

**AUTOMATED STUDENT'S ATTENDANCE ENTERING
SYSTEM BY ELIMINATING FORGE SIGNATURES**

K. P. M. L. P. Weerasinghe

149235H

Faculty of Information Technology

University of Moratuwa

June 2017

AUTOMATED STUDENT'S ATTENDANCE ENTERING SYSTEM BY ELIMINATING FORGE SIGNATURES

K. P. M. L. P. Weerasinghe

149235H

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the partial fulfillment of the requirements of the Degree Master of Science in Information Technology.

June 2017

Declaration

We declare that this thesis is our own work and has not been submitted in any form for another degree or diploma at any university or other institution of tertiary education. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

K. P. M. L. P. Weerasinghe

Date:

Supervised by

Mr. B. H. Sudantha

Date:

Acknowledgement

First and foremost, I would like to express my deepest gratitude to my supervisor, Mr. B. H. Sudantha, for his dedicated guidance, invaluable advice, and constant encouragement throughout my master study. I am also indebted to him for the efforts he has devoted to serious consultations and serious review of this thesis. His enthusiasm and insights in many research problems have provided me with a source of thoughts and actions.

I am thankful to lecturer, the Head of the Department of Information Technology, Dr. L. Ranathunga for his help to understanding Image Processing Techniques during the course module IN 5610.

This thesis would not have been completed without the constant support of my husband Mr. Lakshan Aponsu. I would express my wordless thanks to my husband for his deep understanding and encouragement during these years.

Abstract

Entering student's attendance into the excel sheets for each of the subjects, is very difficult, time consuming process. At the beginning of some course modules, the number of registered students are unknown. Lecturers use papers to take students attendance, so that the entering of student's attendance is more complex. Automated student's attendance entering system can be used to simplify the task. To build up such a system signature recognition and verification is important.

The signature of a person is an important biometric attribute of a human being which can be used to authenticate human identity. To automate the process, this thesis consists of 3 phases. Signature identification and extraction from the attendance sheets and classification for testing process, Signature recognition by comparing each signature in the database and recognize the owner of the signature and the last phase is signature verification to identify whether the signature is original or counterfeit. In each phase, necessary image processing techniques are applied and useful features are extracted from each signature. Support Vector Machine (SVM) is used for classification of signatures extracted from attendance sheets. For signature recognition, multiclass Support Vector Machine is used and analyze using Fault Acceptance Ratio (FAR) and Fault Rejection Ratio (FRR) to check the accuracy of the classifier. Signature database consists only genuine signatures of each signer so that in signature verification stage a machine learning technique, Kolmogorov Smirnov test is used to verify the signature is belong to the original and if it is not match with the particular student's signature, taken as zero. In this paper, off-line signature recognition & verification is proposed, where the signature is captured and presented to the user in an image format.

A software package, Matlab2016b is used for this procedure. The described method in this thesis represents an effective and accurate approach to automatic signature recognition and verification. It is capable of matching the test signatures with the database of 83.33% accuracy. It is capable of classifying all signatures in the attendance sheet of 100% accuracy. In this work, it verifies 100% of signatures is original.

Eventually, based on the methodologies employed in this thesis, it provides a promising stage for the development of an automated online signature detection system.

Contents

Acknowledgements	i
Abstract	ii
List of Tables	iii
List of Figures	iv

Chapter 1 – Introduction

1.1 Prolegomena.....	1
1.2 Background and Motivation for Automate the Signature Extraction, Recognition and Verification Process.....	2
1.3 Problem in Brief.....	3
1.4 Research Objectives.....	3
1.5 Outline of this Thesis.....	4
1.6 Summary.....	4

Chapter 2 – Review of Literature

2.1 An Overview of Document Signature Extraction.....	6
2.2 An Overview of Signature Recognition & Verification.....	6
2.2.1 Neural Network Approach.....	7
2.2.2 Support Vector Machine Approach.....	8
2.2.3 Hidden Markov Model Approach.....	8
2.2.4 Neural Fuzzy Based Approach.....	9
2.3 Another Approaches for Signature Verification.....	10

2.3.1 Image based approach.....	10
2.3.2 Statistical approach.....	10
2.3.3 Template matching approach.....	11
2.4 Future Challenges of Signature Recognition and Verification.....	11
2.5 Summary.....	12

Chapter 3 – Signature Based Systems

3.1 Types of Signatures.....	13
3.2 Types of Signature Forgeries.....	13
3.3 Background of Signature Extraction from Scanned Documents.....	13
3.4 Background of Signature Recognition and Verification.....	14
3.5 Types of Signature Recognition and Verification Systems.....	14
3.5.1 Off-Line or Static Signature Recognition and Verification.....	14
3.5.2 On-Line or Dynamic Signature Recognition and Verification.....	16

Chapter 4 – Proposed Approach

4.1 Signature Classification using Support Vector Machine.....	17
4.2 Signature Recognition using Multiclass Support Vector Machine.....	20
4.2.1 Signature Recognition Workflow.....	20
4.2.2 Error Correcting Output Code Multiclass Model.....	21
4.3 Signature Verification using Kolmogorov Smirnov Test.....	22
4.3.1 Person Dependent Learning (Person Specific learning).....	23
4.3.2 Within-person Distribution.....	23
4.3.3 Person Dependent Classification.....	24
4.3.4 Comparing Distributions.....	25
4.3.5 Kolmogorov-Smirnov Test.....	25

4.3.6 Person-Dependent Method.....	26
------------------------------------	----

Chapter 5 – Experimental Setup

5.1 Signature Identification and Extraction from Scanned Attendance Sheets.....	27
5.1.1 Convert RGB to Grayscale.....	27
5.1.2 Binarization.....	28
5.1.3 Edge Detection.....	29
5.1.4 Remove Unnecessary Pixels.....	30
5.1.5 Morphological Dilation.....	31
5.1.5.1 Thicken.....	32
5.1.5.2 Bridge.....	33
5.1.6 Image Segmentation and Cropping.....	34
5.2 Signature Classification using Binary SVM.....	35
5.3 Signature Recognition.....	35
5.3.1 Signature Acquisition.....	35
5.3.2 Image Preprocessing.....	36
5.3.2.1 Convert RGB to Grayscale.....	37
5.3.2.2 Binarization.....	37
5.3.2.3 Remove Unnecessary Pixels.....	37
5.3.2.4 Thinning.....	38
5.3.2.5 Auto Cropping of Signature.....	38
5.3.3 Feature Extraction.....	39
5.3.3.1 Global Features.....	40
5.3.3.2 Local Features.....	41
5.4 Signature Verification.....	42
5.4.1 Algorithm for KS Test.....	42

5.4.2 Writing Data to Excel Sheets.....	45
Chapter 6 – Results and Discussion	
6.1 Signature Extraction Process.....	46
6.1.1 Results Analysis in Signature Classification using Binary SVM.....	47
6.2 Signature Recognition and Verification Process.....	47
6.2.1 Kolmogorov Smirnov Test Performance Measure.....	48
Chapter 7 – Conclusion.....	50
References.....	52
Appendix.....	56

List of Tables

Table 5.1: Sample distance distribution of known signature.....	44
Table 5.2: Sample distance distribution of known vs questioned.....	44

List of Figures

Figure 1.1: An illustration of the differences between the three areas of application for signature matching	2
Figure 4.1: A brief outline of proposed framework for signature extraction.....	17
Figure 4.2: Support vectors which are closest to hyperplane	18
Figure 4.3: Signature recognition workflow	20
Figure 4.4: Signature verification where a questioned signature(right) is matched against five knowns.....	22
Figure 4.5: Comparing all possible genuine-genuine pairs.....	24
Figure 4.6: The steps of KS test for signature verification.....	26
Figure 5.1: Grayscale image of scanned attendance sheet.....	28
Figure 5.2: Binarize image of scanned attendance sheet.....	29
Figure 5.3: Detected edges of scanned attendance sheet.....	30
Figure 5.4: Filtered image of scanned attendance sheet.....	31
Figure 5.5: Scanned attendance sheet by applying close operator.....	32
Figure 5.6: Scanned attendance sheet by applying thicken operator.....	33
Figure 5.7: Bridged image of scanned attendance sheet.....	34
Figure 5.8: Scanned attendance sheet marked patches with a box.....	35
Figure 5.9: Sample of signatures.....	36
Figure 5.10: Variation of a signature of the same person.....	36
Figure 5.11: Grayscale signature image.....	37
Figure 5.12: Binarized signature image.....	37
Figure 5.13: Signature image after removing unwanted pixels.....	38
Figure 5.14: Signature image after thinning.....	38

Figure 5.15: Signature image with area cropped.....	39
Figure 5.16: Partitioned signature into 4 parts.....	40
Figure 5.17: HOG features extracted from one signature.....	42
Figure 5.18: Signature recognized as genuine by verification process.....	45
Figure 5.19: Enter attendance records into excel sheet.....	45
Figure 6.1: Signature image with multiple bounded regions due to discontinuity.....	46
Figure 6.2: Signature image after removing the discontinuity.....	46
Figure 6.3: Misclassification of a signature.....	48

Introduction

1.1 Prolegomena

Handwritten signatures are considered as the most natural method of authenticating a person's identity. However human signatures can be handled as an image and recognized using computer vision and machine learning techniques. With modern computers, there is need to develop fast algorithms for signature recognition. There are various approaches to signature recognition with a lot of scope of research. Signatures are composed of special characters and flourishes and therefore most of the time they can be unreadable. Also, intrapersonal variations and interpersonal differences make it necessary to analyze them as complete images and not as letters and words put together [15]. Signature recognition is the process of verifying the writer's identity by checking the signature against samples kept in the database. The result of this process is usually between 0 and 1 which represents a fit ratio (1 for match and 0 for mismatch). As signature is the primary mechanism both for authentication and authorization in legal transactions, the need for efficient automated solutions for signature verification has increased. Unlike a password, PIN numbers etc, the captured values of the handwritten signature are unique to an individual and virtually impossible to duplicate. The inevitable side-effect of signatures is that they can be exploited for the purpose of feigning a document's authenticity. Hence the need for research in efficient automated solutions for signature recognition and verification has increased in recent years to avoid being vulnerable to fraud [12].

Signature matching is used in areas such as extraction [6], recognition [24] and verification [10]. While signature extraction aims to find document images that contain signatures [6] and signature recognition tries to find the corresponding signer of a test sample given a database of signature exemplars from different signers [4], signature verification deals with confirming the authenticity of a signature i.e. decides whether a sample signature is genuine or forgery by comparing it with stored reference signatures. The differences between the three categories are illustrated in Figure 1.1. It shows the respective problems that have to be solved for signature extraction (left), identification (middle) and verification (right).

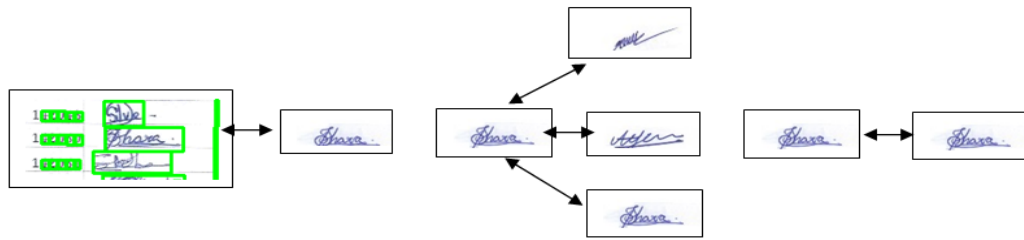


Figure 1.1: Differences Between Three Areas of Application for Signature Matching.

1.2 Background and Motivation for Automate the Signature Extraction, Recognition and Verification Process

Attendance records are very important in the academic activities of universities. Almost all the universities in Sri Lanka, signatures of candidates are taken in lectures, practical sessions, during examinations etc. to verify the presence of the real candidate. Paper based attendance sheet is passed in each session to put the signature of each student. Entering student's attendance into the excel sheets for each of the subjects which is a very difficult, time consuming process. At the beginning of some course modules, the number of registered students is unknown. Even though lecturers use papers to take students attendance, maintaining the attendance database of hundreds of students has become a tedious task. Automated student's attendance entering system can be used to simplify this task. Many attempts were made to automate this process with success to a certain extent. Many of these systems make use of sophisticated biometric equipment while some others use Barcodes and Radio Frequency Identity Cards [8]. But even today the majorly used system is to take the signature of present candidates and then manually enter these records in to the computer. In this study, the process will be automated by developing a system which uses image processing to automatically update the attendance records in the computer. To build up such a system signature extraction, recognition and verification is important.

The handwritten signature is a very common way of authenticity. Despite its known weaknesses (relatively easy to copy, signatures of one person may vary significantly) and development of cryptographic and biometric techniques, it is still the most commonly used way of authentication when dealing with paper documents and forms. In this thesis, we focus on the application of biometric recognition for automatic

student authentication, in particular making use of handwritten signatures, which are one of the most socially accepted biometric traits. In education, signatures are used for attendance control, either to lectures or exams, but not for (automatic) authentication. With the rapid deployment of dynamic signature recognition, this technology is ready to be used for student authentication. Also, the use of this technology can be extended to different administrative services within the education system, in order to add a higher security level to the traditional procedures of authentication (e.g., visually checking the face and/or signature on the person identity card).

Another important task is identification of forge signatures. If we count the number of signatures and the number of heads during a lecture, practical session or in examinations, they should be same. But sometimes some students sign for their friends or replaced by other students. Therefore, identification of forge signatures is very much important in this type of situations. From the viewpoint of automating the attendance entering system it can be viewed as one that involves machine learning from a population of signatures. In this study person, dependent learning will be used so that there are only genuine signatures in the database. This is called special learning. In special learning, a person's signature is learnt from multiple samples of only that person's signature, where within person similarities are learnt to identify the signature is genuine or counterfeit.

1.3 Problem in Brief

Entering student's attendance into the excel sheets for each of the subjects which is a very difficult, time consuming process. Identification of forge signatures is an important in academia.

1.4 Research Objectives

- Apply appropriate image preprocessing techniques.
- Feature extraction using suitable methods.
- Extract signatures from scanned attendance sheets.
- Recognize the signer of the signature from the database.

- Identify the students' forge signatures.
- Automate the process.

1.5 Outline of this Thesis

The thesis is organized as follows. Chapters 2 critically review the research in signature extraction, recognition & verification and define the research problem together with identification technology to solve the problem. Chapter 3 gives a detailed description about signature based systems to get an idea about what are the types of signatures, what are the types of forge signatures, about feature extraction techniques to extract features from images and also about the signature extraction, recognition and verification. Chapter 4 is on the approaches used in this study to signature extraction, signature recognition & verification solution. Chapter 5 gives the experimental setup according to the approaches discussed in chapter 4. Chapter 6 analyzes the results which are given in signature extraction, recognition and verification processes. Chapter 7 concludes the research findings with a note on further research.

1.6 Summary

This chapter presented a description of the overall project described in this thesis. Next chapter will discuss a critical review of the literature in relation to developments and challenges in Signature recognition & verification.

Review of Literature

2.1 An Overview of Document Signature Extraction

First step in the automated attendance entering system is to separate the signatures from its background. Many researches have been done to extract signatures from printed documents.

Ogul and coworkers [6] described a discriminative framework to extract signature from a bank service application document. The framework is based on the classification of segmented image regions using a set of representative features. The segmentation is done using a two-phase connected component labeling approach. Then evaluate solely and combined effects of several feature representation schemes in distinguishing signature and non-signature segments over a Support Vector Machine classifier.

Gupta [9] has done a cursive signature extraction and verification. In his research, he presented a new approach for document image decomposition and verification based on connected component analysis and geometric properties of labeled regions. He also extracted a set of efficient, invariant and compact features for verification purposes using spatial partitioning of the signature region.

Ritesh Banka [1] has presented a new approach for extraction of signature and handwritten regions from official binary document images. He presented a new two-level scale invariant classification technique to extract the gray scale handwritten area from scanned document. In the First level, the printed characters possessing self-symmetry (vertical, horizontal and diagonal) are extracted from the document, based on which the threshold values are estimated. Some printed characters are misclassified as handwritten in this level. To reclassify them, a second level classifier is designed which uses the presence of a symmetric hole as the feature. According to the results the overall accuracy was 95.6%.

Mohan Gautam [7] proposed a new method for Extracting signature from image document based on the auto cropping method. Method improved the performance of

security system based on signature images as well as provided the region of interest of the used image for the biometric system. The method also reduced the time cost associated with signature detection. Using adaptive thresholding they segment the scanned documents into foreground and background. They set all pixels whose intensity values are above a threshold to a foreground value and all the remaining pixels to a background value over the signature document image. To remove the discontinuity between the pixels of signature images they were used morphology. Morphological method used bridge to connect the pixels and remove operator to remove the interior pixel region. The remaining pixel made the signature image skeleton. That was used to select the signature Region of Interest (ROI) using auto cropping method. In that auto cropping method, they used Image Station Automatic Elevations (ISAE) technique to select the connected pixel which sizes are greater than 250 pixels. The cropped signature has no garbage region it cropped only the ROI of signature image.

Manesh [23] proposed a method to automatically identify the signature in the scanned document images. A simple region growing algorithm was used to segment the document into a number of patches. Then the state features of all the patches were extracted to identify the signature in the document images. A label for each such segmented patch was inferred using neural network model (NN) and support vector machine (SVM).

2.2 An Overview of Signature Recognition & Verification

The verification process has to wholly rely on the features that can be extracted from the trace of the static signature images only. Vigorous research has been pursued in handwriting analysis and pattern matching for a number of years. In the area of Handwritten Signature Verification (HSV), especially offline HSV, different technologies have been used and still the area is being explored. In this section, we review some of the recent papers on offline HSV. The approaches used by different researchers differ in the type of features extracted, the training method, and the classification and verification model used.

2.2.1 Neural Network Approach

A research on offline signature recognition using neural network approach has been done by Ali Karouni and coworkers [12]. This research is based on offline verification of signatures using set of simple shape based geometric features such as area, center of gravity, eccentricity, kurtosis and skewness. The authors have implemented the solution to classify the signatures: exact or forged using ANN. The precision of the signature verification system was expressed using false acceptance ratio and false rejection ratio.

Offline handwritten signature verification using ANN [15][25] was another concern on this research paper. Sisodia [25] implemented a Static Signature Verification System with four stages such as: image pre- processing, feature extraction, classification and decision making. Classifier used an ANN with Error Back Propagation algorithm to attain a certain result. The relevant features used by the classification are centroid, length and width of the signature in the 200×100 pixels' image box, quadrant areas, one dimensional first and second derivatives of the image and global slant angle. Menu Bhatia [15] was used maximum horizontal and vertical histogram, center of mass, normalized area of signature, aspect ratio, tri surface feature, six-fold surface feature and transition feature as the extracted features from the candidate signature.

Woods [8] considered image area, vertical center of the signature, horizontal center of the signature, maximum vertical projection, maximum horizontal projection, vertical projection peaks, horizontal projection peaks, number of edge points, number of cross points and Hough transform for feature extraction of each signatures. Extracted values of each signature images from the database of 150 images are given to the feed forward neural network (trained using back propagation gradient descent learning).

Gulzar and coworkers [13] present neural network based recognition of offline handwritten signature system that is trained with low- resolution scanned signature images. And also Prashanth C.R. [21] presents DWT based offline signature verification using angular features (DOSVAF). The signature is resized and Discrete Wavelet Transform (DWT) is applied on the blocks to extract the features.

2.2.2 Support Vector Machine Approach

Support Vector Machines (SVMs) are machine learning algorithms that use a high dimensional feature space and estimate differences between classes of given data to generalize unseen data. The system in [19] uses global, directional and grid features of the signature and SVM for classification and verification. The database of 1320 signatures is used from 70 writers. 40 writers are used for training with each signing 8 signatures thus a total of 320 signatures for training. For initial testing, the approach uses 8 original signatures and 8 forgeries and achieves FRR 2% and FAR 11%.

Ramachandra and colleagues [29] have proposed SVGMC algorithm in which use two concepts Graph Matching and Cross validation for signature verification. Signatures are compared by bipartite graph from which a minimum cost complete matching is obtained and the Euclidean distance is determined. Cross validation was used to solve the problem of selection of reference signatures, which derives the best reference set of signatures for the system producing optimal decision threshold value.

Vahid Kiani [14] proposes a new method for signature verification using local Radon Transform. The proposed method uses Radon Transform locally as feature extractor and Support Vector Machine (SVM) as classifier. The main idea of their method is using Radon Transform locally for line segments detection and feature extraction, against using it globally. The advantages of the proposed method are robustness to noise, size invariance and shift invariance. Having used a dataset of 600 signatures from 20 Persian writers, and another dataset of 924 signatures from 22 English writers, their system achieved good results.

On the same token, many researches have been carried out for signature recognition and identification using ANN. Among others, Piyush Shanker [21] used Dynamic Time Wrapping (DTW), Radhika [19] proposes signature Authentication Based on Moment Invariants Using Support Vector Machine, while Kalera [11] uses distance statistics.

2.2.3 Hidden Markov Model Approach

Hidden Markov Model (HMM) is one of the most widely used models for sequence analysis in signature verification. Handwritten signature is a sequence of vectors of values related to each point of signature in its trajectory. Therefore, a well-chosen set

of feature vectors for HMM could lead to the design of an efficient signature verification system. These Models are stochastic models which have the capacity to absorb the variability between patterns and their similarities. In HMM stochastic matching (model and the signature) is involved. This matching is done by steps of probability distribution of features involved in the signatures or the probability of how the original signature is calculated. If the results show a higher probability than the test signatures probability, then the signatures is by the original person, otherwise the signatures are rejected.

In paper [11] a system is introduced that uses only global features. A discrete random transform which is a sinograph is calculated for each binary signature image at range of 0 – 360, which is a function of total pixel in the image and the intensity per given pixel calculated using non-overlapping beams per angle for X number of angles. Due to this periodicity, it is shift, rotation and scale invariant. A HMM is used to model each writer signature. The method achieves an AER of 18.4% for a set of 440 genuine signatures from 32 writers with 132 skilled forgeries.

In contrast to the previous research, some have also used HMM and Graphometric features [11][12] and conjunction with neural network and support vector machines [18]. Abdullah [5] proposes a new method for signature recognition using Delaunay triangulation. George B. [3] has done finger print identification using Delaunay triangulation.

2.2.4 Neural Fuzzy Based Approach

Rupali Mehra and coworkers [17] present Surf features and neural-fuzzy techniques based recognition of offline signatures system that is trained with low-resolution scanned signature images. Therefore, in their paper; off-line signature recognition & verification using neural-fuzzy is proposed, where the signature is captured and presented to the user in an image format. And signatures are verified based on parameters extracted from the signature using various image processing techniques. Then Off-line Signature Recognition and Verification is implemented with SURF features and Neural Fuzzy techniques in ANFIS in Matlab.

2.3 Another Approaches for Signature Verification

Robert and coworkers [10] carried out a literature review about offline handwritten signature verification. Challenges, the data can be used in the process, different type of data preprocessing, feature extraction and model training algorithms are discussed in their paper.

2.3.1 Image Based Approach

Gautam [22] has used SIFT and Delaunay triangulation for image matching in their research. Mandle [16] has used SIFT/ SURF algorithm for offline signature recognition. They proposed a SIFT-SURF algorithm which is used for enhanced offline signature recognition. The SIFT-SURF algorithm computes integral image, obtains hessian data and interest point for each computed integral image, applied neuro-scaling PCA based radial basis function neural network to compute the optimal features for each signature image to come up with an algorithm that is invariant to scaling and rotation as well as reliably match transposition among genuine samples of a signature image.

2.3.2 Statistical Approach

Using statistical knowledge, the relation, deviation, etc. between two or more data items can easily be found out. To find out the relation between some set of data items we generally follow the concept of Correlation Coefficients. In general, statistical usage refers to the departure of two variables from independence. To verify an entered signature with the help of an average signature, which is obtained from the set of, previously collected signatures, this approach follows the concept of correlation to find out the amount of divergence in between them. A unique method is introduced in [29]. In this approach, various features are extracted which include global features like image gradient, statistical features derived from distribution of pixels of a signature and geometric and topographical descriptors like local correspondence to trace of the signature. The classification involves obtaining variations between the signatures of the same writer and obtaining a distribution in distance space. For any questioned signature the method obtains a distribution which is compared with the available known and a probability of similarity is obtained using a statistical Kolmogorov-Smirnov test. Using only 4 genuine samples for learning, the method

achieves 84% accuracy which can be improved to 89% when the genuine signature sample size is increased. This method does not use the set of forgery signatures in the training.

2.3.3 Template Matching Approach

Enturk [8] proposed two methods for the detection of skilled forgeries using template matching. One method is based on the optimal matching of the one-dimensional projection profiles of the signature patterns and the other is based on the elastic matching of the strokes in the two-dimensional signature patterns. Given a test signature to be verified, the positional variations are compared with the statistics of the training set and a decision based on a distance measure is made. Both binary and grey-level signature images are tested. The average verification error rate of 18.1% was achieved when the local peaks of the vertical projection profiles of grey-level signature images were used for matching and with the full estimated covariance matrix incorporated.

Neha and coworkers [31] have done a research on offline handwritten signature verification using template matching and clustering technique. After the signature acquisition, pre-processing and feature extraction to verify the test signature the positional variations are compared with the statistics of the training set and a decision based on a distance measure is made. Two template matching approaches which are feature based and template based approaches are used in their research.

2.4 Future Challenges of Signature Recognition and Verification

In this work, there is a challenge of creating a system with the ability to recognize handwritten signature and verify its authenticity. This poses a problem because we are trying to get the computer to solve a problem with a method of solution that goes outside the convention of writing an algorithmic process. Offline signature recognition is more difficult than online as dynamic information is not available and it is difficult to recover them from the offline images.

And also, researchers come across two problems in offline signature verification,

- (i) Most of the dynamic information in the signature is lost and

- (ii) Low quantity of available signature samples versus high number of extracted features.

The first issue is addressed by some researchers [2][28] but this is still a challenging problem.

The major problem associated with signature verification is the availability of limited data. As signature data are legally accepted as the authentication means for many financial or other official works, this is difficult to have a sufficient amount of data required to develop a signature verification system. As a result, robust parameter estimation on limited sample sets is still one of the major research issues in this field. One technique to address this problem is to extend the techniques of classical model adaptation for discriminative training.

The other challenging problem in offline signature verification is the feature extraction process. Choice of features depends on the style of the signatures and hence different styled-signatures will have different characteristic features. So, it is difficult to develop one general system to classify every style of signatures. Signatures in different scripts may not be recognized by a single classifier or even a classification system. It has been observed that most of the researchers have proposed or developed their systems for a limited type of signatures. However, achieving an acceptable accuracy in various individual signature styles will make it possible to work out a general signature verification system. Future work should focus on adapting the classification function dynamically to the signature for authentication, and thus combining the advantages of different approaches.

2.5 Summary

This chapter presented a comprehensive literature review on the signature extraction, recognition and verification research and identified the research problem as the inadequate attention to reliability of algorithms. Next chapter will discuss the signature based systems further to understand about this thesis approach.

Signature Based Systems

3.1 Types of Signatures

Handwritten signatures are of different shapes and sizes and the variations in them are so immense that it is difficult for a human being to distinguish a genuine signature from a forged one by having a glance at the signature. There are different types of signatures used in real life. Broadly, signatures can be classified as,

1. Simple Signatures: These are the ones where the person just writes his or her name
2. Cursive signatures: These are the ones that are written in a cursive way
3. Graphical signatures: The signatures can be classified as graphical when cursive signatures depict geometric patterns.

3.2 Types of Signature Forgeries

The main task of any signature verification system is to detect whether the signature is genuine or counterfeit. Forgery is a crime that aims at deceiving people. Since actual forgeries are difficult to obtain, the instrument and the results of the verification depend on the type of the forgery. Basically, there are three types that have been defined:

Random forgery: Random forgery is done by a person who doesn't know the shape and structure of the original signature.

Simple forgery: In this type of forgery the person concerned has a vague idea of the actual signature, but is signing without much practice.

Skilled forgery: Written by a person who knows the shape with much practice of the signature.

3.3 Background of Signature Extraction from Scanned Documents

In spite of a drastic increase in the use of electronic data in many applications, a signature in a printed document is still considered to be most reliable way of user

commitment, approval and verification. In most applications (bank, universities, institutes etc.) manually inspect the signature. Manual inspection is very suspect to errors and misleading interpretations. Furthermore, it requires an additional workload, which causes either an increase in the cost of human resources. Therefore, it has become essential to use intelligent software techniques for automated analysis of documents to perform signature related tasks.

Analysis of a document image involves following tasks:

- (i) Conversion of an editable text for reusability,
- (ii) Extraction of important information,
- (iii) Separation of the text and non-text elements of the image and subsequent analysis,
- (iv) Analysis of the document image, that enables efficient archiving and retrieval.

Many scanned attendance sheets consist of a multiple set of objects such as text (printed and handwritten), signature, logos and seals. An important task in automated processing of scanned attendance sheets is to find the position of the signature. The task of detecting signatures in scanned documents poses several challenges. First, these types of document images have usually very low resolution, which makes them difficult to enhance. Second, the background of each document is different and usually not known beforehand. Third, documents are subject to restricted processing time due to the urgency of applications. Finally, and maybe the most importantly, the documents often contain auxiliary lines and other handwritten characters that resemble or overlap with signatures.

3.4 Background of Signature Recognition and Verification

Signature recognition and verification involves two separates but strongly related tasks: one of them is identification of the signature owner, and the other is the decision about whether the signature is genuine or forged.

Automated recognition of handwritten signatures became imperative when it was difficult to distinguish genuine signatures from simulated forgeries on the basis of visual assessment. This led to computer recognition of handwritten signatures, which though a bit slow, is more reliable and efficient.

The shape of a person's signature remains similar in all translational, scaled and rotational alignments of the sign. That is the number of crests, troughs and curves remains the same irrespective of the size and orientation of the image. The ratio between consecutive crests and troughs there by remain the same and hence can be used to determine the genuineness of a signature. The success of the proposed approach can be determined from the False Accept Rate (FAR) and False Rejection Rate (FRR).

There are some several unique difficulties in signature recognition and verification: high intra-class variability (an individual's signature may vary greatly day-to-day), large temporal variation (signature may change completely overtime), and high inter-class similarity (forgeries, by nature, attempt to be as indistinguishable from genuine signatures as possible).

3.5 Types of Signature Recognition and Verification Systems

Signature recognition and verification involves two separates but strongly related tasks: one of them is identification of the signature owner, and the other is the decision about whether the signature is genuine or forged. Based on the definitions of signature, it can lead to two different approaches of signature recognition and verification.

3.5.1 Off-Line or Static Signature Recognition and Verification

This approach is based on static characteristics of the signature which are invariant. In this sense signature verification, becomes a typical pattern recognition task knowing that variations in signature pattern are inevitable; the task of signature authentication can be narrowed to drawing the threshold of the range of genuine variation.

Off-line handwriting recognition and verification involves the automatic conversion of text in an image into letter codes which are usable within computer and text-processing applications. The data obtained by this form is regarded as a static representation of handwriting. The technology is successfully used by businesses which process lots of handwritten documents, like insurance companies. The quality of recognition can be substantially increased by structuring the document (by using forms). In Off-line recognition case the signature appears as a 2D (gray level or binary) image. The static signature verification is considered to be much more

difficult because timing and dynamic information are highly degraded in that case. The off-line method uses an optical scanner to obtain the handwriting data written on paper. In this mechanism, the user signs on a piece of paper which is read by a scanner or a camera. The image is then fed to a computer. The computer stores the image as specific to the signer. It is used to identify the user by the image.

3.5.2 On-Line or Dynamic Signature Recognition and Verification

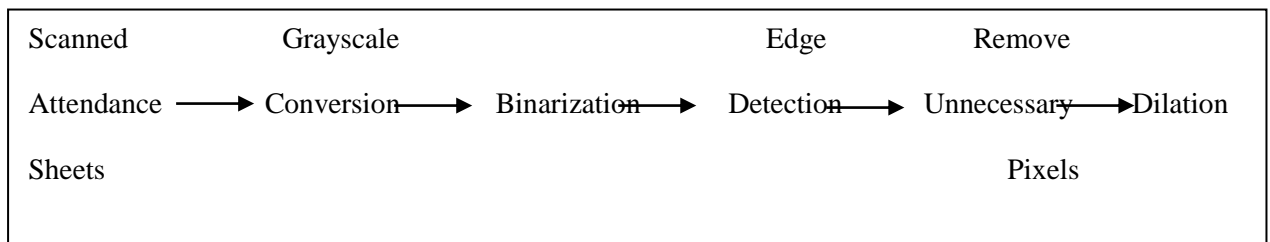
On-line handwriting recognition and verification involves the automatic conversion of text as it is written on a special digitizer, where a sensor picks up the pen-tip movements as well as pen-up or pen-down switching. That kind of data is known as digital ink and can be regarded as a dynamic representation of handwriting. The obtained signal is converted into letter codes which are usable within computer and text-processing applications. This method focusses on Dynamic systems produce signals varying with time (including velocity, acceleration, pressure, position). The signer uses the optical pen and starts writing on the paper. The sensor picks up the image and also the physical characteristics of the handwriting like velocity of movement of hand and acceleration between hand strokes and pressure exerted at the position and records the data along with the dynamic image of the signature. When the signature system is trained then when the signee signs the document the image is picked up dynamically and compared with the data stored for the user. If the data matches, then the user is authenticated otherwise not authenticated. This method is the most viable but expensive. The forgery must be extremely perfect to get around this but the method cannot be bypassed by casual copies of signatures.

Proposed Approach

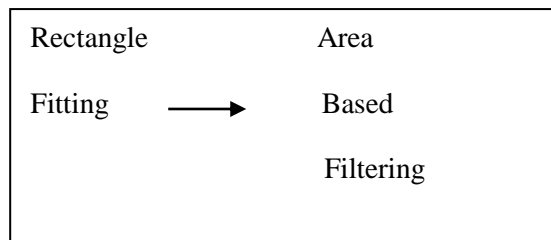
4.1 Signature Classification using Support Vector Machine

The signatures and the non-signature parts extracted from the scanned attendance sheets were classify using binary SVM.

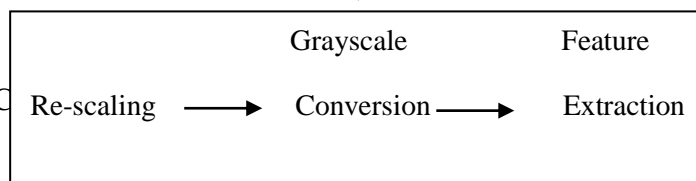
PREPROCESSING



SEGMENTATION



FEATURE EXTRACTION



CLASSIFICATION

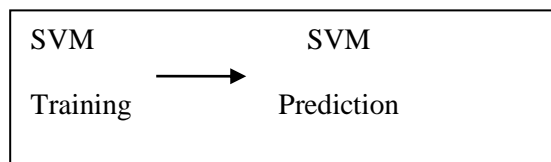


Figure 4.1: A brief outline of proposed framework for signature extraction

Here Support Vector Machine (SVM) has exactly two classes, signature area and non-signature area (with backgrounds and text). SVM classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. The *best* hyperplane for SVM means the one with the largest *margin* between the two classes. Margin means the maximal width of the slab parallel to the hyperplane that has no interior data points.

The *support vectors* are the data points that are closest to the separating hyperplane; these points are on the boundary of the slab. The following figure illustrates these definitions, with + indicating data points of type 1, and - indicating data points of type -1.

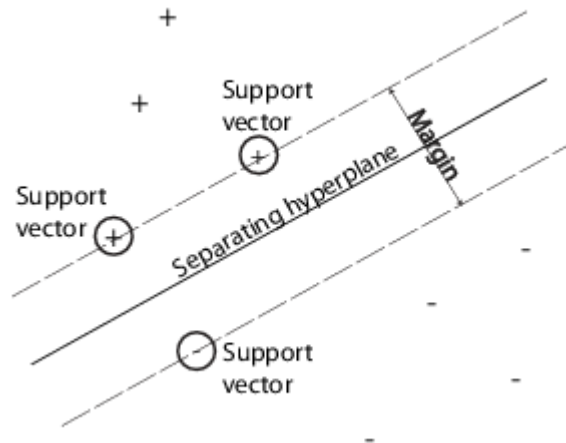


Figure 4.2: Support Vectors which are Closest to Hyperplane

The data for training is a set of points (vectors) x_j along with their categories y_j . For some dimension d , the $x_j \in R^d$, and the $y_j = \pm 1$. The equation of a hyperplane is, $f(x) = x'\beta + b = 0$ where $\beta \in R^d$ and b is a real number.

The following problem defines the *best* separating hyperplane (i.e., the decision boundary). Find β and b that minimize $\|\beta\|$ such that for all data points (x_j, y_j) ,

$$y_j f(x_j) \geq 1$$

The support vectors are the x_j on the boundary, those for which $y_j f(x_j) = 1$.

For mathematical convenience, the problem is usually given as the equivalent problem of minimizing $\|\beta\|$. This is a quadratic programming problem. The optimal solution $(\hat{\beta}, \hat{b})$ enables classification of a vector z as follows:

$$\text{class}(z) = \text{sign}(z'(\hat{\beta}, \hat{b})) = \text{sign}(\hat{f}(z))$$

$\hat{f}(z)$ is the *classification score* and represents the distance z is from the decision boundary.

The `svmtrain` function in Matlab uses an optimization method to identify support vectors s_i , weights α_i , and bias b that are used to classify vectors x according to the following equation:

$$c = \sum_i \alpha_i k(s_i, x) + b,$$

where k is a kernel function. In the case of a linear kernel, k is the dot product. If $c \geq 0$, then x is classified as a member of the first group, otherwise it is classified as a member of the second group.

The `svmclassify` function uses results from `svmtrain` to classify vectors x according to the following equation:

$$c = \sum_i \alpha_i k(s_i, x) + b,$$

where s_i are the support vectors, α_i are the weights, b is the bias, and k is a kernel function. In the case of a linear kernel, k is the dot product. If $c \geq 0$, then x is classified as a member of the first group, otherwise it is classified as a member of the second group.

4.2 Signature Recognition using Multiclass Support Vector Machine

Signature recognition is the procedure of determining to whom a particular signature belong to. In this work, the global and grid features are combined and used to differentiate among the signature images. These combined images are given to multiclass SVM to train it, so that particular signature image is recognized.

4.2.1 Signature Recognition Workflow

Signature recognition is essentially a writer identification problem, whose objective is to find the author of a test signature given a database of signature exemplars from different signers.

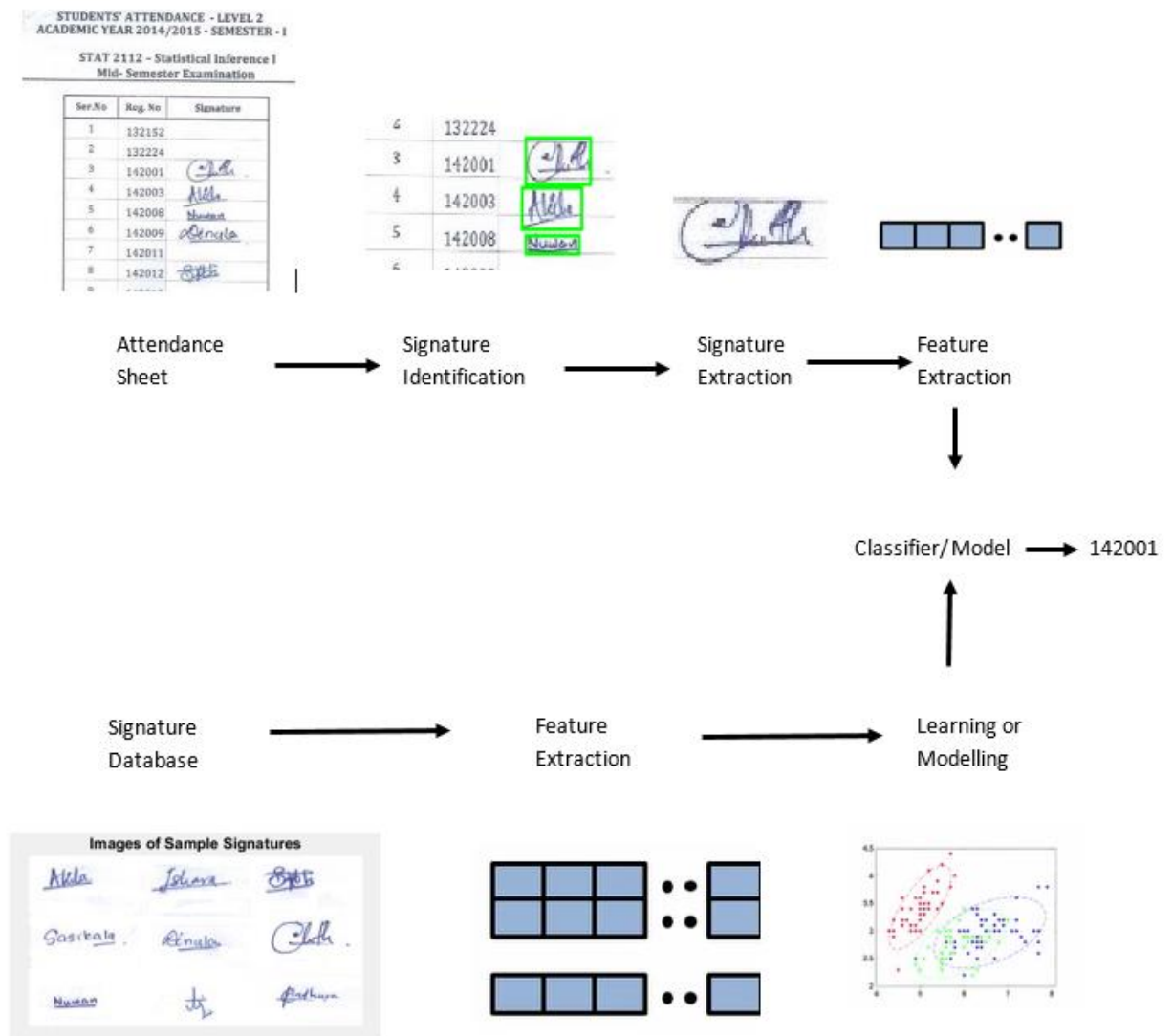


Figure 4.3: Signature Recognition Workflow

4.2.2 Error Correcting Output Code Multiclass Model

An *error-correcting output code multiclass model* (ECOC) reduces the problem of classification with three or more classes to a set of binary classifiers.

ECOC classification requires a coding design, which determines the classes that the binary learners train on, and a decoding scheme, which determines how the results (predictions) of the binary classifiers are aggregated. Suppose that there are three classes, the coding design is one-versus-one, the decoding scheme uses loss g , and the learners are SVMs. To build this classification model, ECOC follows these steps.

1. A one-versus-one coding design is

	Class 1	Class 2	Class 3
Learner 1	1	1	0
Learner 2	-1	0	1
Learner 3	0	-1	-1

Learner 1 trains on observations having Class 1 and Class 2, and treats Class 1 as the positive class and Class 2 as the negative class. The other learners are trained similarly. Let M be the coding design matrix with elements m_{kl} , and s_l be the predicted classification score for the positive class of learner l .

2. A new observation is assigned to the class (k) that minimizes the aggregation of the losses for the L binary learners. That is,

$$\hat{k} = \underset{k}{\operatorname{argmin}} \frac{\sum_{l=1}^L |m_{kl}| g(m_{kl} s_l)}{\sum_{l=1}^L |m_{kl}|}$$

ECOC models can improve classification accuracy, even compared to other multiclass models.

4.3 Signature Verification using Kolmogorov Smirnov Test

The performance task of signature verification is one of determining whether a questioned signature is genuine or not. The image of a questioned signature is matched against multiple images of known signatures.



Figure 4.4: Signature Verification where Questioned Signature(Right) is Matched Against Five Knowns.

Visual signature verification is naturally formulated as a machine learning task. Paralleling the learning tasks of the human questioned document examiner, the machine learning tasks can be stated as general learning (which is person-independent) or special learning (which is person-dependent). In the case of general learning the goal is to learn from a large population of genuine and forged signature samples. The focus is on differentiating between genuine-genuine differences and genuine-forgery differences.

Special learning focuses on learning from genuine samples of a particular person. The focus is on learning the differences between members of the class of genuine. The verification task is essentially a one-class problem of determining whether the questioned signature belongs to that class or not.

In this thesis, only the genuine signatures of the students were considered, so that person dependent learning was used in verification process.

4.3.1 Person Dependent Learning (Person Specific Learning)

In questioned document case work, there are typically multiple genuine signatures available. They can be used to learn the variation across them, so as to determine whether the questioned signature is within the range of variation. First, pairs of known samples are compared using a similarity measure to obtain a distribution over distances between features of samples, this represents the distribution of the variation/similarities amongst samples for the individual. The corresponding classification method involves comparing the questioned sample against all available known samples to obtain another distribution in distance space. The Kolmogorov-Smirnov test can be used to obtain a probability of similarity of the two distributions, which is the probability of the questioned sample belonging to the ensemble of knowns.

4.3.2 Within-person Distribution

If a given person has N samples, $\binom{N}{2}$ defined as $N! / N! (N-r)!$ pairs of samples can be compared as shown in Figure 4.5. In each comparison, the distance between the features is computed. This calculation maps feature space to distance space. The result of all $\binom{N}{2}$ comparisons is a $\{\binom{N}{2} \times 1\}$ distance vector. This vector is the distribution in distance space for a given person. For example, in the signature verification problem this vector is the distribution in distance space for the ensemble of genuine known signatures for that writer. A key advantage of mapping from feature space to distance space is that the number of data points in the distribution is $\binom{N}{2}$ as compared to N for a distribution in feature space alone. Also, the calculation of the distance between every pair of samples gives a measure of the variation in samples for that writer. In essence, the distribution in distance space for a given known person captures the similarities and variation amongst the samples for that person. Let N be the total number of samples and $N_{WD} = \binom{N}{2}$ be the total number of comparisons that can be made which also equals the length of the within-person distribution vector. The within-person distribution can be written as

$$D_W = (d_1, d_2, \dots, d_{N_{WD}})^T \quad (1)$$

where T denotes the transpose operation and d_j is the distance between the pair of samples taken at the j^{th} comparison, $j \in \{1, \dots, N_{WD}\}$.

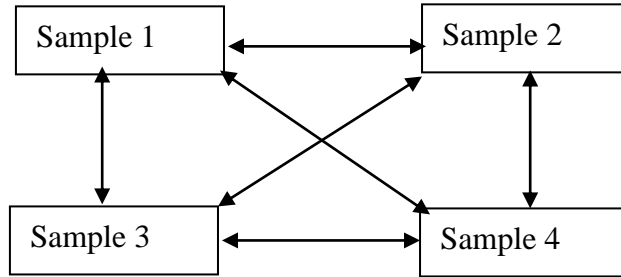


Figure 4.5: Comparing All Possible Genuine-Genuine Pairs

4.3.3 Person Dependent Classification

When multiple genuine are available then the within-person distribution is obtained in accordance with equation 1. A questioned can be compared against the ensemble of knowns for verification. The classification process consists of two steps.

- (i) obtaining questioned vs known distribution; and
- (ii) comparison of two distributions: questioned vs known distribution and within-person distribution.

Questioned vs Known Distribution in Section 4.3.2 and with equation 1 the within-person distribution is obtained by comparing every possible pair of samples from within the given person's samples. Analogous to this, the questioned sample can be compared with every one of the N knowns in a similar way to obtain the questioned vs known distribution. The questioned vs known distribution is given by

$$D_{QK} = (d_1, d_2, \dots, d_N)^T \quad (2)$$

where d_j is the distance between the questioned sample and the j^{th} known sample, $j \in \{1, \dots, N\}$.

4.3.4 Comparing Distributions

Once the two distributions are obtained, namely the within-person distribution, denoted D_w (Section 4.3.2, equation 1), and the Questioned Vs Known distribution, D_{QK} (Section 4.3.3, equation 2), the task now is to compare the two distributions to obtain a probability of similarity. The intuition is that if the questioned sample did indeed belong to the ensemble of the knowns, then the two distributions must be the same (to within some sampling noise). There are various ways of comparing two distributions and these are described in the following sections.

4.3.5 Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov (KS) test can be applied to obtain a probability of similarity between two distributions. The KS test is applicable to unbinned distributions that are functions of a single independent variable, that is, to data sets where each data point can be associated with a single number (Srinivasan et al., 2006). The test first obtains the cumulative distribution function of each of the two distributions to be compared, and then computes the statistic, D , which is a particularly simple measure: it is defined as the maximum value of the absolute difference between the two cumulative distribution functions. Therefore, if comparing two different cumulative distribution functions $S_{N1}(x)$ and $S_{N2}(x)$, the KS statistic D is given by $D = \max_{-\infty < x < \infty} |S_{N1}(x) - S_{N2}(x)|$. The statistic D is then mapped to a probability of similarity, P , according to equation 3

$$P_{KS} = Q_{KS}(\sqrt{N_e} + 0.12 + (0.11 / \sqrt{N_e})D) \quad (3)$$

where the $Q_{KS}(\cdot)$ function is given by:

$$Q_{KS}(\lambda) = 2 \sum_{j=1}^{\infty} (-1)^{j-1} e^{-2j^2 \lambda^2}$$

$$\text{such that: } Q_{KS}(0) = 1, \quad (4)$$

$$Q_{KS}(\infty) = 0$$

and N_e is the effective number of data points, $N_e = N_1 N_2 (N_1 + N_2)^{-1}$, where N_1 is the

number of data points in the first distribution and N_2 the number in the second. The following sections discuss other methods of comparing two distributions.

4.3.6 Person-Dependent Method

In order to measure error rates for this classification technique, once again a decision needs to be made based on the probability of whether or not the questioned sample belongs to the ensemble of knowns. If the probability of match $> \alpha$, then the decision is in favor of the questioned signature to be genuine, and if the probability of match $< \alpha$, the decision is in favor of a forgery.

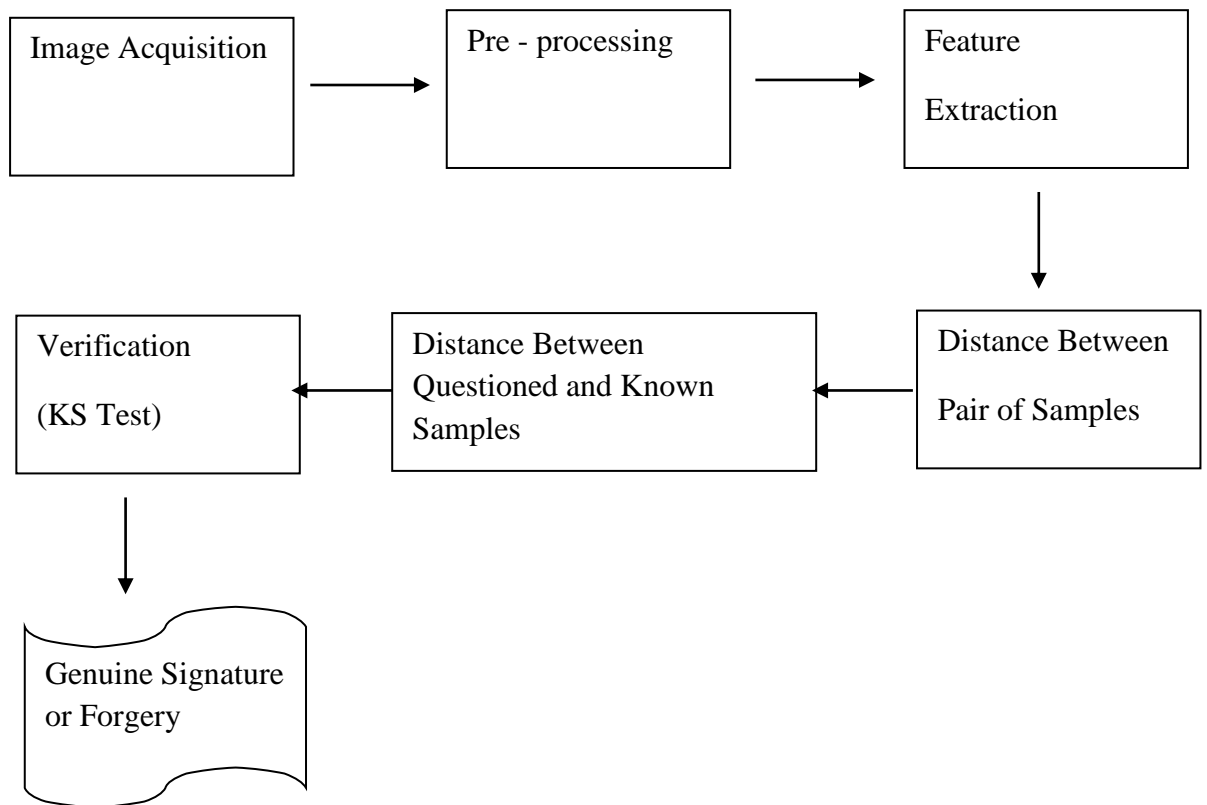


Figure 4.6: The Steps of KS Test for Signature Verification

Experimental Setup

5.1 Signature Identification and Extraction from Scanned Attendance Sheets

In the signature identification phase, signatures should be extracted from the attendance sheets to collect the signatures for testing process. At the beginning attendance sheets are preprocessed to isolate the signatures from the machine printed text, logos, diagrams, etc. Following methods are carried out for preprocessing:

5.1.1 Convert RGB to Grayscale

In present technology, almost all image capturing and scanning devices use color. Therefore, we also used a color scanning device to scan signature images. A color image consists of a coordinate matrix and three-color matrices. Coordinate matrix contains x, y coordinate values of the image. The color matrices are labeled as red (R), green (G), and blue (B). Techniques presented in this study are based on grey scale images, and therefore, scanned color images are initially converted to grey scale.

$$\text{Gray color} = 0.299 * R + 0.5876 * G + 0.114 * B$$

gray image
 Department of Mathematical Sciences
 Faculty of Applied Sciences
 Wayamba University of Sri Lanka
STUDENTS' ATTENDANCE - LEVEL 2
ACADEMIC YEAR 2014/2015 - SEMESTER - I
STAT 2112 - Statistical Inference I
Mid- Semester Examination

Reg. No	Signature	Reg. No	Signature
132224	Niranga	142019	H
142001	Alfa	142022	Esther
142003	Alfa	142025	D.
142008	Nuwara	142027	Manojana
142009	Abenula	142031	R
142012	Siti	142032	R
142013	Sasibala	142033	Methi
142016	Isma	142034	Abel
142017	Ed	142038	KBS
142018	Adyana	142040	Elasidha
		142044	Madusani

Figure 5.1: Grayscale Image of Scanned Attendance Sheet

5.1.2 Binarization

To obtain the white and black pixels of signature, the original signature is converted into binarized form. Otsu's method is implemented for image binarization which automatically executes clustering based Thresholding to convert the image into binary form. The pixels having intensity greater than a threshold are converted to white and less than the threshold is converted to black as shown in the figure below. The thresholding value is calculated automatically by the system and the value was 0.7294.

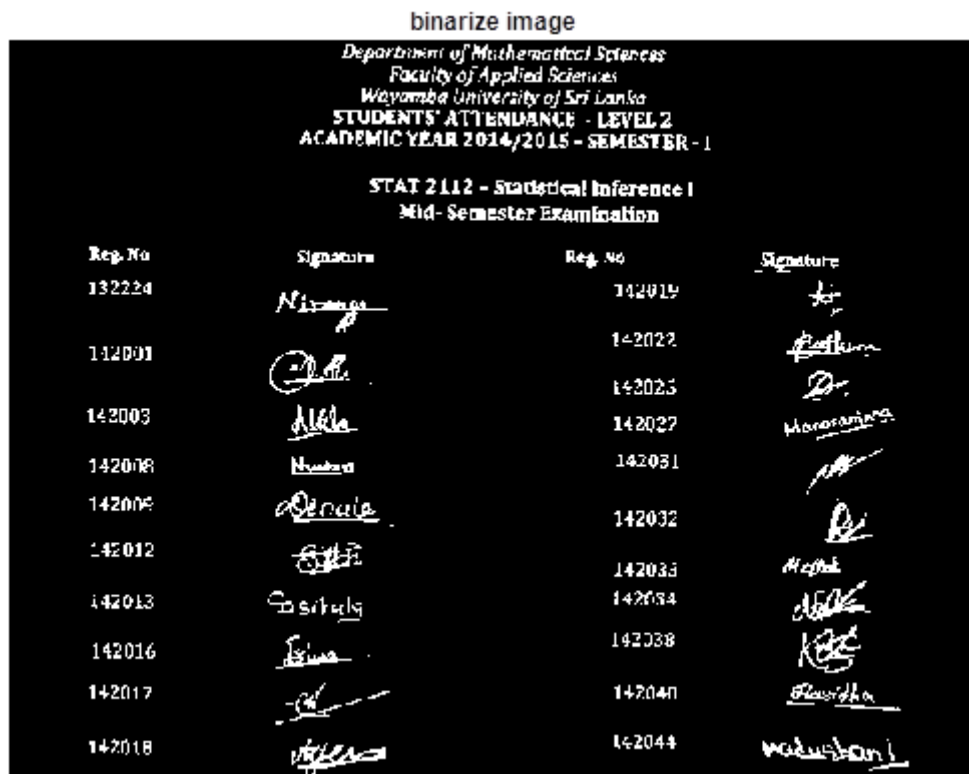


Figure 5.2: Binarize Image of Scanned Attendance Sheet

Working in this form is more useful than any other form, since it is easy to work with 2 bits' representation of the image.

5.1.3 Edge Detection

Edge detection is one of the most commonly used operations in image analysis. An edge is defined by a discontinuity in gray level values. In other words, an edge is the boundary between an object and the background. The shape of edges in images depends on many parameters. In this work, we are using 'Canny' edge detection. The 'Canny' operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial frequency that correspond to edges. Typically, it is used to find the approximate absolute gradient magnitude at each point in an input grayscale as show in Figure 5.3.

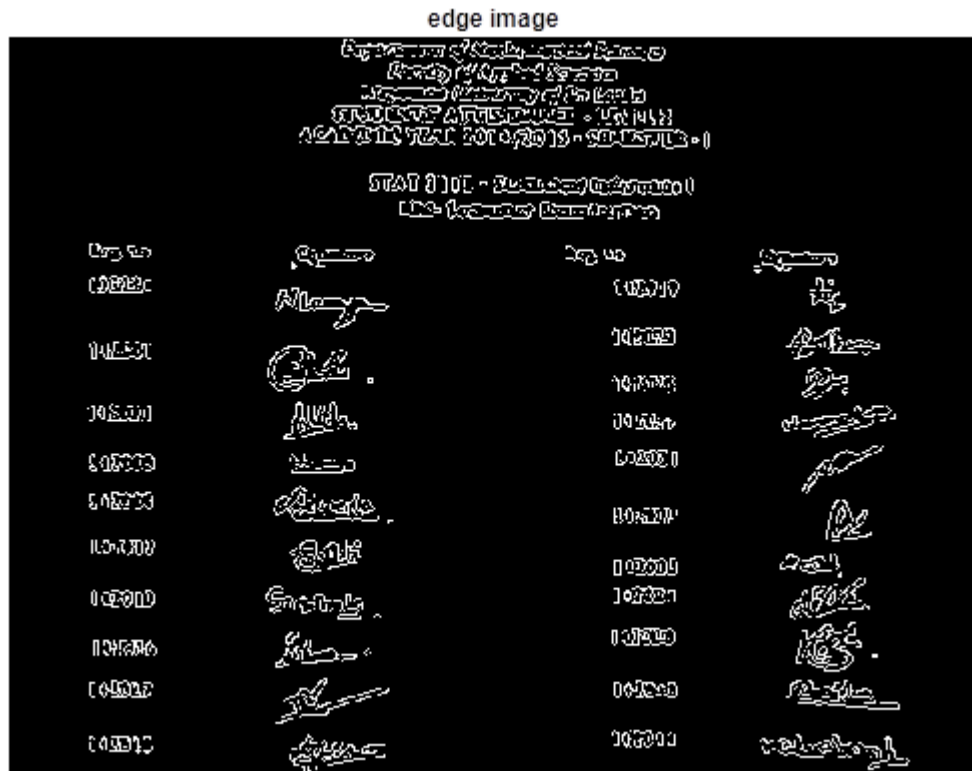


Figure 5.3: Detected Edges of Scanned Attendance Sheet

5.1.4 Remove Unnecessary Pixels

Using `bwareaopen()` unwanted pixels are removed from the attendance sheets. In attendance sheets signatures, have more connected pixels than texts. So, using this function removes all connected components that have fewer than 20 pixels from the binary image.

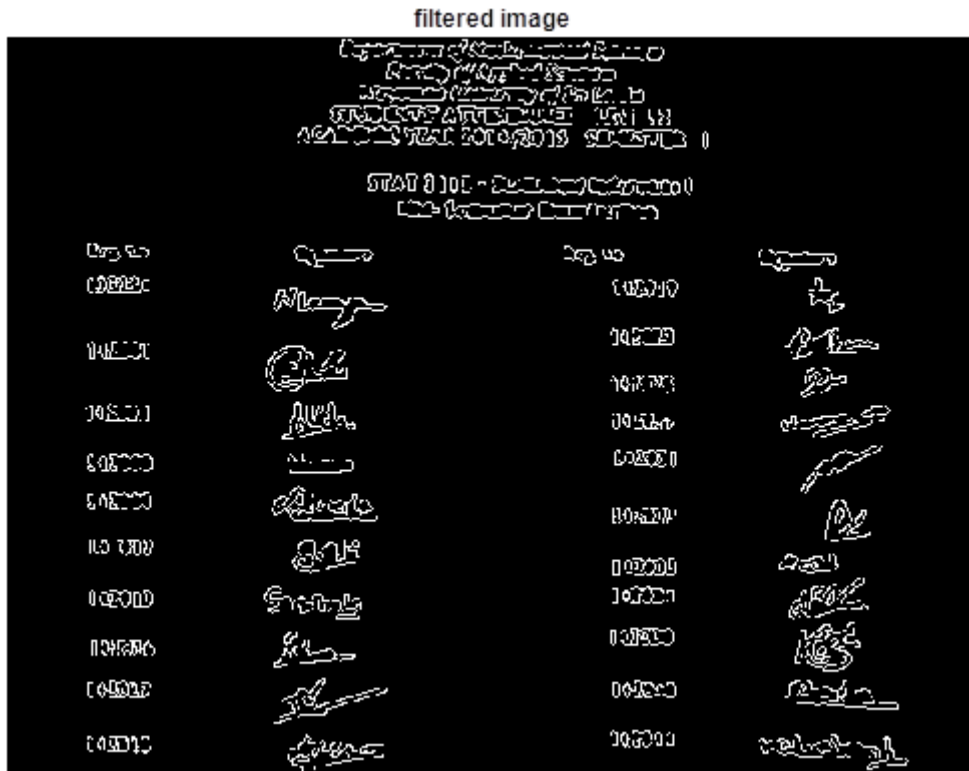


Figure 5.4: Filtered Image of Scanned Attendance Sheet

5.1.5 Morphological Dilation

Dilation is one of the two basic operators in the area of mathematical morphology, the other being erosion. It is typically applied to binary images, but there are versions that work on grayscale images. The basic effect of the operator on a binary image is to gradually enlarge the boundaries of regions of foreground pixels (i.e. white pixels, typically). Thus, areas of foreground pixels grow in size while holes within those regions become smaller. Dilation thickens the image and increases the no. of illuminated pixels there by giving a 'bold' look to the image.

Morphological operation is often performed in digital image processing and deals with shape of signature image. And also, it is a technique to creating an image noise free. In this phase, we have used close operation on the signature image (dilation followed by erosion).



Figure 5.5: Scanned Attendance Sheet by Applying Close Operator

We did not use thinning because of this operator may cause loss of information, but using thicken and bridge operator save the greatest amount of information, since they keep the boundary of signature.

5.1.5.1 Thicken

With $n = \text{Inf}$, thickens objects by adding pixels to the exterior of objects until doing so would result in previously unconnected objects being 8-connected.



Figure 5.6: Scanned Attendance Sheet by Applying Thicken Operator

5.1.5.2 Bridge

In our approach, we used bridge to connect discontinuity of pixels. Bridges unconnected pixels, that is, sets 0-valued pixels to 1 if they have two nonzero neighbors that are not connected.



Figure 5.7: Bridged Image of Scanned Attendance Sheet

5.1.6 Image Segmentation and Cropping

Segmentation is a crucial step in signature detection. The objective of segmentation is to partition an image into regions. Image segmentation here is typically used to locate objects especially words printed as well as hand-written that form regions of interest. Here the segmentation technique, `regionprops()` was used for finding the regions directly. Figure 5.8 shows a sample document showing the result of the segmentation process and each patch is marked with a box around it.

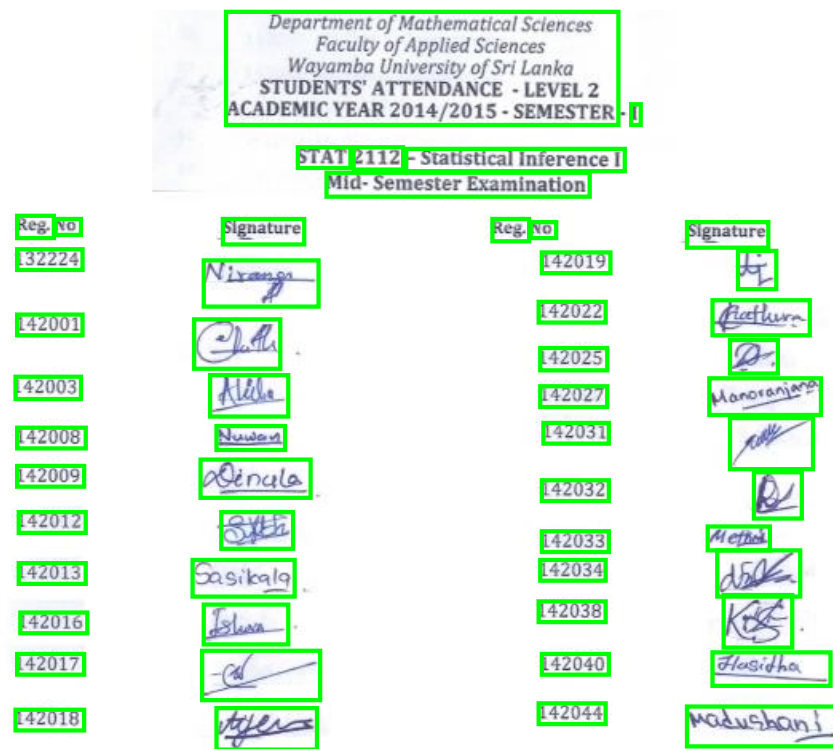


Figure 5.8: Scanned Attendance Sheet Marked Patches with a Box

5.2 Signature Classification using Binary SVM

After extracting the images from attendance sheets, the segmented images should be classify using a set of representative features. Here used features of segmented images in distinguishing signature and non-signature segments over a binary SVM classifier. Extracted signatures and non-signature area from multiple scanned attendance sheets were used to train the SVM and another attendance sheet was used to test the result.

5.3 Signature Recognition

5.3.1 Signature Acquisition

Handwritten signatures were taken from 105 students who followed Statistical Inference-I course module in semester-I of academic year 2014/2015, in Faculty of Applied Sciences, Wayamba University of Sri Lanka. Signatures were scanned and

stored in JPEG format. In this process, only the genuine signatures are taken and stored in the database.



Figure 5.9: Sample of Signatures

The image of the signature is a special type of object when treated as the subject of the recognition process. One of the problems which is likely to arise is that the signatures of a particular person are not exactly the same. Of course, during the application of the recognition system we may require that the signatures should be made carefully but there are always some differences we must deal with. This requires that the identification system should be flexible and allow certain variations within the set of the signatures put down by one person. So, that seven signatures were taken from each student and 735 signatures were stored in the database.



Figure 5.10: Variation of a Signature of The Same Person

5.3.2 Image Preprocessing

The scanned real-world images containing human signatures are processed using several image processing algorithms. These processes are given below.

5.3.2.1 Convert RGB to Gray Scale

A color image consists of a coordinate matrix and three-color matrices. Coordinate matrix contains x, y coordinate values of the image. The color matrices are labeled as red (R), green (G), and blue (B). Techniques presented in this study are based on grey scale images, and therefore, scanned color images are initially converted to grey scale.

$$\text{Gray color} = 0.299 * R + 0.5876 * G + 0.114 * B$$



Figure 5.11: Grayscale Signature Image

5.3.2.2 Binarization

To obtain the white and black pixels of signature, the original signature is converted into binarized form. Otsu's method is implemented for image binarization which automatically executes clustering based Thresholding to convert the image into binary form. The pixels having intensity greater than a threshold are converted to white and less than the threshold is converted to black as shown in the figure below. The thresholding value is calculated automatically by the system and the value was 0.7647.



Figure 5.12: Binarized Signature Image

5.3.2.3 Remove Unnecessary Pixels

Using `bwareaopen()` unwanted pixels are removed from the attendance sheets. In attendance sheets signatures, have more connected pixels than texts. So, using this function removes all connected components that have fewer than 10 pixels from the binary image.



Figure 5.13: Signature Image After Removing Unwanted Pixels

5.3.2.4 Thinning

Thinning operation is very important. From thinned character, many features are extracted. The goal of thinning is to eliminate the thickness differences of pen by making the image one pixel thick. Boundary pixels with limited number of surrounding pixels are selected. Surrounding center Pixel of selected boundary pixels are also selected. Now, boundary pixels are removed based on

1. Surrounding selected center pixel should remain.
2. No discontinuity should arise.

In this mode, it is commonly used to tidy up the output of edge detectors by reducing all lines to single pixel thickness. Thinning is normally only applied to binary image, and produces another binary image as output.



Figure 5.14: Signature Image After Thinning

Thinning was introduced to describe the global properties of objects and to reduce the original image into a more compact representation. Here uses a Zhang-Suen algorithm for thinning process.

5.3.2.5 Auto Cropping of Signature

Using cropping we segment the signature smoothly. Signature cropping process is less complexity in process and time, since the area under process will be reduced. Following figure shows the result.



Figure 5.15: Signature Image with Area Cropped

Here automatic cropping was used. It is saving more work and it is reducing a processing time over and above the cropping rectangle is truly detecting. In auto cropping approach,

firstly, determine the positions of ones in image, then calculate minimum and maximum coordinates from these positions, minimum coordinate will be the first corner (upper-left), the second one (lower-right) will be determined by subtract minimum coordinate from maximum coordinate. After that image cropping, will be used with these corners as:

```
I2=imcrop (I1, x1 y1 x2 y2);
```

where;

I2: cropped image (output),

I1= original image (input),

x1, y1: first corner,

x2, y2: second corner.

According to this mechanism a good and true cropping rectangle was determined, so no lose in the information meaning of the signature object.

5.3.3 Feature Extraction

Before the feature extraction process to increase the accuracy of the system signature image was partitioned into 4 equal parts and extract features from each part. So, that the number of features are increased.

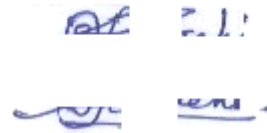


Figure 5.16: Partitioned Signature into 4 Parts

The choice of the features that will be provided to the classifiers of the system is very important. In this work, we use global features. Global features are classic in pattern recognition problems. The global features provide information about specific cases concerning the structure of the signature. Following features are extracted as follows:

5.3.3.1 Global Features

1. Pure Width: The width of the image with horizontal blank spaces removed.
2. Pure Height: The height of the image with vertical blank spaces removed.
3. Baseline Shift: The difference between the vertical centers of gravity of the left and the right part of the image. It was taken as a measure for the orientation of the signature.
4. Skewness: This is a measure of symmetry, or more precisely, the lack of symmetry. A distribution, or data set, is symmetric if it looks the same to the left and right of the center point". Skewness can range from minus infinity to positive infinity. A distribution with an asymmetric tail extending out to the right is referred to as "positively skewed" or "skewed to the right," while a distribution with an asymmetric tail extending out to the left is referred to as "negatively skewed" or "skewed to the left."
5. Kurtosis: "Kurtosis" is any measure of the "peakedness" of the probability distribution of a real-valued random variable. Kurtosis is a descriptor of the shape of a probability distribution and, just as for skewness; there are different ways of quantifying it for a theoretical distribution and corresponding ways of estimating it from a sample from a population. The measurement of skewness allows us to determine how bowed are the lines in each segment of the signature. There are

various interpretations of kurtosis these are primarily peakedness (width of peak), tail weight etc.

6. Max. Projections:

Maximum Vertical Projection

The vertical projection of the signature image is calculated. The highest value of the projection histogram is taken as the maximum vertical projection.

Maximum Horizontal Projection

As above, the horizontal projection histogram is calculated and the highest value of it is considered as the maximum horizontal projection.

7. Vertical Center of Mass and Horizontal center of mass:

The vertical center C_y is given by,

$$C_y = \frac{\sum_{y=1}^{y_{max}} y \sum_{x=1}^{x_{max}} b[x, y]}{\sum_{x=1}^{x_{max}} \sum_{y=1}^{y_{max}} b[x, y]}$$

The horizontal center C_x is given by

$$C_x = \frac{\sum_{x=1}^{x_{max}} x \sum_{y=1}^{y_{max}} b[x, y]}{\sum_{x=1}^{x_{max}} \sum_{y=1}^{y_{max}} b[x, y]}$$

8. The Hough Transform

In the last stage, the Hough Transform (HT) is used. This algorithm searches a set of straight lines, which appears in the analyzed signature. The classical transformation identifies straight-lines in the signature image, but it has also been used to identifying of signature shapes. In the first step the HT is applied, where appropriate curve-lines are found. The analyzed signature consists of large number of straight lines, which were found by the HT.

5.3.3.2 Local Features

To increase the accuracy of the system grid based features are also extracted from the handwritten signatures. Here Histogram Orient Gradient (HOG) features are extracted

as grid features and combine them together with global features in recognition process. Total number of extracted HOG features was 2592.

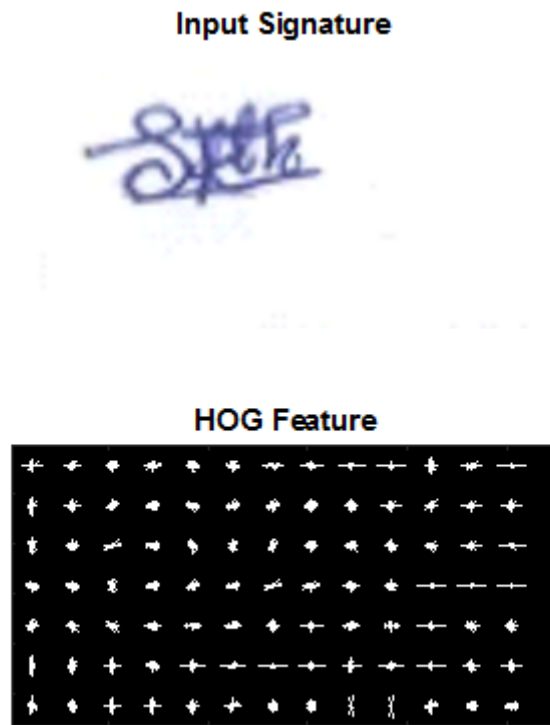


Figure 5.17: HOG Features Extracted from One Signature

5.4 Signature Verification

In the recognition process some genuine signatures are misclassified as another student's signatures. To verify the signatures as genuine only the correctly classified signatures were taken in verification process.

5.4.1 Algorithm for KS Test

This section offers algorithm for the offline signature verification system in which Kolmogorov Smirnov test is used to confirm the genuineness of signature.

Input = Signature image

Output = Conformation from system whether signature is genuine or counterfeit.

1. Acquire matched signature images from the signature recognition process

2. Enhanced the signature images by preprocessing
 - 2.1 Convert original image to gray scale image
 - 2.2 Binarization
 - 2.3 Elimination of unwanted pixels
 - 2.4 Thinning
 - 2.5 Signature Area Cropping

3. Extract the various features
 - 3.1 Global features
 - 3.1.1 Area
 - 3.1.2 Aspect Ratio
 - 3.1.3 Mean
 - 3.1.4 Standard Deviation
 - 3.1.5 Skewness
 - 3.1.6 Kurtosis
 - 3.1.7 Entropy
 - 3.1.8 Euler Number

4. Create a feature vector by combining extracted features from the pre-processed signature images.
5. Obtain the distances of features between each seven samples of the known signature in the database. (results gave 21×8 matrices)
6. Obtain the distances of features between known sample and questioned sample. (results gave 7×8 matrices)
7. Apply KS test for two distributions and obtain the probability of similarity.
8. Repeat step 1-7 to test all the signatures recognized by the system.
9. If the probability is less than 0.01 the signature was identified as “Forge”. Otherwise as “Genuine”.

Table 5.1: Sample Distance Distribution of Known Signature

1	2	3	4	5	6	7
0.440743	0.688526	0.521388	0.878166	1.014914	0.819797	0.935772
1.673837	2.444252	1.93806	2.950935	3.271665	2.802646	3.090603
122	98	139	67	38	69	43
0.27381	0.54329	0.672381	0.088745	0.305322	0.402381	0.285714
0.042208	0.072727	0.051543	0.1	0.121948	0.091223	0.109007
0.102306	0.165106	0.122437	0.214305	0.249553	0.199144	0.229222
5	7	9	7	8	5	5
0.039632	0.063377	0.047273	0.081689	0.094721	0.07609	0.087259

Table 5.2: Sample Distance Distribution of Known Vs Questioned

1	2	3	4		19	20	21
0.24778	0.08064	0.43742			0.19511	0.07914	0.11597
3	5	3	0.57417	...	6	2	5
0.77041	0.26422	1.27709	1.59782		0.46901	0.18106	0.28795
5	3	8	8	...	9	2	7
24	17	55	84	...	31	5	26
0.26948	0.39857	0.18506	0.03151		0.09705	0.01960	0.11666
1	1	5	3	...	9	8	7
0.03051	0.00933	0.05779	0.07974		0.03072	0.01294	0.01778
9	6	2	1	...	6	2	4
0.0628	0.020131	0.111999	0.147247	...	0.050409	0.020331	0.030078
2	4	2	3	...	3	3	0
0.023746	0.007642	0.042057	0.055089	...	0.018631	0.007462	0.01117

5.4.2 Writing Data to Excel Sheets

After identifying whether a particular signature is genuine or forge the attendance records are entered to the excel sheets. If the KS test identify the signature as genuine in verification process '1' is enter in front of the relevant student index number in the excel sheet.



Figure 5.18: Signature Recognized as Genuine by Verification Process

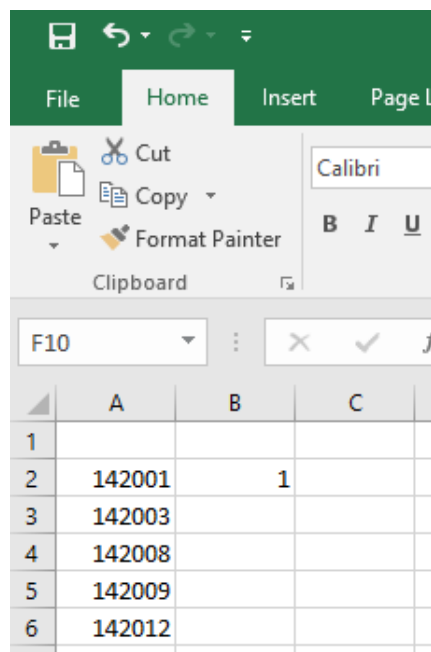


Figure 5.19: Enter Attendance Records into Excel Sheet

Results and Discussion

6.1 Signature Extraction Process

When extract the signatures from scanned attendance sheets in some situations there were some discontinuities in the signatures. In those situations, whole signature was not including in the bounded region as following figure (one signature was separated into parts).

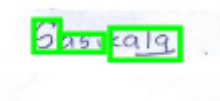


Figure 6.1: Signature Image with Multiple Bounded Regions Due to Discontinuity

To overcome that problem edge detection, morphological dilation, thicken and bridge was used in preprocessing stage. The basic effect of the edge detection is to find the edge pixels and using morphological dilation gradually enlarges the boundaries of regions of foreground pixels. Thus, areas of foreground pixels grow in size while holes within those regions become smaller. After that using thicken and bridge the unconnected pixels are connected. So, that the discontinuity of the signature was disappeared.

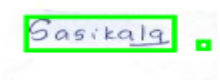


Figure 6.2: Signature Image After Removing the Discontinuity

6.1.1 Results Analysis in Signature Classification using Binary SVM

The accurateness of identifying signature areas of the proposed method is 100%. The Accuracy, FAR (False Acceptance Rate) and FRR (False Rejection Rate) of the succeeding formulas are as given below.

$$\text{Accuracy} = \frac{\text{true_positives} + \text{true_negatives}}{\text{num_of_data}}$$

$$\text{FAR} = \frac{\text{false_negatives}}{\text{true_positives} + \text{false_negatives}}$$

$$\text{FRR} = \frac{\text{false_positives}}{\text{false_positives} + \text{true_negatives}}$$

According to the formula FAR and FRR should be minimize for better performance. In our approach, it gives 100% accuracy so that FAR and FRR is zero.

6.2 Signature Recognition and Verification Process

The type of error we want to reduce at this moment is the rejection of the genuine signatures. On the other hand, in order to reduce misclassification and improve forgery resistance we must require that certain important features should be exactly recurrent and we must strictly demand their presence. The errors we are trying to minimize in this case are: acceptance of a fake signature and classifying one person's signature as belonging to another one. Incorporating those two aspects – acceptance of the variance and the requirement for exactness of certain features in one system is a very difficult task and still there is no perfect solution. The techniques developed so far give good results but they are still affected by a relatively significant error.

Following the above considerations, here focuses on the general shape of the signatures in order to prepare data for the first recognition step. Information acquired

at this stage should enable differentiation between the signatures given by different students and be general enough to reduce influence of the variations among different occurrences of the same signature. This stage should be supplemented by more precise local investigations to form the complete recognition system. Some signatures are misclassified by another student's signature due to some similarities between two signatures in the recognition process.



Figure 6.3: Misclassification of a Signature

In training phase signature database was partitioned in to two sets training set and test set. Among the signatures 70% were taken as training set and 30% were taken as test set. To test the accuracy of classifier test with another 30 signatures which are extracted from an attendance sheet. Among those 30 signatures 25 signatures recognized the actual signer and remain 5 were misclassified as another person's signatures. So, that the accuracy was 83.33% in the recognition phase. To clarify the genuineness of the signature Kolmogorov Smirnov test is applied with the correctly classified signatures in the recognition process.

6.2.1 Kolmogorov-Smirnov Test Performance Measure

For signature verification, the problem is to check the genuineness of the signature by comparing it with stored reference genuine signatures. In a verification system, there are two types of errors that can be made. If the system accepts an impostor it is a False Acceptance. The rejection of a valid signature is a False Rejection.

Correctly classified 25 signatures were test with KS test and measure the probability of similarity to check the genuineness of the signatures. All the signatures were identified as genuine signatures with the alpha values 0.05 and 0.01. The evaluation criteria were if the probability of similarity is greater than alpha value signature classified as 'Genuine', otherwise 'False'. To evaluate the performance of a

verification system, a set of trials is processed by varying the α value and the sample size of the signatures.

Conclusion

Today information technology has proved that there is a need to retrieve, search, query and store large amount of electronic information efficiently and accurately. In this thesis, we presented an approach to automate the student's attendance entering to excel sheets. To automate the process this thesis basically identified three phases. First one is extract the student's signatures from scanned attendance sheets for testing purpose. Second phase is recognizing the actual signer from the registered student's database. Last phase is verifying the recognized signature as genuine to eliminate forge signatures.

When extracting signature images from scanned documents first the discontinuity of signatures was removed. The whole part of the signature not extracted due to some reasons. Those are: excessive dusty noise, logos, figures, printed and handwritten text etc., large ink- blobs joining disjoint characters or components, degradation of printed text due to poor quality of paper and ink, text overlapping the signature. To avoid text and signature overlapping there should be some amount of space between text and signatures in the attendance sheet. After extracted the signature and non-signature parts those are classify and separate using binary SVM. It gives 100% accuracy by separating signature and non-signature parts.

In recognition process the combination of global and local features were used to train the multiclass SVM. As local features HOG features were extracted from each image. System was tested with 30 signatures and identified 83.33% signatures correctly so that the results suggest that the use of gradient-based feature sets with global features can serve the most reliable way of detecting signatures in signature recognition process. When increase the number of signatures in the database the processing time also increased and it was taken some amount of time to give the final result.

The signature verification is also important to eliminate forge signatures. A machine learning approach was used in signature verification process. Only the genuine signatures were in the registered student database. So, that person dependent learning approach, Kolmogorov Smirnov was used in the system. Only the correctly classified

signatures in recognition phase were taken for signature verification process. All the signatures were identified as genuine by the KS test. The person dependent comparison also appears to be a promising direction for future exploration. This task is attractive because it mirrors the situation of a real-world application of signature verification.

Finally, we can conclude that this system can be used in a university educational environment for automatic student authentication.

References

- [1] Banka, R., Nourbakhsh, F., 2010. EXTRACTION OF SIGNATURE AND HANDWRITTEN REGIONS FROM OFFICIAL BINARY DOCUMENT IMAGES.
- [2] Basavaraj, L., Samuel, R.S., 2009. Offline-line Signature Verification and Recognition: An Approach Based on Four Speed Stroke Angle. *Int. J. Recent Trends Eng.* 2.
- [3] Bebis, G., Deaconu, T., Georgiopoulos, M., 1999. Fingerprint identification using delaunay triangulation, in: *Information Intelligence and Systems, 1999. Proceedings. 1999 International Conference on. IEEE*, pp. 452–459.
- [4] Bhattacharyya, D., Kim, T., 2010. Design of artificial neural network for handwritten signature recognition. *Int. J. Comput. Commun.* 4, 59–66.
- [5] Coetzer, J., 2005. *Off-line signature verification*. Stellenbosch: University of Stellenbosch.
- [6] Cüceloğlu, İ., Oğul, H., 2014. Detecting handwritten signatures in scanned documents, in: *Proceedings of the 19th Computer Vision Winter Workshop*. pp. 89–94.
- [7] Gautam, C.M., Sharma, S., Verma, J.S., 2012. A GUI for Automatic Extraction of Signature from Image Document. *Int. J. Comput. Appl.* 54.
- [8] Gonzalez, R.C., Woods, R.E., 2002. *Digital image processing*. Pearson Education, Delhi, India.
- [9] Gupta, C.S., Dixit, U.D., 2015. A REVIEW ON SIGNATURE DETECTION AND SIGNATURE BASED DOCUMENT IMAGE RETRIEVAL 4.
- [10] Hafemann, L.G., Sabourin, R., Oliveira, L.S., 2015a. Offline handwritten signature verification-literature review.

- [11] Kalera, M.K., Srihari, S., Xu, A., 2004a. Offline signature verification and identification using distance statistics. *Int. J. Pattern Recognit. Artif. Intell.* 18, 1339–1360.
- [12] Karouni, A., Daya, B., Bahlak, S., 2011. Offline signature recognition using neural networks approach. *Procedia Comput. Sci.* 3, 155–161. .2010.12.027
- [13] Khuwaja, G.A., Laghari, M.S., 2011. Offline handwritten signature recognition. *World Acad. Sci. Eng. Technol.* 59, 1300–1303.
- [14] Kiani, V., Pourreza, R., Pourreza, H.R., 2009. Offline signature verification using local radon transform and support vector machines. *Int. J. Image Process.* 3, 184–194.
- [15] Kumar, P., Singh, S., Garg, A., Prabhat, N., 2013a. Hand written signature recognition & verification using neural network. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* 3.
- [16] Mandle, P., Shaligram, V., 2015. A Novel Image Matching Technique using SIFT and SURF 4.
- [17] Mehra, R., Gangwar, R.C., 2014. A Survey: Enhanced Offline Signature Recognition Using Neuro-Fuzzy and SURF Features Techniques Vol 5(3), 4350–4353.
- [18] Nguyen, V., Blumenstein, M., Muthukkumarasamy, V., Leedham, G., 2007. Off-line signature verification using enhanced modified direction features in conjunction with neural classifiers and support vector machines, in: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). IEEE, pp. 734–738.
- [19] Özgündüz, E., Şentürk, T., Karşılıgil, M.E., 2005. Off-line signature verification and recognition by support vector machine, in: *Signal Processing Conference, 2005 13th European*. IEEE, pp. 1–4.
- [20] Piyush Shanker, A., Rajagopalan, A.N., 2007. Off-line signature verification using DTW. *Pattern Recognit. Lett.* 28, 1407–1414.

- [21] Prashanth, C.R., Raja, K.B., Venugopal, K.R., Patnaik, L.M., 2012. DWT based Offline Signature Verification using Angular Features. *Int. J. Comput. Appl.* 52.
- [22] Purohit, N. (2010). *OFFLINE HANDWRITTEN SIGNATURE VERIFICATION* .
- [23] Pushpalatha, N., Gautam, A., 2014. Offline signature Verification using spatial domain feature sets and support vector machine. *Int. J. Emerg. Technol. Adv. Eng.* 4, 544.
- [24] Ramachandra, 2009. *Signature Verification using Graph Matching*, International Journal of Recent Trends in Engineering.
- [25] Shirdhonkar, M.S., Kokare, M.B., 2010. Discrimination between printed and handwritten text in documents. *IJCA Spec. Issue On*.
- [26] Singh, N., Kaushal, S., 2015. INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY OFF-LINE SIGNATURE RECOGNITION USING MODIFIED NEURAL NETWORKS APPROACH.
- [27] Sisodia, K., Anand, S.M., 2009. Off-line handwritten signature verification using artificial neural network classifier. *Int. J. Recent Trends Eng.* 2, 205–207.
- [28] Srinivasan, H., Srihari, S., 2009. Signature-Based Retrieval of Scanned Documents Using Conditional Random Fields, in: Argamon, S., Howard, N. (Eds.), *Computational Methods for Counterterrorism*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 17–32.
- [29] Srinivasan, H., Srihari, S.N., Beal, M.J., 2006. Machine learning for signature verification, in: *Computer Vision, Graphics and Image Processing*. Springer, pp. 761–775.
- [30] T.S. enturk. E. O' zgunduz. and E. Karshgil, "Handwritten Signature Verification Using Image Invariants and Dynamic Features," *Proceedings of the 13th European Signal Processing Conference EUSIPCO 2005, Antalya Turkey, 4th-8th September, 2005*.

- [31] Zimmer, A., Ling, L.L., 2003. A hybrid on/off line handwritten signature verification system, in: Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on. IEEE, pp. 424–428.

APPENDIX

Some Sample Codes

Code for Thinning

```
continue_it = 1;
    while continue_it
        BW_old=BW;
        BW_del=zeros(size(BW));
        for i=2:size(BW,1)-1
            for j = 2:size(BW,2)-1
                P = [BW(i,j) BW(i-1,j) BW(i-1,j+1) BW(i,j+1)
BW(i+1,j+1) BW(i+1,j) BW(i+1,j-1) BW(i,j-1) BW(i-1,j-1) BW(i-1,j)];
                if P(2)*P(4)*P(6)==0 && P(4)*P(6)*P(8)==0 &&
sum(P(2:end-1))<=6 && sum(P(2:end-1)) >=2
                    A = 0;
                    for k = 2:size(P(:,1))-1
                        if P(k) == 0 && P(k+1)==1
                            A = A+1;
                        end%if
                    end%for
                    if (A==1)
                        BW_del(i,j)=1;
                    end%if
                end%if
            end%for
        end%for

BW(find(BW_del==1))=0;

    for i=2:size(BW,1)-1
        for j = 2:size(BW,2)-1
            P = [BW(i,j) BW(i-1,j) BW(i-1,j+1) BW(i,j+1)
BW(i+1,j+1) BW(i+1,j) BW(i+1,j-1) BW(i,j-1) BW(i-1,j-1) BW(i-1,j)];
            if P(2)*P(4)*P(8)==0 && P(2)*P(6)*P(8)==0 &&
sum(P(2:end-1))<=6 && sum(P(2:end-1)) >=2
                A = 0;
                for k = 2:size(P(:,1))-1
```

```

        if P(k) == 0 && P(k+1)==1
            A = A+1;
        end%if
    end%for
    if (A==1)
        BW_del(i,j)=1;
    end%if
end%if
end%for
end%for

BW(find(BW_del==1))=0;

if prod(BW_old(:)==BW(:))
    continue_it=0;
end%if

end%while

```

Code for SVM classification

```

Dataset = 'C:\Users\HETC\Documents\MATLAB\Signature\vv';
Testset  = 'C:\Users\HETC\Documents\MATLAB\Signature\test';

width=70; height=30;
DataSet      = cell([], 1);

for i=1:length(dir(fullfile(Dataset, '*.jpg')))

    % Training set process
    k = dir(fullfile(Dataset, '*.jpg'));
    k = {k(~[k.isdir]).name};
    for j=1:length(k)
        tempImage      = imread(horzcat(Dataset, filesep, k{j}));
        imgInfo        = imfinfo(horzcat(Dataset, filesep, k{j}));
    end
end

```

```

        % Image transformation
        if strcmp(imgInfo.ColorType,'grayscale')
            DataSet{j} = double(imresize(tempImage,[width
height])); % array of images
        else
            DataSet{j} =
double(imresize(rgb2gray(tempImage),[width height])); % array of
images
        end
    end
end
TestSet = cell([], 1);
for i=1:length(dir(fullfile(Testset,'*.jpg'))

    % Training set process
    k = dir(fullfile(Testset,'*.jpg'));
    k = {k(~[k.isdir]).name};
    for j=1:length(k)
        tempImage = imread(horzcat(Testset,filesep,k{j}));
        imgInfo = imfinfo(horzcat(Testset,filesep,k{j}));

        % Image transformation
        if strcmp(imgInfo.ColorType,'grayscale')
            TestSet{j} = double(imresize(tempImage,[width
height])); % array of images
        else
            TestSet{j} =
double(imresize(rgb2gray(tempImage),[width height])); % array of
images
        end
    end
end
end

% we have 30 images and we divided it into two label groups here.
train_label = zeros(size(30,1),1);
train_label(1:15,1) = 1; % 1 = backgrounds
train_label(16:30,1) = 2; % 2 = signatures

% Prepare numeric matrix for svmtrain

```

```

Training_Set=[];
for i=1:length(DataSet)
    Training_Set_tmp = reshape(DataSet{i},1, 70*30);
    Training_Set=[Training_Set;Training_Set_tmp];
end

Test_Set=[];
for j=1:length(TestSet)
    Test_set_tmp = reshape(TestSet{j},1, 70*30);
    Test_Set=[Test_Set;Test_set_tmp];
end

% Perform first run of svm
SVMStruct = svmtrain(Training_Set , train_label, 'kernel_function',
'linear');
Group      = svmclassify(SVMStruct, Test_Set);

testSet = imageSet('test');

for i=1:testSet.Count
    if(Group(i,1)==1)

imwrite(read(testSet,i),fullfile('C:\Users\HETC\Documents\MATLAB\Sign
ature\b',[ 'B',num2str(i), '.jpg']));
        else

imwrite(read(testSet,i),fullfile('C:\Users\HETC\Documents\MATLAB\Sign
ature\s',[ 'S',num2str(i), '.jpg']));
        end
    end
end

```

Sample Code for KS test

```

for x = 1:size(trainingFeatures,2)
    D(x,:) = pdist(trainingFeatures(1:7,x));
end

for x = 1:size(trainingFeatures,2)
    %for j = 1:size(queryFeatures,2)

```

```

        DT(x,:) = abs(queryFeatures(x) -
trainingFeatures(1:7,x));
        %end
    end

    [H, P] = kstest2(D(:),DT(:), 'Alpha',0.01);

```

Code for signature extraction

```

clc;
x = imread('attendance.jpg');
x1 = rgb2gray(x);
figure;
imshow(x1);
title('gray image');

% binarization using Otsu method
threshold = graythresh(x1);
x2 = ~ imbinarize(x1,threshold);
figure;
imshow(x2);
title('binarize image');
x2 = edge(x2, 'Canny');
figure;
imshow(x2);
title('edge image');
x2 = bwareaopen(x2,10);

figure;
imshow(x);

% Label connected components
[L, Ne]=bwlabel(x2);

% Measure properties of image regions
prop = regionprops(L);
sign = cell(1,length(prop));

hold on

```



```
for n = 1:length(prop)

rectangle('Position',prop(n).BoundingBox,'EdgeColor','g','LineWidth',
2);
    rect = prop(n).BoundingBox;

    sign{n} = imcrop(x, rect);

imwrite(sign{n},fullfile('C:\Users\HETC\Documents\MATLAB\Signature\te
st',[ 'C',num2str(n),'.jpg']));

end

hold off
```