in the warping path. In the absence of this restriction there may be omission of some features which is shown in Figure 2.4(b). In terms of path notation, having the complete path as $W = [w_1, w_2, .., w_K]$ where $w_k = (i(k), j(k))$. It can be implemented by checking the condition $|i(k+1)-i(k)| \leq 1$ and $|j(k+1) - j(k)| \leq 1$ at every node in the warping path.



(a) Continuity Constraints

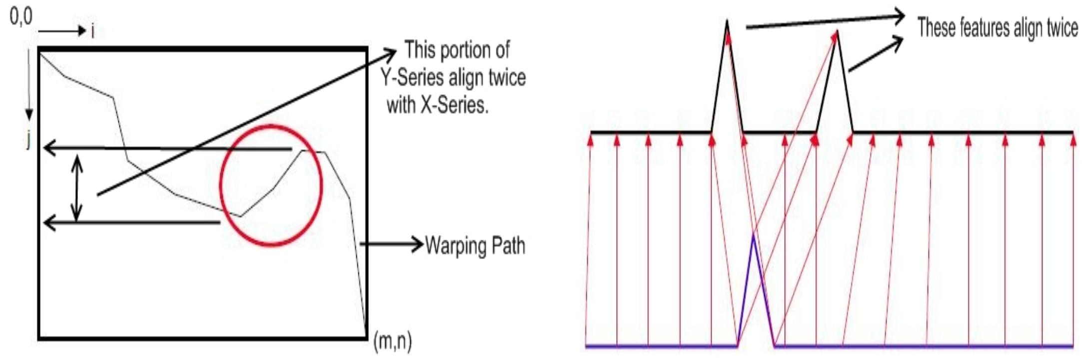


(b) Absence of Continuity Constraints

Figure 2.4: Importance of Continuity constraints

2.7.4.3 Monotonicity

Each node in the path can be expressed using two function i(k) and j(k). This constraint ensures that alignment path should not go back in time by making i(k) and j(k) monotonically increasing. In terms of path notation, having the complete path as $W = [w_1, w_2, .., w_K]$ where $w_k = (i(k), j(k))$. It can be implemented simply by checking the condition $i(k+1) \geq i(k)$ and $j(k+1) \geq j(k)$ at every node in the warping path. In the absence of monotonicity same feature of one of the series matches with more than one feature of the other series as shown in Figure 2.5(b).

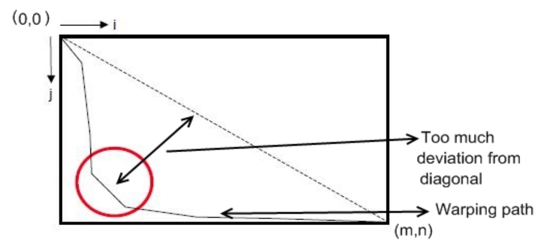


(a) Monotonicity Constraints (b) Absence of Monotonicity Constraints

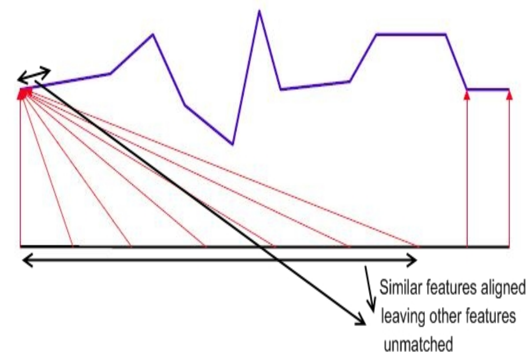
Figure 2.5: Importance of Monotonicity constraints

2.7.5 Global Constraints

As the computation cost of DTW is $O(nm)$ and algorithm require the storage of two grids of size m X n. In order to improve the the computational cost and optimize the DTW sensitivity global constraints have been introduced. Even after implementation of local constraints due to the too much alignment between similar feature, can make the optimal path deviate too much from the diagonal (Figure 2.6(a)) which results in the pathetic alignment between the features of the the two sequence (Figure 2.6(b)). Global constraint can restrict the deviation globally, can result in better feature alignment practically and as intended.



(a) Optimal Path Deviation from Diagonal



(b) Effect of the deviation of optimal path from diagonal

Figure 2.6: Global Optimal Path

2.7.5.1 Itakura Parallelogram

This constraint is given by Itakura [12]. It is shown in Figure 2.7(a). This constraint makes the warping path inside the limits of parallelogram. The recurrence relation (2.15) ensures the shape of warping as other half of the parallelogram either below or above the diagonal. Itakura constraints can be imposed by simply checking at every node $w_k(i,j)$ in the warping path, the following conditions.

$$1 + \frac{i(k)-1}{E_{max}} \leq j(k) \leq 1 + E_{max} * i(k-1) \tag{2.13}$$

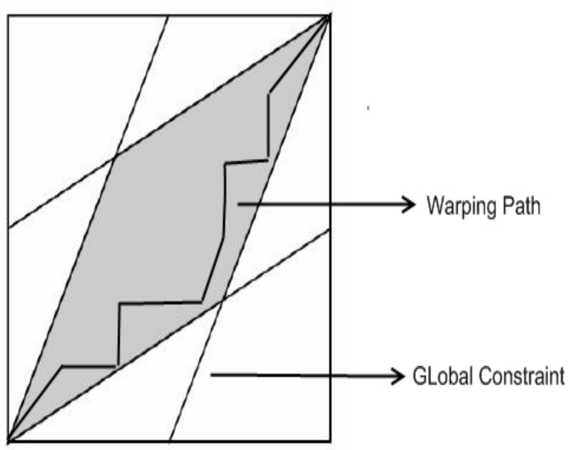
$$m + E_{max}(i(k)-n) \leq j(k) \leq m + \frac{i(k)-n}{E_{max}} \tag{2.14}$$

where m, n are the length of the two series and E_{max} is the slope of the one of the side of the parallelogram. For a given choice of E_{max}, the two pair of sides of parallelogram have slope of E_{max} and $1/E_{max}$. The equation of straight lines representing the sides of parallelogram are shown in Figure 2.7(b)

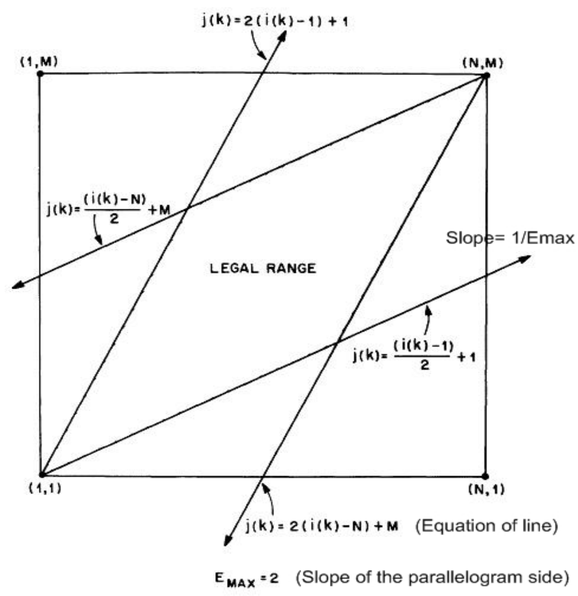
$$D_{Commu}(i,j) = min \begin{cases} D_{Commu}(i-1,j) + \alpha * D_{Node}(i,j) \\ D_{Commu}(i-1,j-1) + D_{Node}(i,j) \\ D_{Commu}(i,j-2) + D_{Node}(i,j) \end{cases} \tag{2.15}$$

where

$$\alpha = \begin{cases} \infty \ (i(k-1) \neq i(k-2)) \\ 1 \ (i(k-1) = i(k-2)) \end{cases}$$



(a) Itakura Parallelogram



(b) Global Path Range for constraint implementation

Figure 2.7: Itakura Parallelogram Implementation

2.7.5.2 Sakoe-Chiba Band

This constraint is given by Sakoe[23]. It is shown in Figure 2.8. It can be implemented simply by checking this $|i(k) - j(k)| \leq R$ condition for every node point in warping path. The recurrence relation which make the warping path quite near to the diagonal which is given below.

$$D_{Commu}(i,j) = min \begin{cases} D_{Commu}(i-1,j) + D_{Node}(i,j) \\ D_{Commu}(i-1,j-1) + D_{Node}(i,j) \\ D_{Commu}(i-1,j-2) + D_{Node}(i,j) \end{cases} \tag{2.16}$$

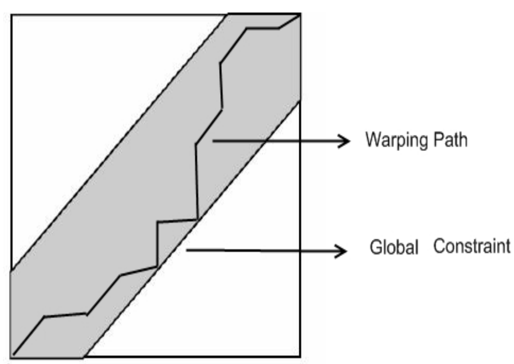


Figure 2.8: Sakoe-Chiba Band

2.7.5.3 Ratanamahatana-Keogh Band

Sakoe-Chiba band and Itakura Parallelogram apply constraint to the whole length along the diagonal which in some cases is able to give intended score to differentiate the two series of different class, but also gives bad mapping for the series of the same class. We need constraints which can produce mapping such that we can classify the series whether it belongs to the reference class or not. In short we need more adaptable constraints to the different length of the diagonal rather than same constraint along the diagonal. Ratanamahatana-Keogh have developed a iterative learning based algorithm for adaptable global constraints whose output generates the various sakoe band with different width around the diagonal as shown in Figure 2.9 is called as R-K band. It yields the best prediction accuracy by checking heuristic measure at every iteration. The detail of the algorithm is given in [21].

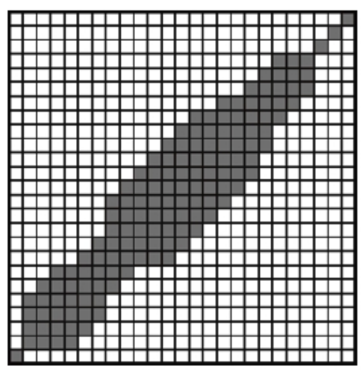


Figure 2.9: Ratanamahatana-Keogh Band

Chapter 3

Proposed System

The chapter proposes an online signature verification system. It takes the online signature which mainly contains the series of dynamics feature of the signature points captured as the user started signing on the digital pad. Each point on the signature represent a seven tuple namely *pressure, x-position, y-position, z-position, altitude, azimuthal and time*. Like any other biometric system it consist of five major modules and they are (i)Data Acquisition, (ii)Preprocessing, (iii)Feature Extraction, (iv)Training and learning and (v)verification. The various modules involved in the working of the system are explained below.

- **Data Acquisition** Data acquisition of such type of system plays an important role. Better the data better,one gets better accuracy of the system. in order to make user friendly a gui based software using Wintab API is developed in windows platform.

- **Preprocessing** The captured data undergoes through unusual process. The data captured may have variabilities due to the variation in starting position of signature, orientation of hand with respect to tablet and mental state of person (hurry, calm). These variabilities (noise) are to be normalised and removed before actual feature extraction.

- **Feature Extraction** After removing the noise and normalising the signature, features are extracted which are used in latter stages for verification and training. This module discusses those features which can be extracted from the online data.

- **Training and Learning** This module proposes a the novel learning approach for the implementation of adaptable global constraints. After the learning process, training is performed in which various parameters are obtained from the reference templates. Theses parameters are used in verification module.

- **Verification Approach** In this module, templates are extracted from the test biometric sample. These templates are matched to reference templates prepared during the training phase using the DTW with the global constraint. This section discusses fusion of feature vectors at score level and decision level for the final decision of the proposed system.

Each of the above modules has been discussed in detail in subsequent section .The proposed system is shown in Figure 3.1.

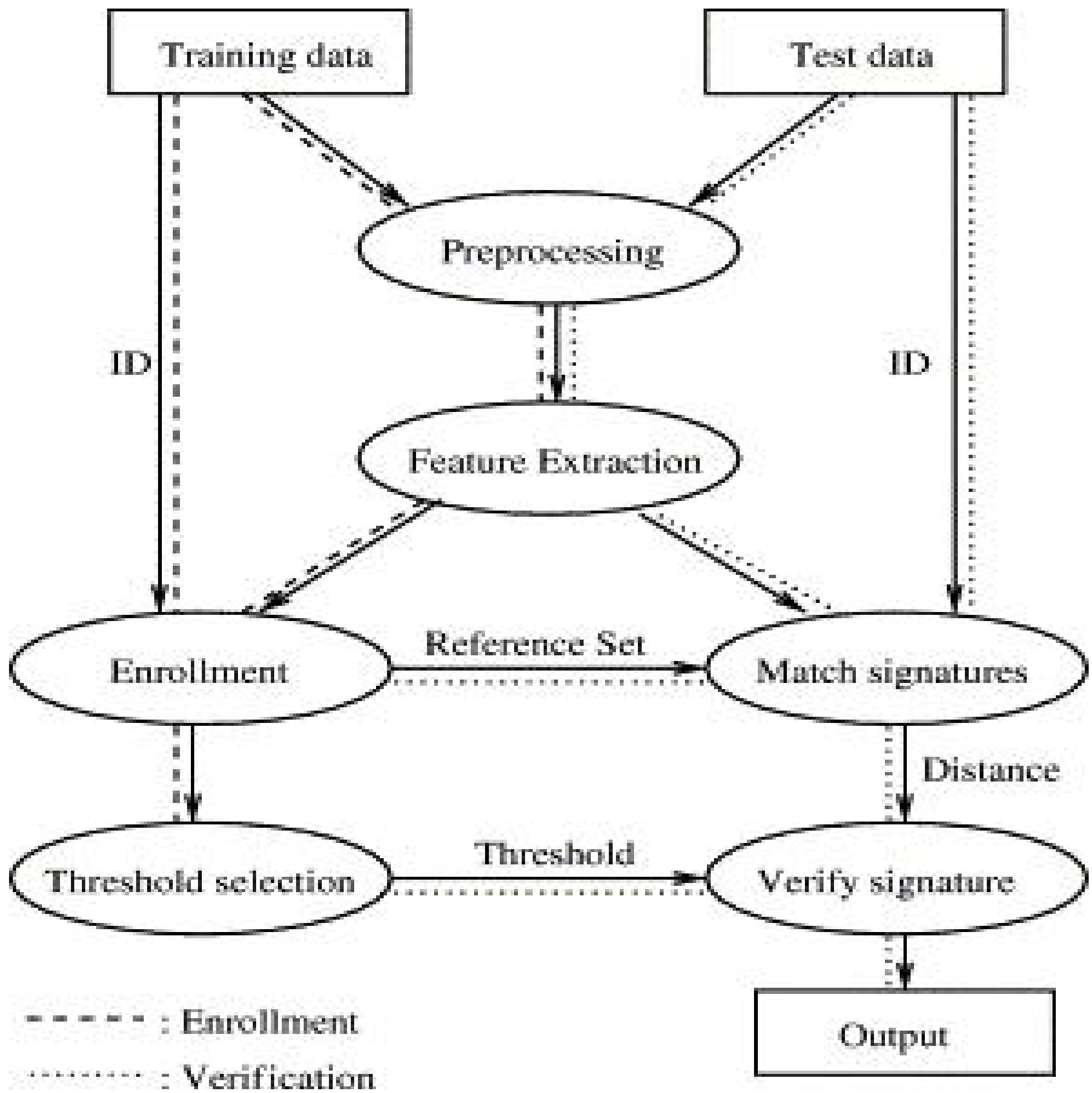


Figure 3.1: The proposed Signature verification System

3.1 Data Acquisition

The proposed system has been tested on two kinds of databases, viz. SVC2004 (signature verification competition) [4] and IITK database. in order to collect data we have used the Intuos2 model (XD 0405 R) with PS-2 interface graphic tablet from Wacom as data acquisition device. Developed system can display the real-time movement of the pen tip and tilt on the screen while user is writing the signature. This system has some of the features same as in commercially available software like SIGPLUS. Real time display is necessary for better quality of data acquisition as the signature involves both hand motion and visual decision in joining different strokes. Some of the graphic tablets have digital screen which can display in real-time what user is writing but they lack the pressure and tilt sensitivity which are crucial dynamics of signature. During acquisition phase, following basic dynamic data gathered at the time of signing :-

- At the moment, pen tip comes within the range of the tablet. X-coordinate of the pen tip according to the Tablet plane is captured for various points in order of time stamp.

$$X =< x_1, x_2, x_3, ...x_N > \tag{3.1}$$

- Similarly y-coordinate of the pen tip according to the Tablet plane is captured for various points in order of time stamp.

$$Y =< y_1, y_2, y_3, ...y_N > \tag{3.2}$$

- Similarly pressure inflicted by the pen tip on Tablet surface is captured for various points in order of time stamp.

$$P =< p_1, p_2, p_3, ...p_N > \tag{3.3}$$

- Altitude is the angle between the pen and the surface as shown in Figure 3.3. It is captured for various points in order of increasing time stamp.

$$L =< l_1, l_2, l_3, ...l_N > \tag{3.4}$$

- Azimuth denotes the clockwise rotation of the pen around the X axis as shown in Figure 3.3. It is captured for various points in order of increasing time stamp.

$$A =< a_1, a_2, a_3, ...a_N > \tag{3.5}$$

An example of signature captured by the Tablet device is shown in Figure 3.2.

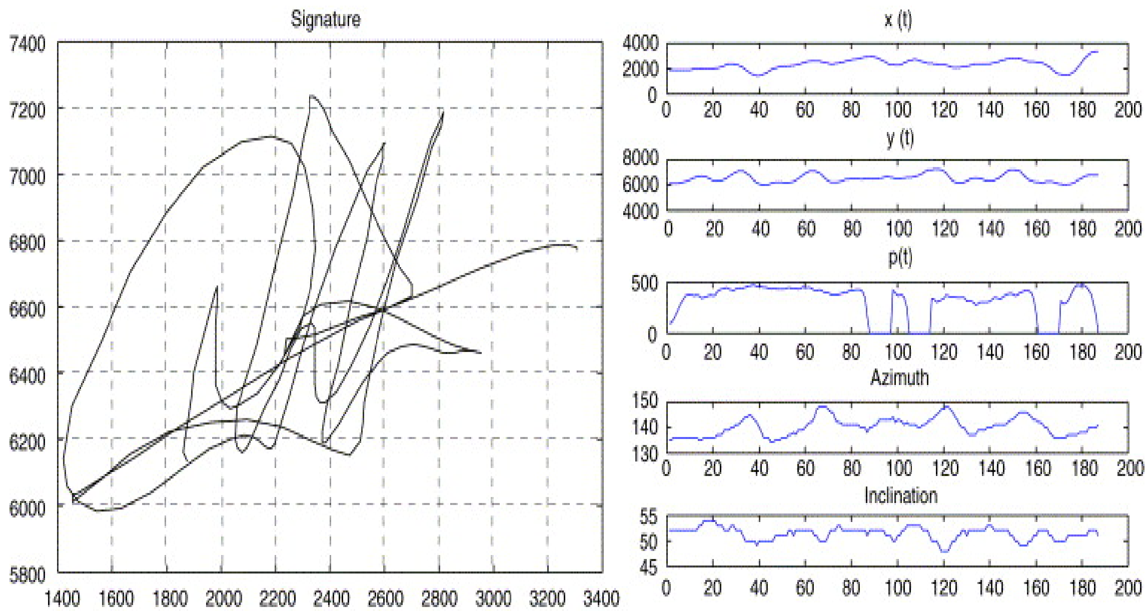


Figure 3.2: A sample signature captured by Tablet Device and plot of different dynamic features with time

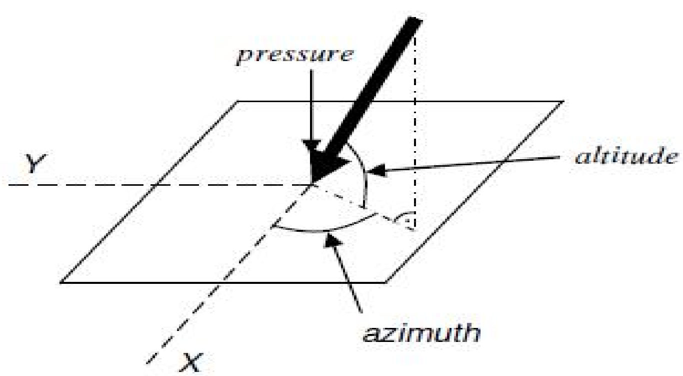


Figure 3.3: Diagram showing the semantics of dynamic features

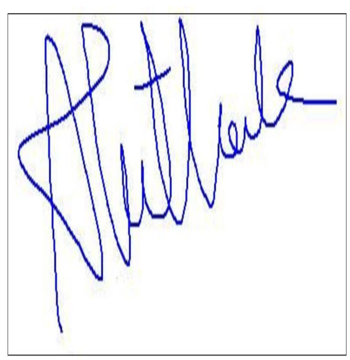
3.2 Preprocessing

Unlike other biometrics, Signature from the same person has too much intra class dissimilarities. As signature being a behavioural biometric generally it depends on the mental state of the person. It varies whenever he is in hurry or calm and can be observe in his hand movement. The noises which are originated due to the above reason are explained below.

- **Scaling**

  Sometime signatures made by the same user have quite different size even though they have same trajectory. It is also found that people signing in hurry usually makes a smaller sign to save time. An example of signatures having different sizes are shown in Figure 3.4.

(a) signature1

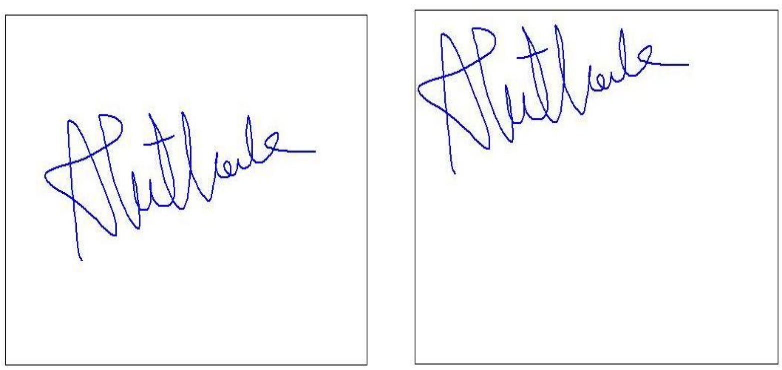


(b) signature2

Figure 3.4: Signatures with Different Scaling

- **Translation**

  There is always a possibility that signatures done by user differ in the starting position. Due to the difference in the starting coordinate of signature, it may result in the significant changes in the position of subsequent points captured from the Digital Tablet. An example of this kind of noise is shown in Figure 3.5.



(a) signature1 (b) signature2

Figure 3.5: Signatures with Translation Effect

- **Rotation**

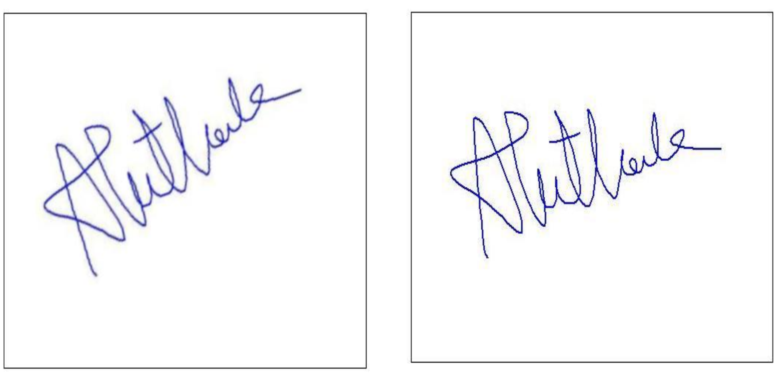
  Due to change in the orientation of Hand or Digital Tablet, signature major variance axis have different slope. In our case during data acquisition time we request user to sign again in case the signature orientation differ too much. An example is shown in Figure 3.6.

Due to the above noise generated, it may affect the performance of system in later stages, so it is necessary to remove them. Their are several techniques to remove these kind of noises. In this system each signature is processed with following techniques to remove above mentioned noises:-

- **Z Score Normalisation**

  To make signature invariant to scaling and translation all x and y values of the signature is normalised with mean -standard deviation normalisation. After this normalisation the mean of the data becomes zero and standard deviation becomes one. That means, data is centred at the origin and all the points are scaled to same range.

$$X_{new}(i) = \frac{X_{old}(i) - \mu}{\sigma} \tag{3.6}$$



(a) signature1 (b) signature2

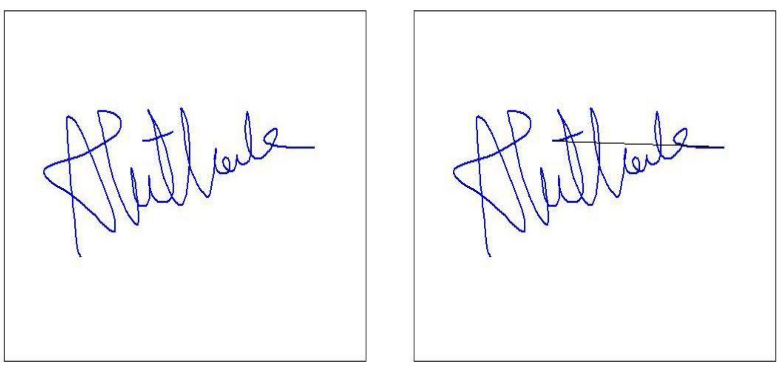
Figure 3.6: Signatures with Rotation Effect

$$Y_{new}(i) = \frac{Y_{old}(i) - \mu}{\sigma} \tag{3.7}$$

where μ is average of all the points in the respective series and σ is the standard Deviation of all the points in the respective series.

- **Stroke Concatenation**

  At the time of signing, pen up movement is also captured by the tablet. We need to remove these points as they do not contribute for the signature trajectory. All the points between pen down point to pen up point contributes to single stoke. To facilitate the trajectory matching all storks are concatenated to form long strings as shown in Figure 3.7.



(a) signature1 (b) signature2

Figure 3.7: Signatures with and without Stroke Concatenation

- **Smoothing and Arch length Normalisation**

  End points of stroke and points where trajectory changes carry more information than any other points, theses points are called critical points. Re-sampling of these points discard important information about speed and trajectory which results in poor accuracy in case of skilled forgeries. This fact is also proved by [15].

3.3 Feature Extraction

Feature are mainly classified into static and dynamic. Static features include (geometry and shape of signature) while dynamic features contains information about its acceleration, velocity and trajectory profiles. Static features can be further divided into local and global features. Local features are specific to some part of the signature while on the other hand global feature are obtain from the signature as a whole. Based on the literature survey, total number of features considered in this thesis is eight out of which three are the basic dynamic features captured during data acquisition and rest are prepared from them. Theses features are explained below

- Pressure series consist of pressure captured at various points from the beginning to the end of the signature. This feature is just separated from the normalised data.

- Altitude series consist of angle made by the pen axis with the vertical at various points from the beginning to the end of the signature. It is also captured during data acquisition.

- Trajectory series consist of two tuple points each containing (x, y) where the pressure captured is not zero, ie there exist only pen down movements. It is extracted from the original data after preprocessing step.

- Curvature at any point of the trajectory is given by the following relation.

$$\kappa = \frac{x'y" - y'x"}{(x'^2 + y'^2)^{\frac{3}{2}}} \tag{3.8}$$

It is calculated at every point and is made series consist of curvature points from the beginning to end of the signature.

- Velocity series consists of velocity vector(v_x, v_y) at all the points in the signature in order the points are captured by digital tablet.

$$v_x(i) = \frac{\Delta x_i}{\Delta t_i} = \frac{x_i - x_{i-1}}{t_i - t_{i-1}} \tag{4.8a}$$

$$v_y(i) = \frac{\Delta y_i}{\Delta t_i} = \frac{y_i - y_{i-1}}{t_i - t_{i-1}} \tag{4.8b}$$

- Absolute velocity series consists of magnitude of the velocity vector at all the points.

$$v(i) = \sqrt[2]{v_x^2(i) + v_y^2(i)} \tag{3.9}$$

- Acceleration consists of acceleration vector at various points. It is calculated as follows:-

$$a_x(i) = \frac{\Delta v_x(i)}{\Delta t_i} = \frac{v_x(i) - v_x(i-1)}{t_i - t_{i-1}} \tag{4.9a}$$

$$a_y(i) = \frac{\Delta v_y(i)}{\Delta t_i} = \frac{v_y(i) - v_y(i-1)}{t_i - t_{i-1}} \tag{4.9a}$$

- Absolute acceleration Series consists of magnitude of the acceleration at various points

$$a(i) = \sqrt[2]{a_x^2(i) + a_y^2(i)} \tag{3.10}$$

3.4 Learning

This thesis proposes a learned global constraint algorithm based on some modification on the R-K Band Algorithm [19]. R-K band algorithm is developed in recent year and global constraints are giving good results when using for time series classification. When applied with DTW, global constraint of a particular user helps in giving intended scores that can correctly classify the signature of different users. Global constraint also decreases the heavy computation of DTW, as only the computation are done for points which are within the constraint in the grid. This module explains how RK-band global constraint is obtained for each user in the database. The proposed modifications and detail learning algorithm is explained further in this section.

3.4.1 Proposed Modification to RK-band global constraint

To make RK-band global constraint applicable in the field of signature biometric, the proposed approach suggested some changes in the original. The proposed approach differs from the original in following sense:-

- **Dataset**

  Although the author [21] has tested the R-K band algorithm on various data set such as (stock, medical data ..etc). It has never been used in the field of online signature biometric. The proposed algorithm is tested on the signature database.

- **Variable length series**

  The original RK-Band algorithm is very much dependent on the length of the series. Author has claimed that they use normalisation for variable length of the series like interpolation and spline techniques. It was shown in this thesis that normalisation techniques degrade the performance of online signature verification. When the length of the series are not normalised then they will result in different size grids in DTW , hence the slope and length of the diagonal will change. In order to make this algorithm applicable despite length normalisation we need to rotate the global constraint according to the different slope of the diagonal in the grid for the every DTW calculation. As the rotation can severely degrade the response time of DTW. Unlike other global constraint such as Itakura and SC-Band are stored as the equation of the straight lines, the proposed algorithm stores the global constraints in rotation invariant form as shown in Figure 3.8. Implementation point of view rather than storing the global constraint as the different straight lines it is stored as the list of nodes where each node is SC-band having two fields first the distance from the diagonal and second, the length of the constraint.

- **Heuristic Measure**

  The original RK-Band algorithm is learning algorithm that check some heuristic parameter for every iteration to do optimisation. Original algorithm used the Silhouette index as the optimisation parameter. The Silhouette index for each data i in the database is defined by the following equation:

$$s(i) = \frac{b(i) - a(i)}{max(b(i), a(i))} \tag{3.11}$$

  where $b(i)$ is the minimum average distance between i^{th} data and each of the different cluster data and $a(i)$ is the minimum average distance between the i^{th} data and each of the same cluster data. Thus, $s(i)$ lies between -1 to 1. Greater the value of $s(i)$, better separated the different cluster data. This measure is useful when the number of classes are very less but this is not practical in signature biometric. Due to above reason, proposed system have used the variance of the DTW scores obtained by the signature samples of same class as the heuristic measure which has performed well as shown in result section.

- **Iteration**

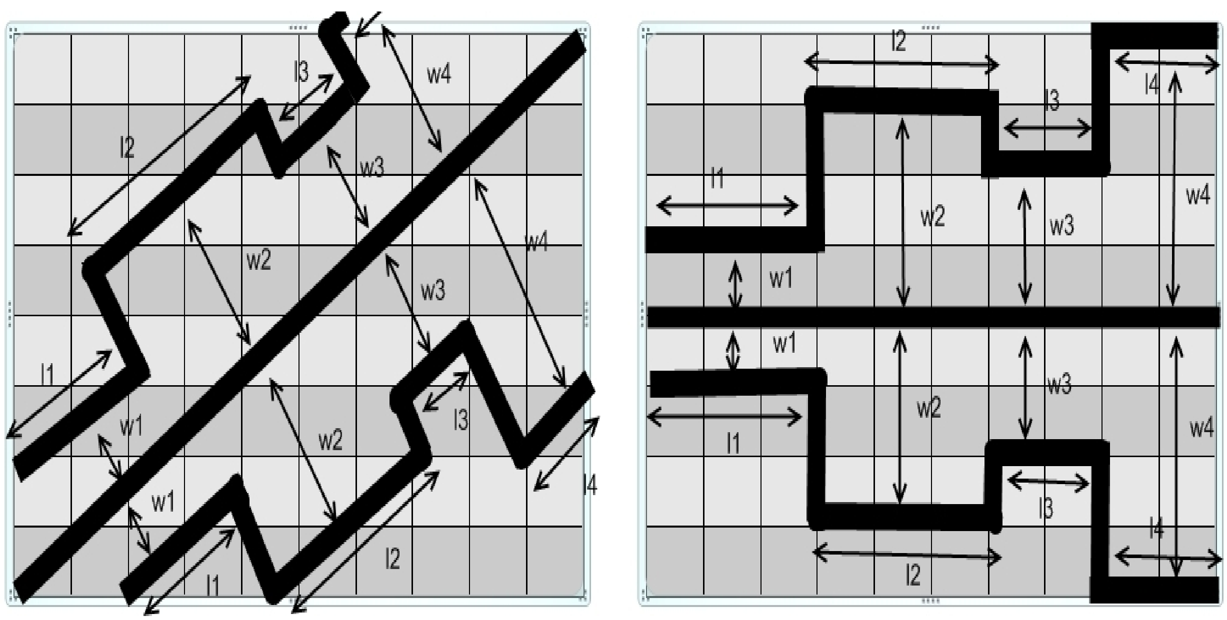
  To make the algorithm converge faster rather than using constant factor to modify parameters, we modify them exponentially in the factor of two before every iteration as compared to original approach,. It makes the proposed algorithm giving satisfactory result in lesser time.

- **Implementation Issues**

  Implementation of global constraints comes in the field of computational geometry. Sometimes algorithms which are quite simple and trivial like triangulation, can become extremely difficult to implement. As the global constraint is not uniform we cannot implement some simple relation such as Itakura Section2.7 and Sakoe-Chiba Band Section 2.8. Global constraint is shown in Figure 3.8. Original approach [21] is silent about the implementation issues so a new approach is proposed to implement it. The detail of the implementation is explained along with pseudo code under the headline "constraint implementation" in next section.

3.4.2 Proposed Learning Global Constraint Algorithm

The proposed approach work in four major modules namely (i) Max Deviation (ii) Modifying Global Constraint (iii) Evaluate heuristic and (iv) Constraint Implementation. Algorithm start with the "Max Deviation" module which obtain the initial global constraint from the training signatures. The initial global constraint is given as input to the main module "Modifying Global Constraint" which is the main loop that updates the global constraint. Every iteration of the second modules checks the heuristic measure obtained from the third module "Evaluate heuristic". Third mod-



(a) Global Constraint

(b) Rotated Global Constraint

Figure 3.8: Global Constraint Implementation

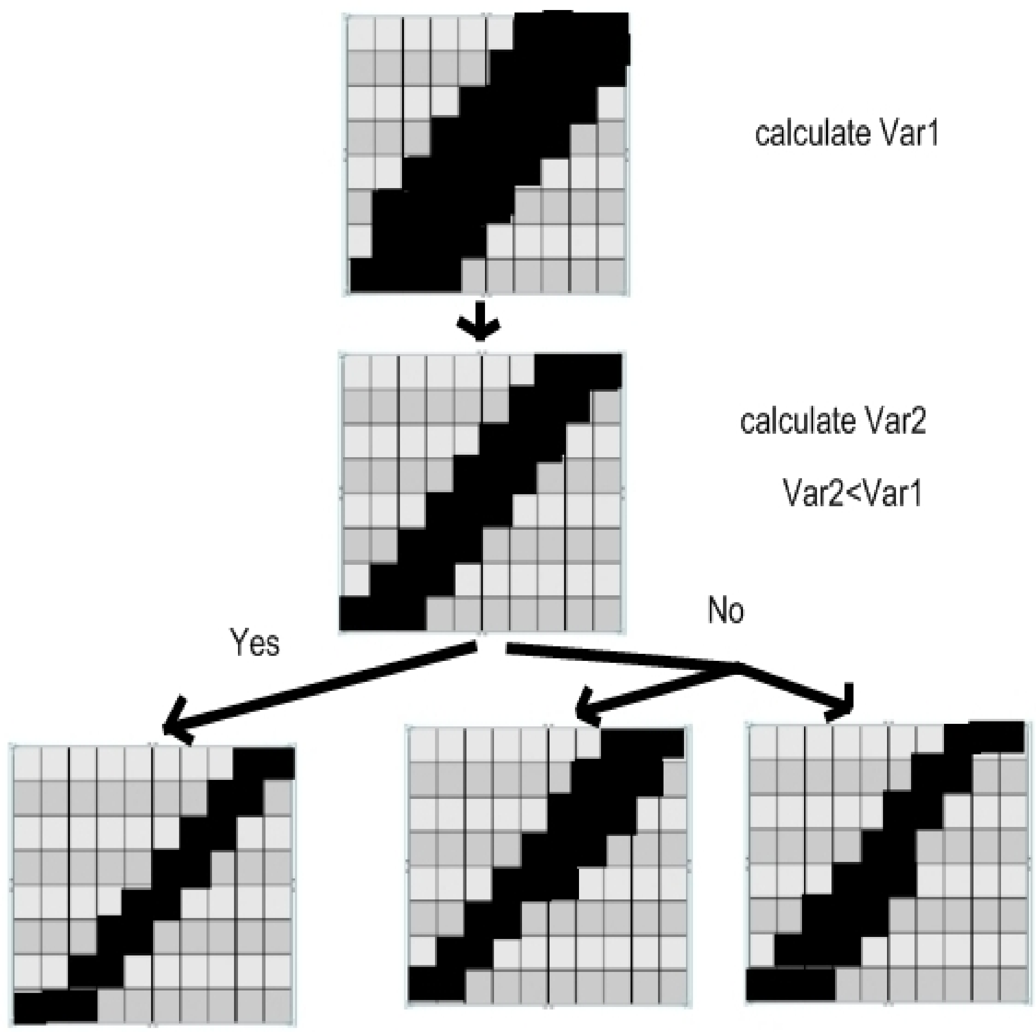


Figure 3.9: Algorithm for modifying Global Constraint

ule calculate heuristic (variance) of data by evaluating the DTW with current global constraint of the training signatures among themselves. Each of these modules are explained below.

- **Max Deviation**

  It is the starting point of the proposed algorithm where we find the initial global constraint. The initial global constraint has the shape of a single Sakoe Band whose length is same as the diagonal and width being the maximum deviation of warping path with global constraint. This algorithm find the maximum width and diagonal of the warping path for the signatures of particular user among all the warping paths produced by the signature sample of same user among themselves. Final output of this algorithm is used as initial parameter for "Modifying Global Constraint" algorithm.

  Function[width, DiagLength]= MaxWidthAndDiagonal(D_i**)**

```
maxDiag , currDiag , maxWidth ,currWidth
for all warping path in D
     Let w=current warping path in D
          for all node points in warping path in w
              currWidth=calculate node point distance from the diagonal
                  if (currWidth >maxWidth)
                        maxWidth=currWidth
                  end
                  currDiag=Current Warping path length
                  if (currDiag >maxDiag)
                      maxDiag=currDiag
                   end
          end for loop
end for loop
```

- **Modifying Global Constraint**

  This algorithm start with a given global constraint and try to modify it whenever the new variance of DTW scores for signature sample using updated global constraint decreases. As variance is decreases for every modification of global constraint, that means the scores obtained by DTW for the signature of same user are more discriminating than scores obtained with the alignment of other users signatures. Due to this reason it can improve the performance measure of the verification system. Final output of the algorithm is the global constraint for which the signatures of the same user have minimum variance. From the implementation point of view first we make a queue and insert the global constraint obtained from "Max Deviation" algorithm. We decrease the width of global constraint as long as variance decreases. As soon as that point is reach where variance no longer decreases we divide this global constraint into two equal parts and insert each parts again in the queue. The same procedure is applied to the queue till it becomes empty. we do not insert any new element in the queue under following two conditions:-

  (1) Length of one of the parts of global constraint on further division becomes less than certain threshold.

  (2)Width of some part of global constraint on further decrement becomes less than certain threshold.

  An iteration of this algorithm is shown in Figure 3.9 and the pseudo code is given below

Function[B]= GlobalConstraint(D_i)

```
maxDiag , maxWidth=MaxWidthAndDiagonal(D)
enqueue(1,maxDiag,maxWidth)
 while(!empty(Queue))
           [start, end, current_width] =  dequeue(Queue)
            \\adjust function checks the 2 conditions described above
            adjustable = adjust( start, end, current_width);
             if (adjustable)
                       heuristic = EvaluateHeuristic(T, D);
                      if(heuristic > best heuristic)
                               best heuristic = heuristic;
                               enqueue(start, end, label, Queue);
                        else
                              undo adjustment( start, end,current_width);
                              if( (start  end) < threshold)
                                    enqueue(start, mid-1, label, Queue);
                                     enqueue(mid, end, label, Queue);
                               endif
                        endif

             endif
 endwhile
```

- **Evaluate heuristic**

This section explains the algorithm which calculate the heuristic measure at every iteration of the "Modifying Global Constraint" algorithm and return the variance of the training signatures using the modified global constraint.

Function[B]= EvaluateHeuristic(T, D)

```
\\ calculate the distance measure of all the samples
\\ of the class using current global constraint and
\\ return the variance of these values
```

```
for all i in Training sample
    for all j in Training sample and i!=j
     D(i,j)=DTW(i,j,current global constraint)
    end
    Add D(i,j) to Array VarData[]
end
return variance(VarData[])
```

- **Constraint Implementation**

  Global constraint is implemented as list of nodes where each node is a Sakoe band having different length and width. For every DTW evaluation a grid mask is created using global constraint in which all points inside the global constraint is set to 1 and those out side the grid as 0. After creating the grid mask, DTW is only evaluated on the node points for which the grid mask is 1. This fact makes the warping path inside the global constraint and also saves the time in DTW evaluation. Grid mask is obtained by rotating the grid and then evaluating grid mask as written below and further transforming the points to original orientation.

  Function[B]= ConstraintImplementation(L)

```
L //Global constraint stored in the form of list
while((L->Next)!=NULL)
{
  Node * tmp =L
  l1 = tmp ->l
  w1 = tmp ->w
  for( i=0 to l1 )
   {
     for(j=0 to w1)
       {
```

```
            grid[i][j]=1
          }
    }
    L=L->Next
}
```

3.5 Verification Approach

This section explains how the system developed in this thesis is used to classify the (test) signature as genuine or imposter. The verification protocol for the proposed DTW based system is quite similar for Random and Skilled Forgeries.

3.5.1 Overview

To verify that the whether the claim signature belongs to specific writer, it is first compare to the certain number of reference templates of writers signature. A threshold is decided based on the statics of the claimed writers signature and a dissimilarity score is calculated based on the test and model signature. When this dissimilarity score is less than threshold it is accepted as genuine writer otherwise rejected as imposter(fraudulent) user.

3.5.2 DTW based verification

The verification approach is based on the ideals that in a normal distribution 68% population are within $\mu \pm \sigma$ and 95% within $\mu \pm 2\sigma$. Due to the scores obtained by proposed approach are quite discriminating, the simple Mean-Standard based verification is giving satisfactory results. First from the database, few genuine signature of particular user are selected as reference (training signatures) templates. We calculate the average μ and the standard deviation σ of minimum of DTW distance of all these training signatures among themselves as below.

$$\mu = \frac{\sum_{i=1}^{N} min\{D(T_i, T_j), \forall i \neq j \exists T_i, T_j \in T\}}{N} \tag{3.12}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} \left(min\{D(T_i, T_j), \forall i \neq j \exists T_i, T_j \in T\} - \mu\right)^2}{N}} \tag{3.13}$$

where T is the set of training signatures and $N = |T|$ i.e. number of training samples. After that the test signature is compared with each of the reference signatures and among all the scores obtained minimum is taken as the distance measure (*TestScore*) for the test signature. The best threshold (Thr) is chosen by varying it, which correctly classifies most of the test signatures as genuine or imposters according to following rule:-

$$TestSignature = \begin{cases} Genuine, & \mu - Thr * \sigma \leq TestScore \leq \mu + Thr * \sigma \\ Imposter, & otherwise \end{cases} \tag{3.14}$$

Verification based on decision and score level fusion of 8 features is explained as follows:-

- **Score Level Fusion:**

  We have certain number of signatures of each user. From each signature first we extract features and then normalise them in range of [0,1] using MinMax normalisation. Feature vector extracted from the K^{th} signature is given by $F^k =< f_1^k, f_2^k, f_3^k, ..., f_8^k >$. Let $min\{f_i\}$ be the minimum of the i^{th} feature of all the signatures of the particular user and $max\{f_i\}$ be the maximum of the i^{th} feature of all the signatures of the particular user. The the normalised feature f_i' is defined as

$$f_i' = \frac{f_i - min\{f_i\}}{max\{f_i\} - min\{f_i\}} \tag{3.15}$$

where $f_i' \epsilon [0, 1]$ and $F' =< f_1', f_2', f_3', ..., f_8' >$

when all the features of all the signature samples of a particular user is normalised. We obtain a single score corresponding to all the feature using PCA (principal composite analysis) [3] which reduces dimensionally the 8 dimensional feature vector to single dimension along which the data has maximum covariance among all signature. The verification is done using Equation 3.14.

- **Decision Level Fusion:**

  Here as compared to previous approach verification is applied based on each feature extracted from the signature rather than final score obtained from all the features. A final verification is done based on the number of features matched individually. Decision on a acceptance is taken if the number of feature is more than a threshold which varies from one to eight. We can not show the performance curves for the final decision but we can show the contribution of individual feature in verification by displaying individual DET curve per feature.

Chapter 4

Experiment And Results

4.1 Database

IITK Database has 70 users each having 10 signatures and SVC 2004 has 40 users each having 20 genuine and 20 skilled forgery signatures. System developed in this thesis is tested for random forgery on IITK and SVC2004 database. The skill forgery is only tested on the SVC2004 Database .

4.2 Experiment

Three experiment have been conducted in this thesis as part of evaluation of the proposed system. Two experiments are conducted for random forgery in IITK and

SVC database. Skill forgery experiment is conducted only on SVC database. Various metrics like database size, number of training samples and imposter/genuine attempts used in the three experiment is tabulated below.

Random Forgery			
Database	Training Signatures	Genuine Signatures	Imposter signatures
IITK	630	70	70
SVC2004	400	400	800

Skill Forgery			
Database	Training Signatures	Genuine Signatures	Imposter signatures
SVC2004	400	400	800

- Experiment1: Random Forgery in IITK database

  For both the approaches 9 training signatures are chosen as reference templates. After extracting the eight features from each of the signatures. They are aligned among themselves using proposed DTW with global constraint and normal DTW. For each of the training signature the minimum of the scores obtained while aligning with other training signatures is chosen as a new similarity score for that signature in user cluster. This similarity score is calculated using each of the eight feature per signature. Finally a eight dimensional similarity vector for each training signature is obtained. Average and standard deviation of each dimension of the similarity vector of all the training sample is calculated which is later used for verification of test signature. For performance evaluation there is one genuine and imposter signature per user id. These test signatures are matched with each of the nine reference signatures and a score vector is obtained on a per feature alignment basis. The minimum of the nine scores values obtained by the test signature for each of the eight feature is taken as a similarity vector for the test signature. The test signature is classified as genuine or imposter by verifying this similarity vector with the obtained average and standard deviation during the training phase as explained in previous chapter. After evaluating all the users in the database DET curves are plotted.

- Experiment2: Random Forgery in SVC database

  In SVC database each user has 40 signatures out of which 20 are genuine and rest 20 are skilled forgery signature. For random forgery we take first 20 as genuine attempts and the last 20 signatures of some other user id as imposter attempts. The training and verification approach is same as above.

- Experiment3: Skilled Forgery in SVC database

  For Skilled forgery we take first 20 as genuine attempts and the last 20 signatures of same user id as imposter attempts. The training and verification approach is same as in experiment 1.

4.3 Results analysis in IITK Database

The experiments on IITK database are conducted using the normal DTW and proposed DTW having global constraint. DET curve for the performance of verification system using the above approaches is shown in Figure **??**. Both the systems are tested with eight features which are explained in the previous chapter. The DET curve shown in Figure 4.1 is plotted using verification based on score level fusion of all the eight features as explained in Section 3.5.2. The Figure 4.2 and Figure 4.3 are plotted using decision level fusion, while accounting individual features for verification by the above mentioned two approaches. From all the figures it visible that the proposed approach has lower EER than the normal DTW. DET curve shown in Figure 4.2 and Figure 4.3 of Altitude , Trajectory and Velocity of two approaches are crossing each other while in the rest of the plots their is no crossing between the curves. DET curves for Pressure and Absolute Velocity have the lowest EER rate which is less than 5% among all the eight features for the proposed approach. DET curve for Absolute Velocity feature has minimum value of EER (3.58%) while Acceleration feature has the maximum value of 10.67% EER for the proposed approach. The DET curve for Altitude has the maximum difference of 15% EER between the proposed approach and normal DTW. From the various DET curves of individual features it is clear that Pressure and Absolute velocity are maximum, while Acceleration is minimum and rest are moderate discriminating features in IITK online signature database. By comparing the combined feature DET plot shown in Figure 4.1 with the DET curve of Absolute Velocity Figure4.3, it can also be concluded that Score level Fusion has performed better than the any individual feature.

Performance Measure of Random forgery on IITk Database		
Algorithm	Maximum Accuracy	EER
Proposed Algorithm (W=5)	96.96%	3%
Proposed Algorithm (W=10)	96.21%	4%
Normal Dynamic Time warping Algorithm	94.69%	5%

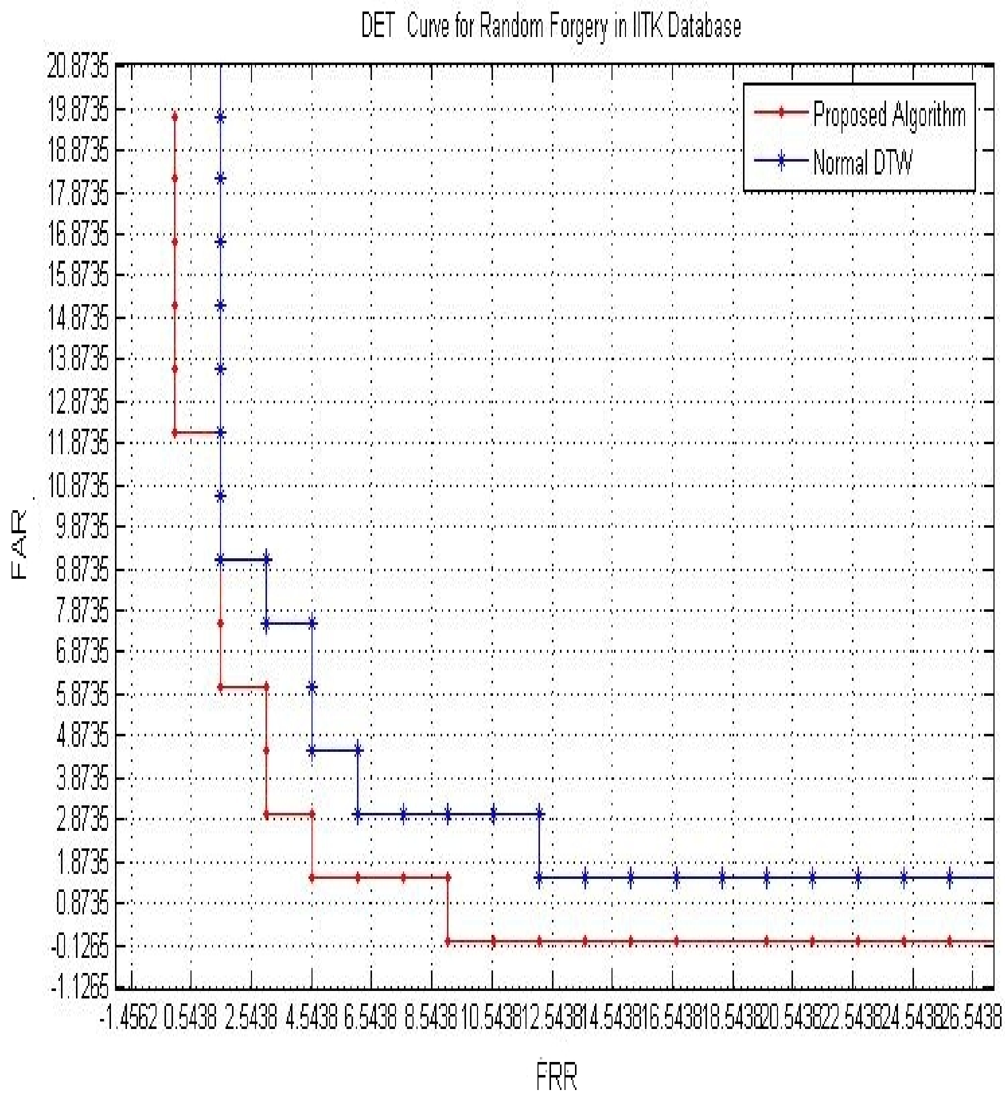


Figure 4.1: DET Curve for Random Forgery in IITK Database using Combined Features

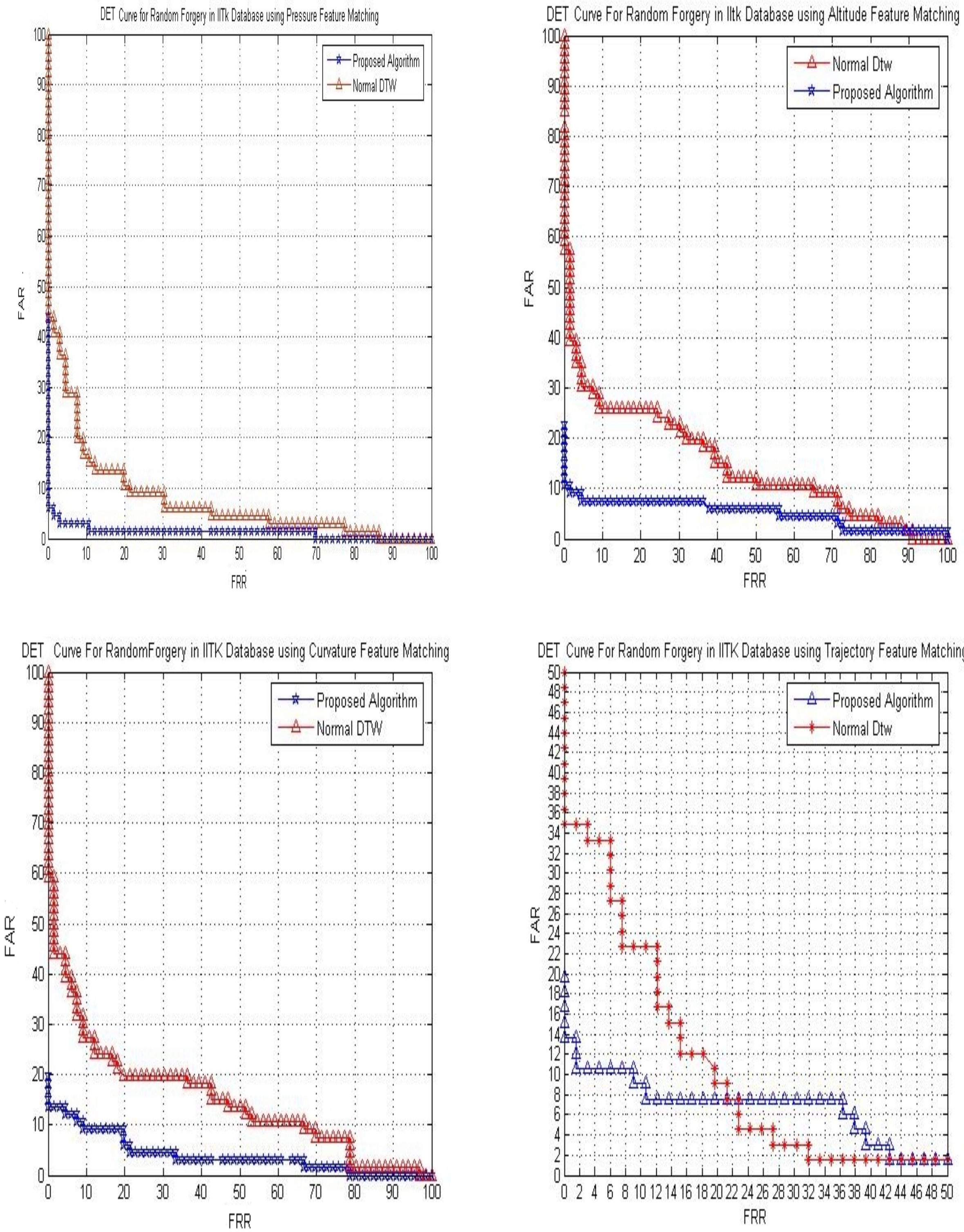


Figure 4.2: DET Curves for Random Forgery in IITK Database with individual Features

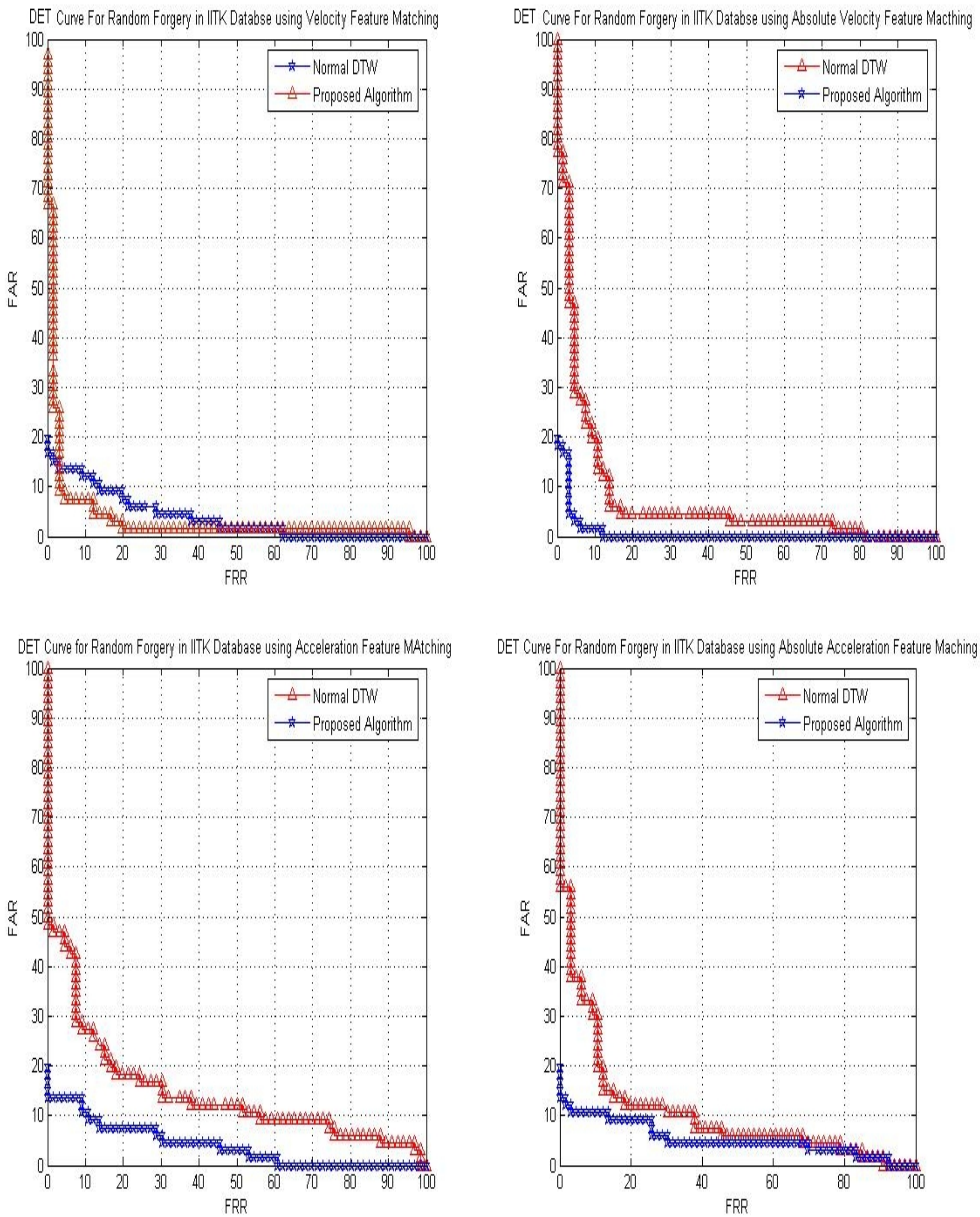


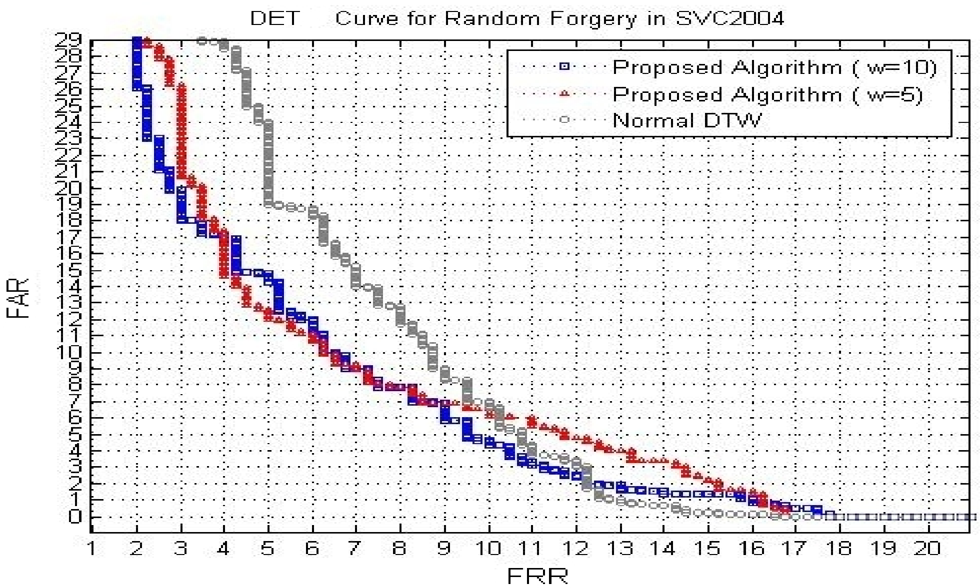
Figure 4.3: DET Curves for Random Forgery in IITK Database with individual Features

4.4 Results analysis in SVC2004

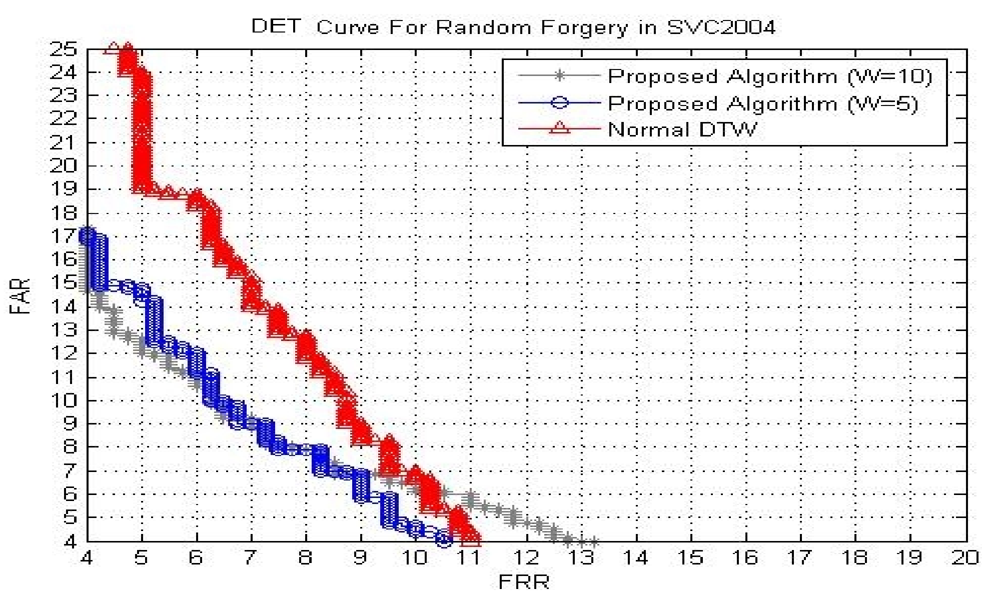
The combined DET curve obtained after accounting all features in the case of SVC2004 database is shown in Figure 4.4. The other DET curves using individual feature tested on SVC database are shown in Figure 4.5 and Figure 4.6. As depicted from the DET plots, Trajectory, Velocity and Absolute Velocity features have EER less than 10%. Velocity feature has the maximum difference between the EER of the proposed approach and normal DTW while Trajectory feature has the lowest difference between the EER of the two approaches. It can also be concluded from the DET curves of the combined features shown in Figure 4.4 has EER of approximately 7% and Velocity DET curve shown in Figure 4.6 has EER of 4%, hence in this case Score level fusion has not performed better than one of the individual feature. The above result may be caused due to the bad performance of other features.

Performance Measure of Random Forgery in SVC Database		
Algorithm	Maximum Accuracy	EER
Proposed Algorithm (W=5)	94.5%	7.8%
Proposed Algorithm (W=10)	93.75 %	8%
Normal Dynamic Time warping Algorithm	90.08%	9.8%

Performance Measure of Skill Forgery in SVC Database		
Algorithm	Maximum Accuracy	EER
Proposed Algorithm (W=5)	82.167%	18.7%
Proposed Algorithm (W=10)	81.75 %	19%
Normal Dynamic Time warping Algorithm	79.16%	21%



(a) Whole plot



(b) Zoom on the same plot near the origin to visualise EER

Figure 4.4: DET Curve For Random Forgery in SVC2004 using Combined Features

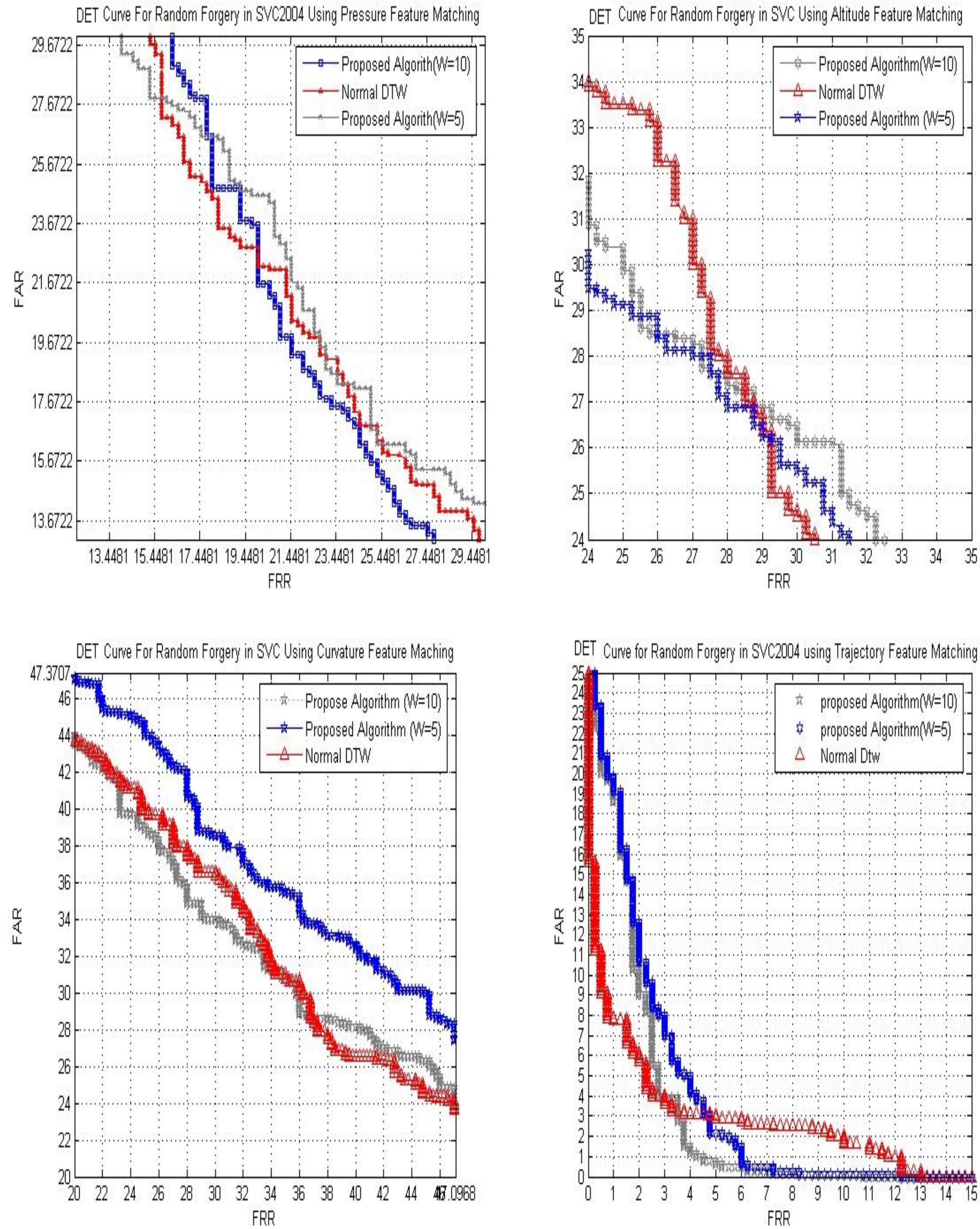


Figure 4.5: DET Curves for Random Forgery in SVC20004 Database with individual Features

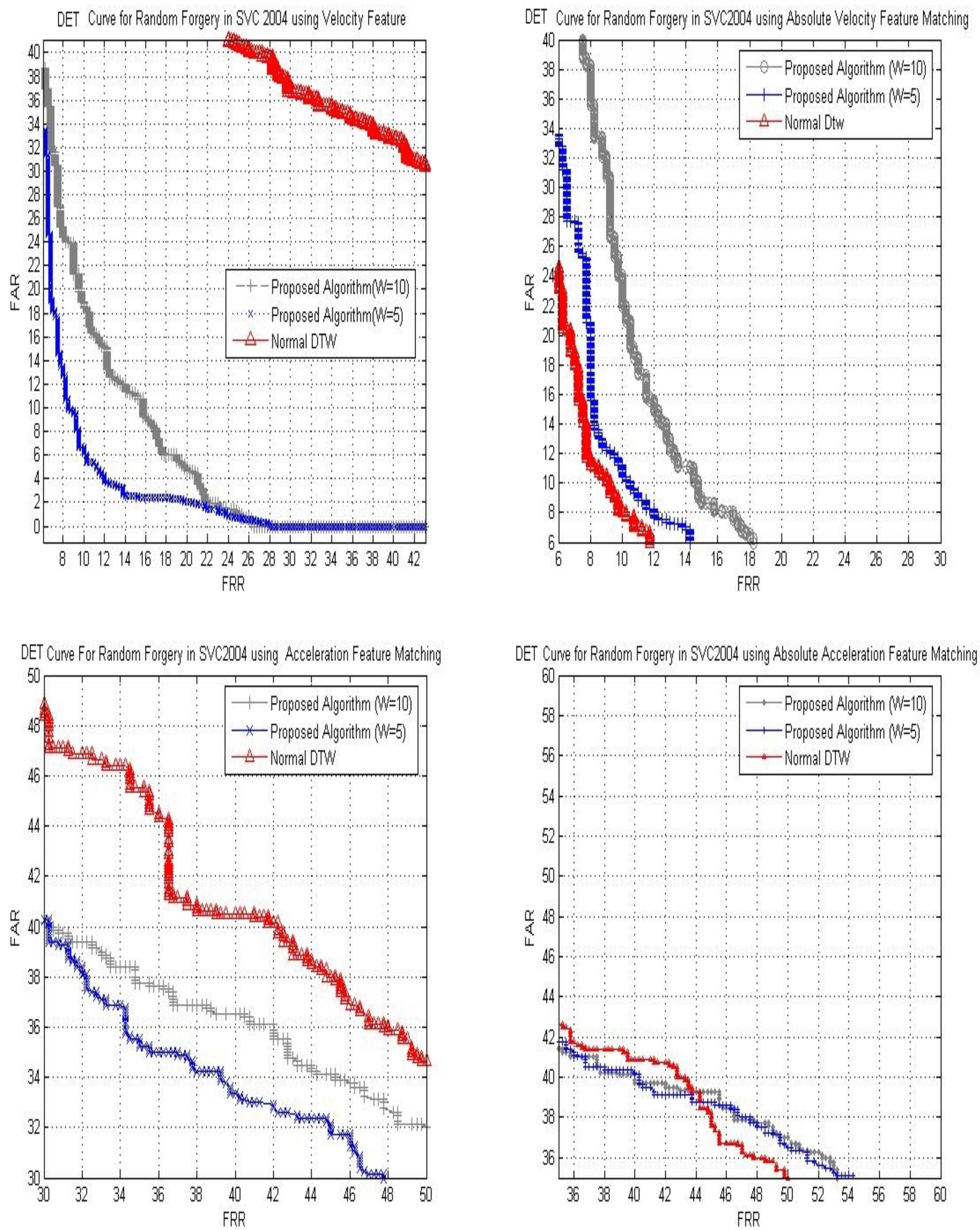


Figure 4.6: DET Curves for Random Forgery in SVC20004 Database with individual Features

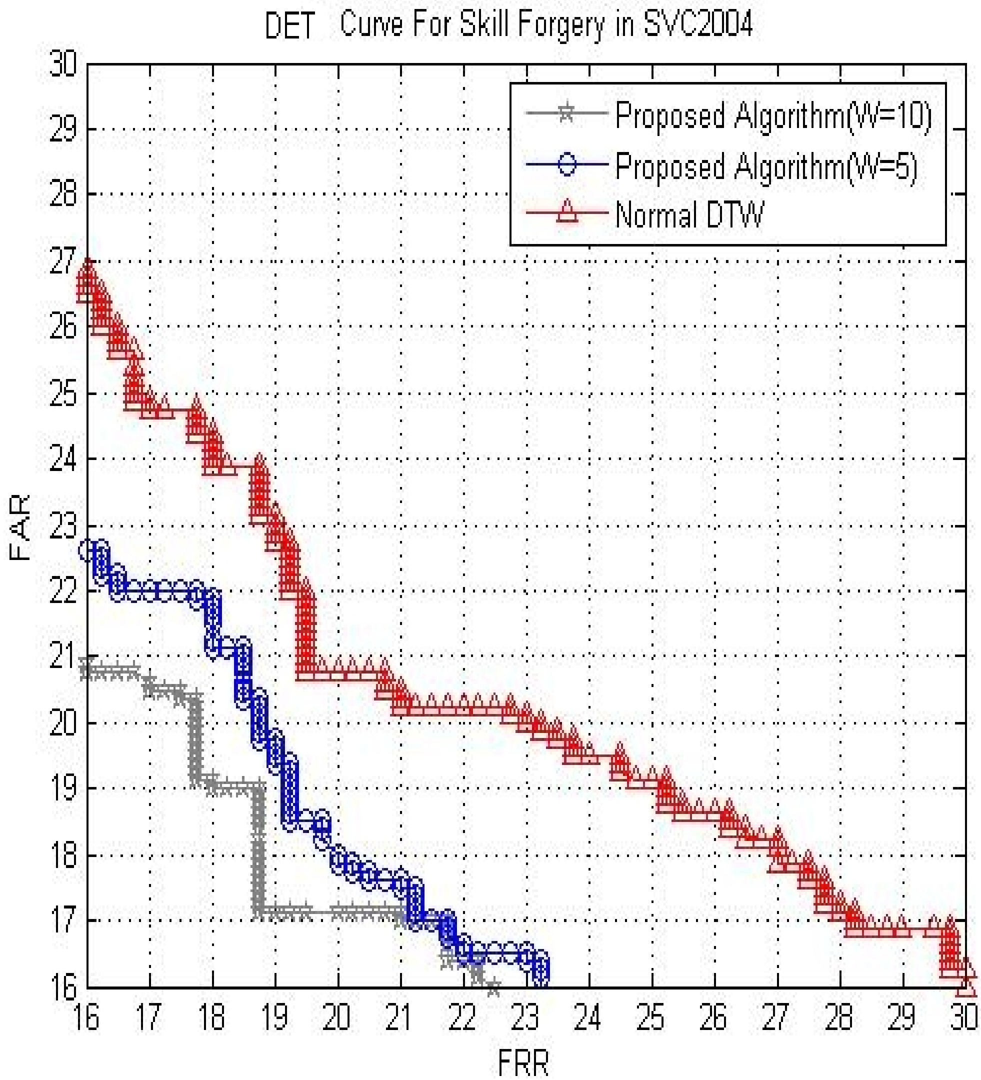


Figure 4.7: DET Curves for Skill Forgery in SVC20004 Database using Combined Features

Chapter 5

Conclusions And Future Work

5.1 Conclusions

The proposed signature verification has performed well in case of random forgeries, however in case of skill forgeries result is found to be satisfactory. The EER obtained in skilled forgery is close to EER obtained by state of art DTW based system such as [26]. Although some of the DTW base system, which have proposed stroke (segment) based DTW system [18, 28] have claimed to achieved an EER of 6%. The performance of the developed learning algorithm which tries to find the best global constraints along the series length showed that the more we decrease the parameter "W"(the constraint is more closer to diagonal the better it differentiated the series from other class data). There is a threshold after which decreasing the "W" will give poor performance, which is visible from the DET Curve of Pressure feature as shown in

Figure 4.5.

Their is a significant speed up in the evaluation of DTW as only the points within the global constraints are evaluated for global and local distance matrix calculation rather than the whole matrix.

5.2 Future Improvement

The proposed system is evaluated based on global threshold. Using writer dependent threshold, it can further improved in performance. Rather than using simple mean-standard deviation based verification, implementing SVM may classify the test signatures better and increase the performance.

Skill forgery performance can be improved by applying this algorithm per stroke basis or segment wise as mentioned in previous section.

Although we have implemented 8 major features, their are several other dynamic features that can also be incorporated in this algorithm and tested their performance.

Bibliography

[1] H. Baltzakis and N. Papamarkos. A new signature verification technique based on a two-stage neural network classifier. *Engineering Applications of Artificial Intelligence*, 14(1):95 – 103, 2001.

[2] D. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *AAAI-94 workshop on knowledge discovery in databases*, pages 229–248, 1994.

[3] Principal Components and Factor Analysis. Internet: "http://www.statsoft.com/textbook/stfacan.html" (accessed 2009-05-24).

[4] The First International Signature Verification Competition (SVC 2004) Dit-Yan Yeung(chairmain). Internet: http://www.cse.ust.hk/svc2004/index.html (accessed 2009-05-17).

[5] JGA Dolfing, EHL Aarts, and J. Van Oosterhout. On-line signature verifica-

tion with Hidden Markov Models. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 2, 1998.

[6] M.M. Fard, M.M. Fard, and N. Mozayani. A new on-line signature verification by Spatio-Temporal neural network. In *IEEE International Conference on Intelligence and Security Informatics, 2008. ISI 2008*, pages 233–235, 2008.

[7] Marcos Faundez-Zanuy. On-line signature recognition based on VQ-DTW. *Pattern Recognition*, 40(3):981 – 992, 2007.

[8] Julian Fierrez, Javier Ortega-Garcia, Daniel Ramos, and Joaquin Gonzalez-Rodriguez. HMM-based on-line signature verification: Feature extraction and signature modeling. *Pattern Recognition Letters*, 28(16):2325 – 2334, 2007.

[9] M. Fuentes, S. Garcia-Salicetti, and B. Dorizzi. On line signature verification: Fusion of a Hidden Markov Model and a neural network via a support vector machine. In *Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition*, pages 253–258, 2002.

[10] C. Gruber, T. Gruber, and B. Sick. Online signature verification with new time series kernels for support vector machines. *Lecture notes in computer science*, 3832:500, 2006.

[11] D. Heckerman et al. A tutorial on learning with Bayesian networks. *NATO ASI SERIES D BEHAVIOURAL AND SOCIAL SCIENCES*, 89:301–354, 1998.

[12] F. Itakura. Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 23(1):67–72, 1975.

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

[14] AK Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[15] A. Kholmatov and B. Yanikoglu. Identity authentication using improved online signature verification method. *Pattern Recognition Letters*, 26(15):2400–2408, 2005.

[16] H. Lv and W. Wang. On-Line Signature Verification Based on Dynamic Bayesian Network. *Lecture notes in computer science*, 4221:507, 2006.

[17] R. Martens, L. Claesen, and L. IMEC. On-line signature verification by dynamic time-warping. In *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, volume 3, 1996.

[18] Loris Nanni and Alessandra Lumini. A novel local on-line signature verification system. *Pattern Recognition Letters*, 29(5):559 – 568, 2008.

[19] V. Niennattrakul, C.A. Ratanamahatana, Z. Chen, D. Wen, C. Bessiere, E. Hebrard, B. Hnich, Z. Kiziltan, T. Walsh, M. Nekovee, et al. Learning DTW Global Constraint for Time Series Classification. *Arxiv preprint arXiv:0903.0041*, 2009.

[20] The oldest document. Internet: http://www.besner.org/1-04-The.oldest.doc.html (accessed 2009-05-24).

[21] C.A. Ratanamahatana and E. Keogh. Making time-series classification more accurate using learned constraints. In *Proceedings of SIAM International Conference on Data Mining*, pages 11–22. Lake Buena Vista, Florida, 2004.

[22] G. Rigoll and A. Kosmala. A systematic comparison between on-line and off-line methods forsignature verification with hidden Markov models. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 2, 1998.

[23] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *Readings in speech recognition*, page 159, 1990.

[24] VN Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[25] C. Vielhauer, R. Steinmetz, and A. Mayerhofer. Biometric hash based on statistical features of online signatures. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 1, pages 123–126 vol.1, 2002.

[26] M. Wirotius, JY Ramel, and N. Vincent. Improving DTW for online handwritten signature verification. *Lecture Notes in Computer Science*, pages 786–793, 2004.

[27] Xuhong Xiao and Graham Leedham. Signature verification using a modified Bayesian network. *Pattern Recognition*, 35(5):983 – 995, 2002.

[28] J. Zhang and S. KAMATA. Online Signature Verification Using Segment-to-Segment Matching.