FEATURE EXTRACTION AND CLASSIFICATION FOR ONLINE HANDWRITTEN GURMUKHI CHARACTER RECOGNITION

A Dissertation submitted in fulfillment of the requirements for the Degree
of

MASTER OF ENGINEERING
in
Electronic Instrumentation & Control Engineering

Submitted by

Ramandeep Kaur
801451020

Under the Guidance of
Dr. M.D. Singh
Assistant Professor, EIED

2016

Electrical and Instrumentation Engineering Department
Thapar University, Patiala
(Declared as Deemed-to-be-University u/s 3 of the UGC Act., 1956)
Post Bag No. 32, Patiala – 147004
Punjab (India)

DECLARATION

I hereby certify that the work which is presented in dissertation entitled, "Feature Extraction and Classification for Online Handwritten Gurmukhi Character Recognition", in partial fulfillment of the requirements for the award of the degree of Master of Engineering in Electronic Instrumentation & Control, submitted to Electrical & Instrumentation Engineering Department of Thapar University, Patiala is as authentic record of my own work carried under the supervision of Dr. M.D. Singh. It refers others researcher's work which is duly listed in the reference section. The matter contained in this dissertation has not been submitted, neither in part nor in full to any other degree to any other university or institute except as reported in the text and references.

Ramandeep Kaur

Place: Patiala **Ramandeep Kaur**

Date: 15.7.16 **801451020**

It is certified that the above statement made by the student is correct to the best of my knowledge and belief.

Dr. M.D. Singh

Date: 15.7.16 Electrical and Instrumentation Engineering Department

Thapar University, Patiala - 147004

Countersigned by:

Dr. Ravinder Agarwal
Professor and Head
Electrical and Instrumentation Engineering Department
Thapar University
Patiala – 147004

Dr. S.S. Bhatia
Dean of Academic Affairs
Thapar University
Patiala –147004

ACKNOWLEDGEMENT

It has been an enriching experience to pursue ME from the esteemed institute. I take this opportunity to thank all people who have helped and guided me in successful completion of my work.

I would like to express my sincere gratitude to my guide, Dr. MD Singh, Assistant Professor, Electrical and Instrumentation Engineering Department, Thapar University for his able guidance and valuable inputs towards my research work. I am thankful to my guide for his time, patience and valuable comments. This research work would not have been possible without his support and encouragement.

I am thankful to Mr. Karun Verma, Assistant Professor, Computer Science and Engineering Department for his help in providing data for the thesis work. I would also like to thank for the guidance he has provided towards my work.

I would like to thank Dr. R. Agarwal, Professor and Head, Electrical and Instrumentation Engineering Department, Thapar University. I am also thankful to Mr. Nirbhowjap Singh, Assistant Professor and PG coordinator, Thapar University. I am thankful to all the faculty and non-teaching staff within the department as well as outside the department for all the help that I have received during my tenure.

I thank Dr. S.S. Bhatia, Dean of Academic Affairs, Thapar University during the tenure of my ME, for providing adequate facilities in department for research.

Finally, I would like to thank my parents for their love and support. They have encouraged me at every stage of my life. I thank God Almighty for being with me and guiding me through life.

Ramandeep Kaur

Ramandeep Kaur

15/7/16

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

KNN – K-Nearest Neighbor

MLP – Multilayer Perceptron

SVM – Support Vector Machines

OCR – Optical Character Recognition

PDA – Personal Data Assistant

HMM – Hidden Markov Model

BDD – Background Directional Distributions

PNN – Probabilistic Neural Network

TDNN – Convolution Time Delay Neural Network

FFT – Fast Fourier Transform

FD – Fourier Descriptor

ABSTRACT

Online handwriting character recognition is gaining attraction from the researchers across the world because with the advent of touch based devices, a more natural way of communication is being explored. Online system is real time processing in which characters are recognized as they are written. There are various issues associated with online recognition process. Due to variations in handwriting, it is very difficult to achieve high degree of accuracy. Therefore, the research work presented in the thesis aims to develop an efficient system to recognize the input natural handwriting. Script for which recognition is done is Gurmukhi script.

Stroke based approach is followed for online recognition of handwritten Gurmukhi characters because of the uniqueness of the strokes in comparison to characters. In the present work, 32 stroke classes have been considered and implemented for online character recognition of Gurmukhi script. Three types of features are extracted, namely Spatiotemporal features, Tangential features and Spectral features. In the thesis work, a hybrid method consisting of multiple features has been proposed to improve the performance of the recognizer. Two types of hybridization have been obtained. First, by combining Spatiotemporal and Tangential features and second, by combining Spatiotemporal and Spectral features.

Three different types of classifiers which are K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP), and Support Vector Machines (SVM) have been used for recognition. Recognition is implemented using two methods, namely cross validation technique and percentage split method. Highest accuracy is achieved using MLP and SVM using the hybrid features. KNN also observed good accuracy rate.

CHAPTER 1

INTRODUCTION

In this era, easiest way of communication between computers and users is natural handwriting. Hence, the online handwriting recognition system is emerging area of research today. Now everyone uses touch screen mobile phones, laptops, tablets and many more gadgets to fulfill their needs but users are still unable to use their native language as a system of communication with the electronic items of daily use. In Indian context, it is difficult to communicate with computers for scripts like Gurmukhi and Devanagiri due to large set of alphabets and complex nature of typing of these scripts. Therefore, it is required to develop a user interface in which users can communicate in their own handwriting with computers. Online handwriting recognition system can replace the need of entering data through keyboards which is time consuming process and is also not suitable to people who are illiterate. An efficient online handwriting recognition system can greatly improve the communication between users and computers [1,2,3].

There are various problems associated with handwriting recognition system. To achieve high degree of accuracy for handwriting recognition system is tedious task because of the variations in the writing styles. Variations in writing styles exist among different writers because every individual possesses their own style of writing in terms of speed, position or size of writing characters. Variation also exists within same writer due to different situations in which writer writes, it also depends upon the mood of the writer [4,5].

Online handwriting recognition system can also be called as real time processing because as we write characters they are recognized by the system. In this system, input is provided using pen based devices such as digitizer tablets, pads etc. Such device captures the sequence of (x,y) coordinates in the space as the character is written which can be called as a stroke. Stroke can be defined as trace of the tip of pen between Pen Down and Pen Up events [6]. Stroke formation is represented in the Figure 1.1. Red dot represents the initial position of the pen and final position is represented by blue dot, arrows indicate direction of the movement of the pen. In this case direction of writing is from left to right. This figure represents general stroke formation. Research in handwriting recognition has been going from decades and the system has achieved great advancement across the world. But still there is lot to improve as this system is still not reliable and results in poor accuracy in many cases.

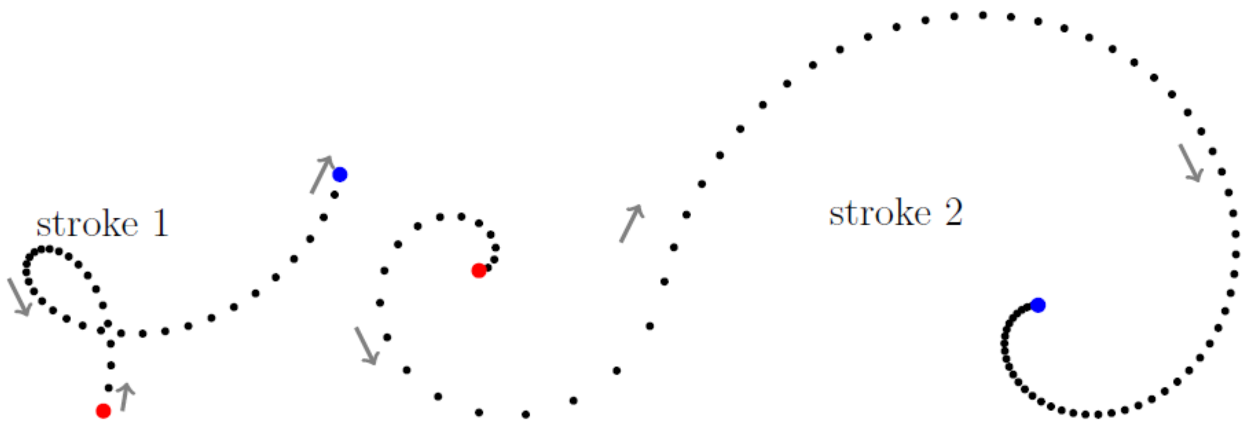


Figure 1.1: Online stroke formation in form of two dimensional (x,y) coordinates [7].

1.1 Types of Handwriting Recognition System

Handwriting recognition system can be classified into two types, offline and online recognition system [8] also shown in Figure 1.2:

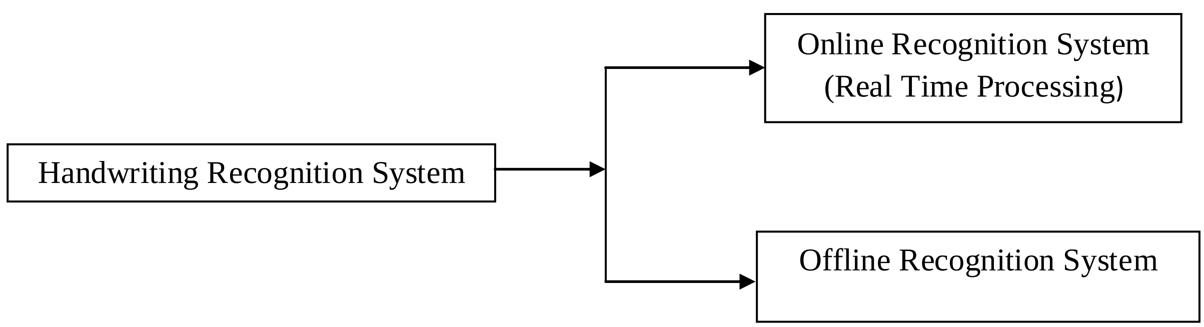


Figure 1.2: Types of Handwriting Recognition System

1.1.1 Offline Recognition System

In offline mode, camera or scanner captures the handwriting optically. It is also known as optical character recognition in which text is in form of image. In this type of system, characters are not recognized as they are written. Firstly, text is written by user on the paper which is then scanned and fed to computer in image format for recognition. In scanned images, individual characters needs to be extracted and therefore, comparatively difficult. Optical character recognition mainly focuses upon machine printed texts. OCR consists of many phases which include storage of grey scale image and further processing to separate characters in the scanned text document to extract desirable features which are further used for recognition. In offline system there is no real time interaction due to which it is not considered suitable for machine human interface [4,9].

1.1.2 Online Recognition System

Online recognition system is real time processing because characters are recognized as they are written. Online recognition system is represented in Figure 1.3. Electronic tablets are used to input characters through pen or stylus which are digital in nature. System captures the coordinates in space as the pen moves on the surface and coordinates are stored as a function of time. Online system in comparison to offline system is more users adaptable because if any character gets wrongly recognized, user by changing its writing style can correct the character in real time. Preprocessing algorithms should be faster in case of online system because they have to take less time for decision. Online recognition system consists of stages which are data acquisition, preprocessing of the acquired data to remove noise, important features are extracted which can best classify the data and the final stage is classification. In classification various models are used in order to map extracted features to different classes so that character can be identified [10].

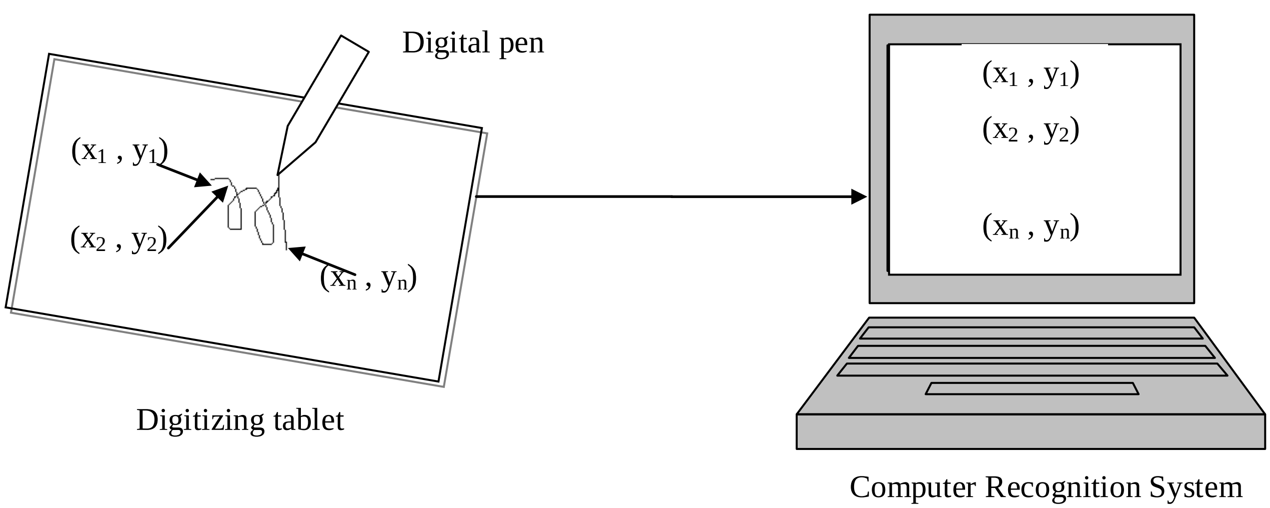


Figure 1.3: Representation of Online Recognition system [3].

In the thesis work, online handwriting recognition is performed for Gurmukhi script because of its various advantages like real time interactivity, user adaptability, easy preprocessing and segmentation [11].

1.2 Gurmukhi Script

The literal meaning of word Gurmukhi is "from the mouth of guru". The Guru Granth Sahib Ji is written in Gurmukhi script and this script was standardized by Guru Angad Dev Ji, the second guru of Sikhism during 16^{th} century. Gurmukhi is script used by Sikhs and Hindus primarily for writing Punjabi language [12]. Gurmukhi consists of three tones which are not represented in the writing system except for isolated use of higher tones. Punjabi is language spoken by 102 million native speakers across the globe making it world's 14^{th} most widely spoken language [4]. Gurmukhi characters are even older than Devanagiri. Gurmukhi script is cursive, written from left-to-right direction and in top-down approach. Gurmukhi script doesn't have concept of upper and lower case letters. Punjabi language consists of 41 consonants, 9 vowels (laga matras), addak to duplicate sound of consonant, bindi and tippi for nasal sounds, 3 subjoined forms of the consonant '*Rara*', '*Haha*' and '*Vava*' [1,13] .

Gurmukhi script is also known as 'painti akhri' as it contains thirty five characters and out of these characters three are different because they form basis for vowels and are not considered as consonants. They are the first three letters as shown in Table 1.1 '*urha*', '*erha*' and '*eerhi*'. These three characters are never used on their own. In addition to these 35 characters, there are six special consonants which are formed by placing dot in their foot [14,15].

A Gurmukhi script follows other Brahmi scripts. All consonants are followed by inherent 'a' sound and this inherent sound can be changed using dependant vowel signs. In some cases, dependant vowel sound cannot be used at the beginning of the letter and therefore independent vowel character is used instead [15]. A word in Gurmukhi script can be partitioned into three horizontal zones, upper, middle and lower zone. Upper zone is region above the headline where some vowels and sub-parts of vowels reside; middle zone is area below the headline where consonants and some sub-parts of vowels reside and finally the lower zone area below middle zone where some vowels, half-characters reside. Upper and middle zone have a separation by headline which is also known as siro rekha. Gurmukhi is written below this line and has no concept of upper and lower case. It is written in logical fashion: vowel: firstly vowel, then consonants and semi-vowels [16]. Gurmukhi script gained Unicode standard in October, 1991 with the release of version 1.0 [15]. Character set of Gurmukhi script is shown in Table 1.1 where consonants, special consonants and vowels are represented along with their pronunciation [37] [40].

Table 1.1: Character set of Gurmukhi script

Basic Characters(Consonants)				
ੳ	ਅ	ੲ	ਸ	ਹ
URHA	ERHA	EERHI	SUSSA	HAHA
ਕ	ਖ	ਗ	ਘ	ਙ
KUKKA	KHUKHA	GUGGA	GHUGGA	UNGGA
ਚ	ਛ	ਜ	ਝ	ਞ
CHUCHA	CHHUCHHA	JUJJA	JHUJHA	YANZA
ਟ	ਠ	ਡ	ਢ	ਣ
TAINKA	THUTHA	DUDDA	DHUDDA	NAHNHA
ਤ	ਥ	ਦ	ਧ	ਨ
TUTTA	THATHA	DUDA	DHUDA	NUNNA
ਪ	ਫ	ਬ	ਭ	ਮ
PUPPA	PHUPHA	BUBBA	BHUBBA	MUMMA
ਯ	ਰ	ਲ	ਵ	ੜ
YAIYYA	RARA	LULLA	VAVA	RAHRHA

Special Consonants				
ਸ਼	ਖ਼	ਗ਼	ਜ਼	ਫ਼
SUSSA PAIR BINDI	KHUKHA PAIR BINDI	GUGGA PAIR BINDI	JAJJA PAIR BINDI	PHUPHA PAIR BINDI
ਲ਼				
LALLA PAIR BINDI				
Vowels				
◌਼	◌ਾ	ਿ◌	◌ੀ	◌ੇ
MUKTA	KANNA	SIHARI	BIHARI	LAVAN
◌ੈ	◌ੁ	◌ੂ	◌ੋ	◌ੌ
DULAVAN	ONKAR	DULANKAR	HORA	KANAURA

1.2.1. Zones in Gurmukhi script

Gurmukhi script consists of three zones:

Upper zone: It is the region above headline where only vowels are found.

Middle zone: This is the area between upper and lower zone where consonants and some parts of vowels reside.

Lower zone: This is the area below the middle zone which is place for vowels, halant and certain half characters which lie at the foot of the consonants. Figure 1.4 represents the three zones for the Gurmukhi script [17].

Figure 1.4: Upper, Middle and Lower zone in Gurmukhi word [18].

1.3 Issues in online handwriting recognition system

For Indian handwriting system, recognition is a tedious task due to its complex nature. This system can become unreliable due to variation in the writing style and distortions produced due the digitizing process. Online recognition system uses electronic devices like digitizing tablets, PDAs, cross pads in which pen or stylus is used to write characters after which the computer converts handwritten text into digital form. High degree of accuracy is required to use these devices for recognition procedure which are acceptable to the user. The main issues of online handwriting system are as follows:

1.3.1 Variation in handwriting styles

Different writers possess different writing styles. Sometimes even the same person is unable to recognize his or her own handwriting. These variations caused has geometrical nature which includes position, aspect ratio of the strokes, size, retraces, slants, skews, number of strokes in a character, overlapping and deformed geometry. Hence it becomes a tedious task for the recognizer to achieve accuracy. Figure 1.5 represents Gurmukhi characters written by same person and variation can be seen in size, shape etc for the same letter. Figure 1.6 represents Gurmukhi characters written by two different writers and writing style, size and shape is different [3,19].

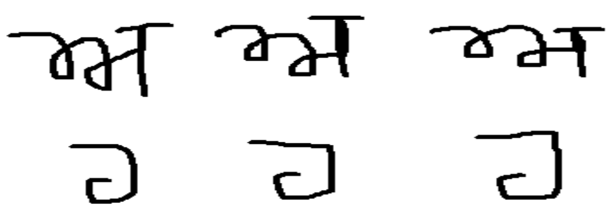


Figure 1.5: Variation in handwriting by same writer.

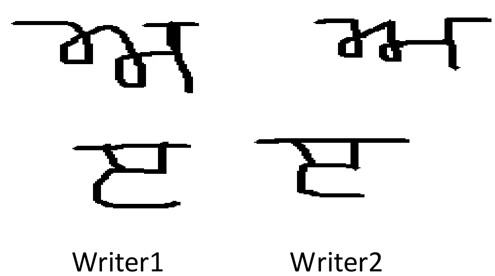


Figure 1.6: Variation in handwriting by two writers.

1.3.2 Constrained and Unconstrained Handwriting

Handwriting styles can be classified as constrained or unconstrained. Constrained handwriting can be boxed or spaced discrete in nature. Unconstrained handwriting can be cursive or mixed cursive in nature. Figure 1.7 shows the boxed discrete handwriting in which each character is written inside a special box. In spaced discrete handwriting, each character is written with space inside and no character is in contact with other character. In run on discrete handwriting, characters are spaced apart but touch each other. In cursive handwriting, characters are connected in a word and strokes are used more than one in the individual character. Mixed cursive style is combination of spaced, run on and cursive styles of handwriting. For a recognizer, it is difficult task to obtain accuracy for cursive styles of writing due to great amount of variability and no clear boundaries are marked among characters which can distinguish them. Figure 1.8 shows spaced, run on discrete, cursive and mixed cursive writing styles [3].

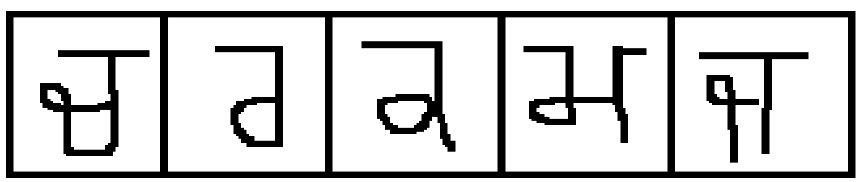


Figure 1.7: Boxed discrete writing style in Gurmukhi script.

Figure 1.8: Different writing styles of word in Gurmukhi script.

1.3.3 Personal, Situational and Material aspects

Personal aspects involve the writer’s handedness that whether writer is left or right handed. It has been observed that writer’s handedness results in different positions and directions of handwriting. Writing style of person also depends upon profession up to some extent. For good recognition it is required that handwriting should be neat and clean.

Situational factors involve the mood of the writer whether he or she is in good or bad mood and writer’s way of presentation. These factors involve the situation of the writer that he wants to write or not, any interruption occurred while writing. Material factors deal with the hardware used for writing. This includes whether the hardware is comfortable or uncomfortable for writer to write which can result into variations. Size and position of writing pad also matters. Figure 1.9 represents the difficulty in recognizing different writing styles [3].

Boxed discrete writing style	Easy
Spaced discrete writing style	↓
Run on discrete writing style	↓
Pure cursive writing style	↓
Mixed cursive, discrete, run on writing style	Difficult

Figure 1.9: Recognition difficulty for different writing styles.

1.3.4 Writer dependant versus writer independent systems

Writer dependant recognition system involves training of the system for all possible variations produced by single or group of writers. This type of system is based on known writing styles. This type of system can only recognize the samples of those writers whose samples are already taken to train the recognition system.

Writer independent recognition system is a generalized system as it can also recognize the handwriting of unknown writers. This system is more difficult to develop in comparison to writer dependant system. In this system, training has to be done for all commonly and possible style variations used. This system includes collection of huge samples from large number of writers due to which it has complex nature. Independent systems have lower recognition rates but are in demand due to generalized applications [3].

1.4 Objectives of the Work

To date, extensive research has been carried out in the field of handwriting recognition for different languages like English, Chinese, Korean and Japanese. But limited amount of work has been carried out for online recognition of Gurmukhi script in the recent past [1]. Different features and classifiers have been explored by various researchers in handwriting recognition to obtain good efficiency and accuracy but yet there is lot to achieve. The objective of this work is to develop a system which is writer independent for recognition of online handwritten Gurmukhi script. This task is accomplished by adopting preliminary approach to identify strokes by proposing hybridization scheme for features and by using these features to train the different classifiers for obtaining better accuracies. The methodology adopted for this purpose is as follows:

1. Online recognition involves data acquisition, preprocessing to remove distortions, feature extraction stage and finally recognizing the strokes to identify the character or word.
2. Data acquisition is done through pen device in form of two dimensional positional (x,y) coordinates sequence corresponding to each stroke.
3. Preprocessing stage removes the noise from the strokes and provides uniform representation. From the preprocessed stroke, different features are extracted in order to improve the recognition rate.
4. Finally, recognition is done with three different classifiers K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP) and Support Vector Machines (SVM). Figure 1.10 represents the block diagram of the online handwritten character recognition.

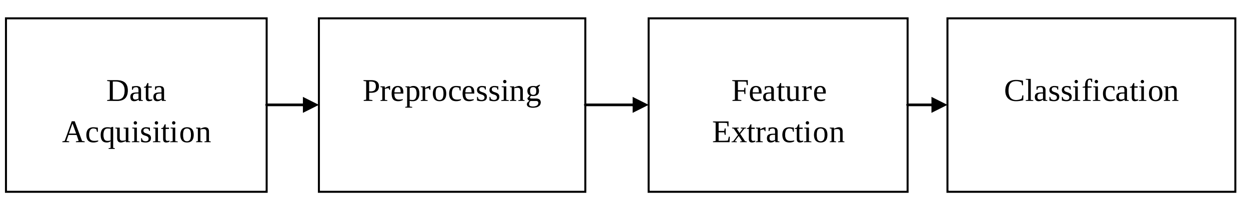


Figure 1.10 Block Diagram of Online Handwriting Recognition System

1.5 Organization of the Thesis

In the present chapter, types of recognition systems, Gurmukhi script overview and various issues related to online recognition system has been discussed. Chapter 2 provides literature review which explains the background knowledge in the area of online recognition systems. It presents the various methods used to recognize in the past and how much accurate these methods. This chapter also has tabular representation of the work done by different authors and the accuracy they achieved. Chapter 3 deals with the methodology adopted in online handwritten Gurmukhi character recognition. This chapter explains the various phases of online recognition system which are data acquisition, preprocessing, feature extraction, classification. This chapter explains procedure of collecting strokes, various stages of preprocessing, computation of suitable features from the strokes and stroke recognition algorithms which include KNN, MLP and SVM. Chapter 4 presents the results obtained and comparison of accuracies obtained with different systems. Chapter 5 consists of final conclusion and possible work that could be done in the future.

CHAPTER 2

LITERATURE SURVEY

The basic idea behind the literature survey is to study the various researches done worldwide in the field of online handwriting character recognition. It provides information about the various techniques followed by the authors upon various languages and their respective accuracy rates. This information further helps to analyze the best algorithms for online character recognition. The literature survey include both online and offline systems and focused on Indian scripts. Earlier, keyboards were the most effective method for interaction with the computers. With the advent of touch screen devices, entirely new way of communication with devices has been explored. Natural handwriting is one of such ways [1].

Almuallim H et al. [20] proposed structural recognition method for cursive Arabic handwritten words. Proposed recognition method was applied on the dataset of 400 words obtained from two writers in which words are segmented into strokes. Classification of these strokes is based upon their geometrical and topological properties. Technique used by author for recognition is elastic matching and authors obtained good accuracy rate. Drawback of structural classification model like elastic matching is that they are slow process.

Lehal GS et al. [21] presented a system of recognition for machine printed Gurmukhi script. Recognition was done at sub character level in which words are segmented into sub-characters which are then classified. Authors employed hybrid classification model consisting of binary decision trees and nearest neighbor technique for classification. Although authors obtained good accuracy but offline methods are time consuming and are not user adaptable due to lack of real time interactivity.

Aparna K et al. [22] presented online recognition system for handwritten Tamil characters. Authors followed stroke based approach in which characters are formed using sequence of strokes. The technique followed is finite state automation for recognizing strokes. Authors used shape or structure based features which represents the stroke for recognition procedure in which they compared the unknown stroke with the database of strokes with the help of flexible string matching technique. Finite state automation determines the termination of the character. By identifying the component strokes, full character can be recognized. As authors have considered the shape based features due to which accuracy rate was low because while writing characters shape can vary due to varying writing styles.

Joshi N et al. [23] worked upon various elastic matching schemes for online recognition of writer dependant isolated handwritten Tamil characters. Three types of features are considered namely, preprocessed x-y coordinates, dominant point coordinates and quantized slope values. Authors have compared seven elastic matching techniques based on these features and time warping distance measure for accuracy, speed and number of training templates. Authors obtained good accuracy rate using these schemes.

Jayaraman A et al. [24] addresses issues in online handwriting recognition for Indian script, Telugu. Telugu script consists of large dataset of 5000 characters which are written as a sequence of strokes. 253 unique strokes exist for this large dataset. Due to the high similarity among several strokes, authors proposed modular approach for recognition. Depending on the relative position of the stroke in the character, authors have divided the dataset of stroke into baseline strokes, top and bottom strokes. SVM classifiers are used for different sets of strokes separated using baselines. On the basis of comparative study on HMM had shown that SVM based classifiers gave significantly better performance.

Bhattacharya U et al. [25] presented novel direction code based approach for online handwritten bangla basic characters. Recognition is performed using multilayer percepton. Proposed classification model is performed on the database of 7043 samples and represented a 50-class recognition problem.

Sharma A et al. [17] proposed online recognition for Gurmukhi handwritten characters. Technique used for recognition is elastic matching. Authors have introduced a process which recognizes characters in two steps. Recognition of strokes is performed in the first stage and then characters are evaluated on basis of recognized strokes in the second stage. Before the recognition, authors have performed preprocessing and also extracted features. Authors have extracted low level features which include linearity, curliness, width, height, aspect ratio, slope, area etc and high level features which include loops, crossings, straight line and dot. Dataset was obtained from 60 writers and consists of 41 Gurmukhi characters. Authors have also discussed simple way of storing strokes and characters. Character recognition is the benchmark of the proposed system irrespective of stroke recognition. Authors have also provided their developed algorithm for recognition purpose in C++.

Sharma A et al. [16] proposed method to recognize the handwritten Gurmukhi words. Authors have introduced new step in which words are recognized using rearrangement of recognized strokes. This rearrangement procedure includes three steps namely, strokes identification as dependant and major dependant strokes, the rearrangement of strokes with respect to their positions and the combination of strokes to recognize word. Dataset of 2576 Gurmukhi words is used for recognition. Authors have focused upon testing whether some unknown word shape can be recognized correctly or not. Exact word recognition is the major bench marking of this work irrespective of the number of characters involved in the word recognition. Authors have developed application in C++ for implementing the recognition procedure.

Mondal T et al. [26] presented recognition results of four popular Indian scripts namely, Bangla, Devanagiri, Tamil and Telugu. Authors compared the two existing feature extraction methods that are point-float and direction-code histogram. Recognition is performed using three types of classifiers namely, Nearest Neighbor (KNN), Multilayer Perceptron (MLP) and Hidden Markov Model (HMM) in order to evaluate the effectiveness of classification models. It has been observed experimentally that direction-code features are performing better than the point-float features and also NN classifier has provided better recognition accuracies. This work provides the benchmark for the future research work on online handwriting recognition for these scripts.

Siddharth KS et al. [14] worked upon offline recognition of Gurmukhi script. Features extracted includes projection histograms (horizontal, vertical and both diagonal), zonal density, distance profiles (from left, right, top and bottom sides), background directional distribution (BDD) features. Recognition is done using three classifiers: KNN, SVM and PNN. Authors aimed to compare the performance when different combinations of features are used with different classifiers. SVM gave highest accuracy rates for the discussed work. Recognition rates can fall due to confusing nature of some characters because of their similarity. For this purpose, some more relevant features can be added to improve the accuracy. Advantages of these statistical classification models are they model the temporal relationship well and classification time is fast in comparison to structural models like elastic matching etc.

Wadhwa D et al. [27] presented online handwritten recognition system for Hindi numeral using Support Vector Machines (SVM). This paper presents the preprocessing, feature extraction and classification phases. Direction angle, curvature along with x-y coordinates are the features extracted for recognition. Performance comparison was done for four kernel schemes in SVM. Accuracies can be improved by adding more features, new preprocessing algorithms and also by increasing the dataset.

Singh G et al. [28] proposed technique for offline recognition on basis of Multilayer Perceptron Model (MLP). Recognition was performed for isolated Gurmukhi characters. MLP has advantage that it uses generalized delta learning rules and can be easily trained with lesser number of iterations.

Singh G et al. [10] aimed to elaborate the online recognition system for Gurmukhi script. Authors explained the different phases involved in the online process namely, data collection, preprocessing, feature extraction and classification. Preprocessing algorithms like normalization, centering of stokes, identification of missing points, smoothing and resampling of strokes are discussed. Segmentation process and feature extraction stage with issues like identification of headline, zones etc have been discussed. Authors also discussed pros and cons of rule based, structural and statistical methods like TDNN, HMM and SVM. To improve the performance hybrid of classifiers like TDNN, HMM and SVM can be used. Authors also concluded that temporal information can increase the recognition accuracies.

Verma K et al. [1] presented effective online handwritten recognition system for Gurmukhi script. Authors studied seventy-two different combinations of HMM- and SVM- classifiers using five different features sets. Feature sets used include normalized x-y traces, region based features, curvature features, curvature feature based classes, directional features. For this purpose, 72 stroke classes have been implemented for character recognition. Recognition was done for the dataset of 1750 Gurmukhi characters obtained from 10 writers. Drawback is that due to some confusing characters recognition rate drops which can be improved by adding more robust zone detection algorithm.

Table 2.1 represents the various techniques adopted for online recognition system for various Indian scripts and their respective accuracies obtained by various researchers.

Table 2.1: Recognition techniques and their respective accuracies for different Indian scripts.

Language	**Author**	**System**	**Features**	**Recognition method**	**Accuracy**
Arabic	Almuallim H et al. [20]	Offline	Geometrical and Topological features	Elastic Matching	91%
Gurmukhi	Lehal GS et al. [21]	Offline	Junctions, loop, headline, sidebar, endpoints, horizontal projection count, profile depth, direction code, aspect ratio.	Hybrid classification using binary decision trees and nearest neighbor technique	96%
Tamil	Aparna K et al. [22]	Online	Shape features (dot, cusp, line terminals, bumps)	Finite state automation	82%
Tamil	Joshi N et al. [23]	Online	Preprocessed x-y coordinates, quantized slope values, dominant point coordinates	Elastic matching technique, DTW.	94.8%
Telugu	Jayaraman A et al. [24]	Online	Preprocessed x-y coordinates, baseline detection	SVM based classifiers	83%
Bangla	Bhattacharya U et al. [25]	Online	Direction code based features	Multilayer perceptron	83.6%
Gurmukhi	Sharma A et al. [17]	Online	Low level features(linearity, curliness, width, height etc) and High level features(loops, crossings, straight line, headline, dot)	Elastic matching	90.08%

Gurmukhi	Sharma A et al. [16]	Online	Low level and high level features	Elastic matching	81.02%
Bangla, Devanagiri, Tamil, Telugu	Mondal T et al. [26]	Online	Point-float and direction-code histogram features	KNN, MLP, HMM	97.78%
Gurmukhi	Siddharth KS et al. [14]	Offline	Projection histograms, zonal density, distance profiles, background directional distribution (BDD) features.	SVM, KNN, PNN	95.04%
Hindi	Wadhwa D et al. [27]	Online	Direction angle and curvature features	SVM	98.9%
Gurmukhi	Singh G et al. [28]	Offline	Spatial information in terms of image coordinates	MLP	98.96%
Gurmukhi	Verma K et al. [1]	Online	Normalized x-y traces, region based features, curvature features, curvature feature based classes, directional features	SVM, HMM	96.7%

2.1 Summary of Literature survey

From the literature survey, it has been observed that online recognition system is better than offline system because of its advantages such as real time interactivity, user adaptability, easy preprocessing and segmentation. Preprocessed data provides better recognition accuracies because this step removes noise present in the handwriting. Section 1.3 outlines the noise which can occur in the data. Feature extraction is very important phase in recognition procedure because it identifies different meaningful patterns. Features are categorized into low level and high level features. Classification phase involves structural, rule based methods and statistical classification models. It has been observed that performance of statistical classification models is better in comparison to structural and rule based methods because they model the temporal relation and also classification time is faster [10].

Good amount of work has been done for offline handwritten character recognition and good accuracies have been obtained [14,21,28]. But very less amount of work has been done for online recognition of Gurmukhi script. Therefore, this thesis is aimed towards online handwritten Gurmukhi character recognition. The approach followed is to obtain good accuracy with lesser number of features and in less time. By using lesser number of features, comparable accuracies have been obtained in the work with respect to Verma K et al. [1] work. Also, better accuracy have been obtained in comparison to Sharma A et al. [16,17].

For the present thesis work, both low level (spatial features) and high level (spectral features) features are extracted. Three different feature sets are used in different combinations and recognized using three different classifiers, namely KNN, MLP and SVM. Finally, performance comparison is made to achieve the highest accuracy. From literature review it has been concluded that these statistical models provides good accuracy in character recognition and are also faster than structural or rule based models.

CHAPTER 3

MATERIALS AND METHODOLOGY

Data acquisition, preprocessing, feature extraction and classification are important phases of online handwriting recognition process. The following sections elaborate on these phases used for the thesis work. Section 3.1 explains about collection of online handwritten data, Section 3.2 explains the various stages of preprocessing algorithms, Section 3.3 explains about the algorithms used to extract important features and Section 3.4 explains the methods used to obtain accuracies.

3.1 Data Acquisition

Handwritten data is captured using devices like digitizing tablets, Personal Data Assistants (PDAs), Tablet PCs etc. With the help of digital pen or stylus, writer is able to write the handwriting samples on the writing pad. Figure 3.1 shows some of the data capturing devices. Handwritten data used for online recognition process in the form of (x,y) coordinate points captured in a time sequential manner by pen or stylus tip movement. The set of these coordinate points make a stroke. Stroke can also be defined as collection of sequence of coordinate points between Pen Down and Pen Up events. Pen trace is sampled at constant rate due to which coordinate points are uniformly distributed in time and not in space. Slow handwriting produces densely located data points in space whereas fast handwriting produces sparsely located data points. Slow handwriting results on sharp corners, in beginning and at the end of stroke, also depends upon the mood of writer [10,29].

(a) Digitizing tablet (b) PDA (c) Digimemo

Figure 3.1: Data capturing devices for Online Recognition.

For the true pen trace, sampling rate and resolution should be high. If in case sampling rate is very low due to which odd corners will be added to the pen trace. True corners and extremely small features will be missed. Figure 3.2 shows the formation of sample stroke for Gurmukhi letter erha. Stroke can be explained as list of recorded (x,y) coordinate points which are in time sequential manner starting from Pen Down event and ending at Pen Up event. As it can be seen in the figure that (X_0, Y_0) and (X_n, Y_n) are the initial and final position of the pen trace [3].

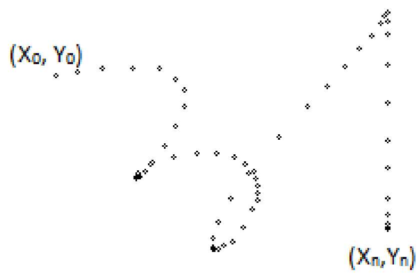


Figure 3.2: Sample Stroke for letter erha(ਅ) in Gurmukhi script [3].

In Gurmukhi script, characters can be uni-stroke or can formed with the combination of more than one stroke. Figure 3.3 represents Gurmukhi character gugga formed with the combination of three strokes.

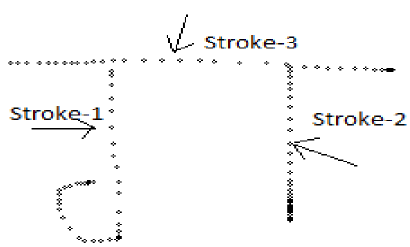


Figure 3.3: Letter gugga(ਗ) formed with the combination of 3 strokes in Gurmukhi script [3].

As the characters in Gurmukhi script are formed with the combinations of different strokes. Therefore, stroke recognition forms the important part in the process. It is required that stroke classifier is trained to recognize the strokes. For this work, samples from different writers have been collected using tablet PC. Strokes collected are stored in the text file as shown in Figure 3.4. For each stroke, x-y traces obtained using digital pen on the writing pad between successive pen-down and pen-up events have been recorded. For the thesis work, 32 stroke classes have been considered. Text file consists of various samples of each stroke, stroke ID, (x,y) coordinate sequence of strokes. Table 3.1 contains the strokes, stroke IDs and the number of samples associated with each stroke.

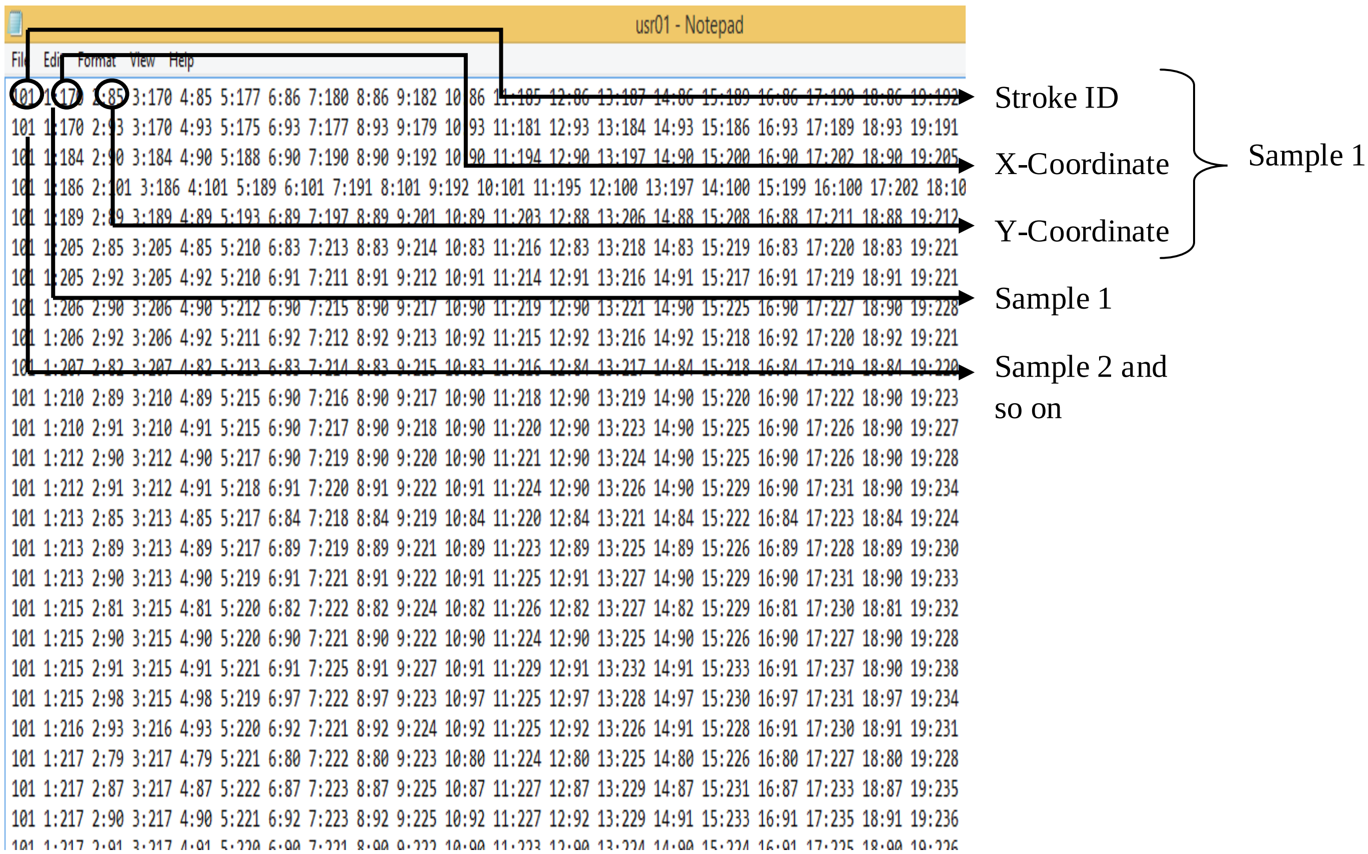
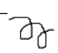
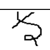
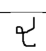
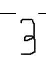


Figure 3.4: Text file in which various samples of stroke are stored, stroke ID represents the stroke class to which sample belongs.

Table 3.1: Gurmukhi strokes used for recognition.

Stroke	Stroke ID	Sample Count
—	121	62
	141	62
	145	68
।	162	65

Stroke	Stroke ID	Sample Count
	146	64
	147	51
	149	62
	151	65
	155	59
	157	70
	161	60
	164	60
	167	65
	168	61
	170	60
	172	64
	174	62
	176	70
	179	61
	191	60
	183	65
	186	60
	187	70
	193	60
	196	70
	200	70
	202	60
	204	67
	207	60
	198	60
	211	64
	212	62

3.2 Preprocessing

Preprocessing phase in online recognition system is performed to remove noise, distortions or unwanted information in the database due to hardware or software limitations so as to produce smooth handwriting. Unwanted information can be in form of irregular size of the stroke, jitter in the text, missing points in the coordinates while capturing the pen movements which depends upon the speed of handwriting, right or left bends, accidental pen lifts, pen stationary or moving slowly give rise to redundancy, unequal distances between adjacent points . To remove such kind of imperfections from the database five preprocessing steps are performed which is removal of duplicate points, size normalization, centering, interpolation of the missing points, resampling of the points. After the stroke has been captured, it is preprocessed to extract the features from the x-y traces [2,10]. Figure 3.5 represents the five preprocessing algorithms used in online recognition system in form of flowchart.

Preprocessing algorithms

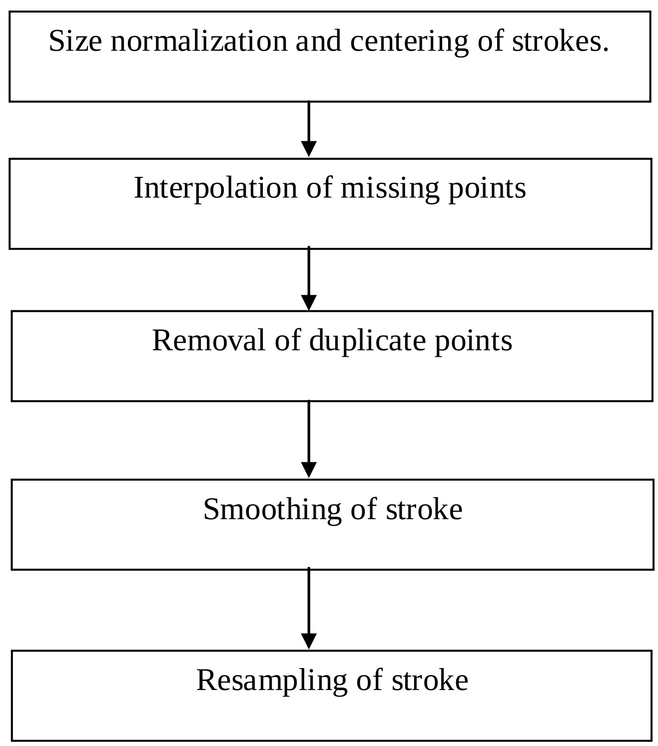


Figure 3.5: Flowchart representing preprocessing algorithms performed in online recognition system.

3.2.1 Size Normalization and Centering of Strokes

Input stroke size depends on the writer, movement of the pen and the writing pad. Also, stroke is not centered when pen is moved along the boundary of writing pad. Therefore, it is necessary to normalize and center the stroke to recognize it. With the help of size normalization, every stroke normalizes to same size and also centering to constant frame with text placed at fixed distance from origin [2].

To achieve size normalization and centering of strokes, comparison is done between input text border frame and an already assumed fixed frame along with moving input text to assumed location for centering. For the thesis work, size of window is 300x300 pixels. Stroke less than or greater than this window size is transformed to fixed size of 300x300 pixels [3].

3.2.2 Interpolation of Missing Points

Stroke captured with high speed will have missing points. Hardware limitations can also result to missing points. Therefore, interpolation of missing points is necessary to achieve accuracy during recognition. Various techniques can be used for interpolation like Bezier and B-Spline curve interpolation. In the work, piecewise Bezier interpolation is used in which set of four consecutive points is used to obtain Bezier curve and next set gives next Bezier curve. In the piecewise Bezier interpolation, interpolation is done where distance between consecutive points is greater than one. This algorithm first computes the distance between the consecutive points and if the distance is greater than one, Bezier curve is drawn [2,3].

3.2.3 Removal of Duplicate Points

Duplicate points are aggregation of excessive points in some part of the stroke while writing. So, it is required to remove these excessive points from the stroke to achieve equally spaced resampling points. In this algorithm, only those points are taken who have distance between them greater than some threshold value [3,30].

3.2.4 Smoothing of Stroke

Flickers are introduced into the handwriting due to individual writing style and also due to hardware. Therefore, smoothing is required for recognition. This is done by considering five points in the list. If in case three points are considered, then it will not affect the stroke and if more than five points are taken, it will result into sharp edge [3,5].

3.2.5 Resampling of stroke

Resampling is performed to fix the number of points in the stroke and at the same time preserve the original shape of the stroke. It has been observed from the literature survey that best results are obtained when the number of resampled points is 64. Slow handwriting results into more number of points in the stroke and fast handwriting introduces less point in the stroke. In order to remove such variations, resampling is done to equalize the distance between consecutive points. Resampling is very useful in noise removal and data reduction [2,5].

In this algorithm, all points should be at distance of one from its neighboring points and fixed numbers of points are selected. Removal of points can be performed in two ways: remove all the points between pair of points having distance less than one or remove points at constant distances. After this phase, stroke having fixed number of points which in this case is 64 is obtained and also these points are equidistant from each other [3].

After implementation of preprocessing steps, 64 preprocessed x-y traces of the stroke are obtained in 300x300 window size. Preprocessed data is stored in text file. Each text file consists of various samples of single stroke from different writers. As for the work, 32 stroke classes have been considered. 32 text files which contain preprocessed samples are further used for feature extraction stage. Figure 3.6 shows the one of the text file in which samples of stroke are stored.

```
M_121_125 - Notepad
File  Edit  Format  View  Help
121 1:0 2:148 3:3 4:148 5:6 6:148 7:9 8:148 9:11 10:148 11:17 12:151 13:21 14:151 15:24 16:151 17:27 18:151 19:31 20:151 21:31 22:149 23:34 24:148 25:38
121 1:0 2:150 3:4 4:150 5:7 6:150 7:10 8:150 9:13 10:150 11:17 12:150 13:23 14:150 15:26 16:150 17:31 18:150 19:35 20:150 21:39 22:150 23:45 24:150 25:49
121 1:0 2:150 3:4 4:147 5:7 6:148 7:10 8:146 9:14 10:146 11:19 12:146 13:23 14:146 15:27 16:146 17:32 18:146 19:37 20:146 21:44 22:146 23:53 24:146 25:58
121 1:0 2:154 3:5 4:154 5:8 6:154 7:12 8:154 9:15 10:154 11:19 12:154 13:25 14:154 15:28 16:154 17:32 18:154 19:37 20:154 21:41 22:154 23:44 24:154 25:48
121 1:0 2:143 3:3 4:143 5:8 6:143 7:14 8:143 9:18 10:143 11:24 12:143 13:28 14:143 15:32 16:143 17:36 18:143 19:40 20:143 21:44 22:143 23:49 24:143 25:53
121 1:0 2:147 3:4 4:147 5:7 6:147 7:11 8:147 9:16 10:147 11:20 12:147 13:25 14:147 15:30 16:147 17:33 18:147 19:38 20:147 21:43 22:147 23:48 24:147 25:51
121 1:0 2:155 3:6 4:152 5:10 6:151 7:15 8:150 9:18 10:149 11:22 12:148 13:25 14:148 15:28 16:146 17:31 18:145 19:36 20:145 21:40 22:145 23:43 24:145 25:4
121 1:0 2:145 3:3 4:145 5:7 6:146 7:13 8:147 9:16 10:147 11:21 12:150 13:25 14:150 15:30 16:150 17:35 18:150 19:40 20:150 21:45 22:150 23:49 24:150 25:53
121 1:0 2:147 3:4 4:147 5:6 6:147 7:10 8:150 9:16 10:150 11:21 12:150 13:23 14:150 15:29 16:150 17:31 18:149 19:34 20:149 21:37 22:149 23:39 24:149 25:42
121 1:0 2:145 3:5 4:145 5:9 6:145 7:13 8:145 9:17 10:145 11:20 12:145 13:24 14:145 15:28 16:145 17:33 18:145 19:37 20:145 21:41 22:145 23:46 24:145 25:51
121 1:0 2:147 3:3 4:147 5:7 6:147 7:12 8:147 9:16 10:147 11:21 12:147 13:26 14:147 15:29 16:149 17:35 18:152 19:38 20:152 21:43 22:152 23:47 24:152 25:52
121 1:0 2:140 3:5 4:140 5:9 6:140 7:13 8:140 9:16 10:140 11:19 12:140 13:22 14:140 15:26 16:140 17:31 18:140 19:35 20:140 21:39 22:140 23:44 24:140 25:48
121 1:0 2:150 3:4 4:150 5:10 6:150 7:13 8:151 9:18 10:151 11:21 12:152 13:25 14:152 15:28 16:153 17:32 18:153 19:37 20:153 21:41 22:153 23:46 24:156 25:5
121 1:0 2:147 3:4 4:147 5:10 6:147 7:13 8:148 9:18 10:148 11:21 12:149 13:25 14:149 15:28 16:149 17:31 18:149 19:36 20:149 21:41 22:149 23:44 24:149 25:4
121 1:0 2:150 3:5 4:150 5:8 6:150 7:12 8:150 9:15 10:150 11:19 12:150 13:24 14:150 15:28 16:150 17:33 18:150 19:38 20:150 21:38 22:155 23:43 24:155 25:48
121 1:0 2:150 3:4 4:150 5:7 6:150 7:11 8:150 9:16 10:150 11:20 12:150 13:24 14:150 15:29 16:150 17:32 18:150 19:36 20:150 21:40 22:150 23:45 24:150 25:48
121 1:0 2:156 3:3 4:155 5:9 6:153 7:13 8:153 9:17 10:152 11:21 12:152 13:24 14:152 15:27 16:152 17:32 18:152 19:37 20:152 21:40 22:152 23:45 24:152 25:47
121 1:0 2:153 3:4 4:153 5:9 6:153 7:12 8:153 9:17 10:153 11:21 12:153 13:25 14:153 15:30 16:153 17:34 18:153 19:38 20:153 21:43 22:153 23:46 24:153 25:50
121 1:0 2:148 3:2 4:148 5:5 6:148 7:9 8:148 9:12 10:148 11:14 12:148 13:18 14:148 15:20 16:148 17:24 18:148 19:25 20:147 21:29 22:147 23:33 24:147 25:36
121 1:0 2:156 3:4 4:154 5:8 6:153 7:12 8:153 9:18 10:153 11:21 12:152 13:24 14:151 15:31 16:151 17:34 18:151 19:38 20:150 21:42 22:150 23:46 24:150 25:50
121 1:0 2:146 3:3 4:146 5:5 6:146 7:10 8:148 9:14 10:148 11:19 12:148 13:23 14:148 15:26 16:148 17:31 18:148 19:35 20:148 21:40 22:148 23:44 24:148 25:48
121 1:0 2:152 3:6 4:152 5:10 6:152 7:15 8:152 9:18 10:152 11:22 12:152 13:25 14:152 15:28 16:152 17:33 18:152 19:38 20:152 21:43 22:152 23:46 24:152 25:5
121 1:0 2:157 3:3 4:156 5:8 6:155 7:11 8:155 9:16 10:154 11:19 12:154 13:22 14:154 15:26 16:154 17:29 18:154 19:35 20:155 21:40 22:153 23:43 24:154 25:47
121 1:0 2:144 3:3 4:144 5:8 6:144 7:13 8:144 9:16 10:144 11:22 12:144 13:25 14:144 15:30 16:144 17:33 18:144 19:36 20:144 21:40 22:144 23:45 24:144 25:49
121 1:0 2:147 3:5 4:147 5:8 6:148 7:12 8:148 9:15 10:149 11:19 12:150 13:25 14:150 15:28 16:150 17:33 18:153 19:38 20:153 21:41 22:153 23:46 24:153 25:51
```

Figure 3.6: Text file representing preprocessed samples of one the stroke class considered.

3.3 Feature Extraction

Feature extraction phase in online handwriting recognition is very important as recognition accuracy largely depends on this step. After preprocessing phase, features are extracted to obtain three different representations of the stroke for stroke classification. Features are categorized into low level and high level features. From the literature survey, it has been observed that different types of features are extracted by researchers which include projection histograms, zonal density, distance profiles, background directional distribution (BDD) features, curvature based features, region based features etc. In the present work, spatiotemporal, spectral and tangential features are extracted. Spatiotemporal and tangential features come under the category of low level features whereas spectral features are high level features [31]. For the thesis work, these three feature sets are used in different combinations for recognition. Features are extracted using MATLAB.

3.3.1 Spatiotemporal Features (P_{xy})

A stroke obtained in data acquisition stage is a sequence of data points, where consecutive data points have uniform temporal separation. The preprocessed stroke consists of sequence of fixed number of data points that have uniform spatial separation along its trajectory. Representation of preprocessed stroke is shown in equation (3.1):

$$H_s = [h_{s1}\ h_{s2} \ldots\ldots\ldots\ldots\ h_{sn}]^T \tag{3.1}$$

where H_s represents preprocessed stroke, h_{si} =(x_{si},y_{si}) represents positional coordinates of i^{th} data point and n represents number of data points in stroke. For the present work, the value of n is 64. The value of n can also be variable but in the thesis value of n is fixed. Spatiotemporal features captures the information about the temporal sequence from positional coordinates of the data points. The temporal information of the stroke gets retained in the preprocessed stroke. These features are low level and point oriented features and can be called as local features [1,32].

For the work, 32 stroke classes are considered and information about the number of samples considered for each class has been given in the section 3.1 of this chapter.

3.3.2 Spectral Features (S_{xy})

Spectral features come under the category of global long range features. Spectral features can also be called as high level features. Global features when used with local features represent the combination of high and low level features. Spectral features define the stroke in frequency domain. Fast Fourier Transform (FFT) is used to convert stroke from time domain to frequency domain. FFT representation of the stroke is provided in equation (3.2):

$$h_w = [\ h_{w_{x+iy}}\ ]^T \tag{3.2}$$

where $h_{w_{x+iy}}$ = FFT($\bar{x}$ + i$\bar{y}$), $\bar{x}$ + i$\bar{y}$ = [x_{s1} + y_{s1} x_{s2}+ y_{s2}………….. x_{sn}+ y_{sn} $]^T$. Spectra which have high frequency content implies rapid change in the x or y coordinate because high frequencies indicate rapid change in the signal. Spectra containing little high frequency content indicate little change in the x or y coordinates. Low frequency contents provide general shape properties and high frequency content captures the finer details of the object.

Fourier Descriptors (FDs) have the property of invariance to affine shift in the starting point and to affine shape transformations like translation, rotation, scaling and mirror reflections. Only a small number of Fourier descriptor coefficients are enough to represent the handwritten character and FDs allow complete reconstruction of the original shape of the character.

Feature extraction involves fourier transformation of the (x,y) coordinate vector obtained for each sample stroke. The Fast Fourier Transform (FFT), as shown in (3.3), is used for creating feature vector for each sample stroke.

$$X_k = \sum_{n=0}^{N-1} x_n\, e^{-i2\pi k\frac{n}{N}} \qquad k = 0,\ldots\ldots,N-1. \tag{3.3}$$

We have the boundary points for each sample stroke in form 64 (x,y) coordinates obtained after data acquisition and preprocessing stage. The k^{th} coordinate in the sample is represented by position (x_k,y_k). So, we can describe the sample as parametric equations:

$$x(k) = x_k \tag{3.4}$$

$$y(k) = y_k \tag{3.5}$$

Taking Fourier transform of both the equations (3.4) and (3.5), we get two frequency spectra:

$$z_x(v) = \mathcal{F}\,(x(k)) \tag{3.6}$$

$$z_y(v) = \mathcal{F}\,(y(k)) \tag{3.7}$$

These two frequency spectra are called Fourier descriptors. Considering (x,y) coordinates in the complex plane we get:

$$t(k) = x(k) + iy(k) \tag{3.8}$$

Now we get single fourier descriptor by taking fourier transform of complex function t(k).

$$z(v) = \mathcal{F}(t(k)) \tag{3.9}$$

FFT feature vector was obtained for every sample of the different stokes considered [33-35]. Each FFT feature vector consists of 64 complex coefficients which contains the frequency information of the signal. Feature vectors obtained using FFT for the strokes are further used for classification.

3.3.3 Tangential features (T_{xy})

Tangential features are represented in Figure 3.7. Tangential features can be represented as follows:

$$\alpha_{p_{n-1},p_n} = \arctan\left(\frac{y_{n-1}-y_n}{x_{n-1}-x_n}\right) \tag{3.10}$$

and the feature vector for tangential features is represented as follows:

$$F = [\alpha_{p_1,p_2}, \alpha_{p_2,p_3}, \ldots\ldots \alpha_{p_{n-1},p_n}] \tag{3.11}$$

Direction of text matters in tangential features which mean if the text is written from left to right; it will yield totally different features than text which is written in right to left direction [7].

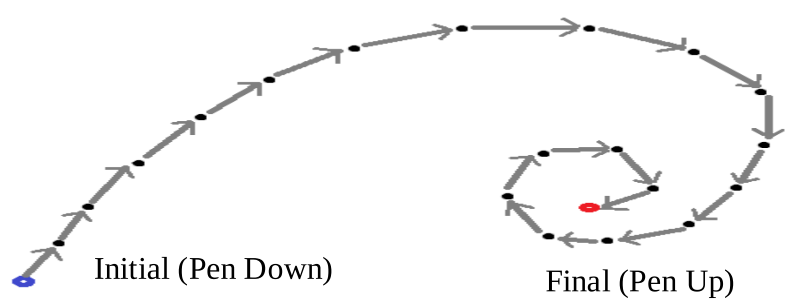


Figure 3.7: An illustration of feature vector; pen-tip position and tangent at every pen tip position along the pen trajectory.

For the thesis work, from the preprocessed samples of every stroke, tangential features are extracted. From 2019 sample vectors, equal number of tangential vectors is obtained and stored in excel file for further recognition procedure. Table 3.2 briefs about the features extracted.

Table 3.2. Features extracted to build classification models.

Features Extracted	**Name**	**Explanation**
Spatiotemporal features	P_{xy}	Preprocessed x-y traces of each stroke (2019 spatiotemporal feature vectors are obtained).
Spectral features	S_{xy}	Preprocessed strokes in frequency domain (2019 spectral feature vectors are obtained)..
Tangential features	T_{xy}	Tangents obtained at every pen tip position along the pen trajectory for all strokes (2019 tangential feature vectors are obtained).

3.4 Stroke Classification

Stroke classification is one of the crucial steps in the online handwriting recognition. From the set of stroke samples, subset is used for training to build models for the strokes. These models can then be used to recognize unknown sample stroke. Before the recognition, strokes are preprocessed from which features are extracted. In this section, three types of classifiers are explained which has been experimented for stroke classification. From the literature survey, it has been inferred that these classifiers have shown good accuracy rates for the online handwritten character recognition process. Therefore, these classifiers are used for obtaining accuracies in the thesis work. Also, three different classifiers are used to make comparison between the accuracies obtained by them.

Classification models are obtained using two methods: cross validation and percentage split. In k-fold cross validation method, the original sample set is randomly partitioned into equal sized k subsample sets. One of the subsample set is kept as validation data for testing the model and k-1 subsample set are used for training purpose. Cross validation technique is repeated k times in such a way that each of the k subsample sets are used exactly once. After that, k results obtained are then averaged to single estimation. On the other side, percentage split method splits the sample data into two parts. Out of the two parts, one part is kept as training data and the other as validation data. User decides the percentage at which data splits and in the same way k in the cross validation technique. Three classification algorithms are used in the thesis work: K-Nearest Neighbor (KNN), Multilayer Perceptron neural network (MLP) and Support Vector Machines (SVM). These classifiers are explained in the next sections.

3.4.1 K-Nearest Neighbor (KNN)

KNN can be classified as lazy or instance based learning in which unknown pattern is related to some known pattern in accordance with some distance or other similarity function. For classification of the sample, it depends on the majority vote of its neighbor. In KNN, k depicts the number of neighbors to be considered and unknown sample is assigned the class of majority of the neighbors. Similarity function which can be used to find the nearest neighbors is Euclidean distance between the testing and reference points in order to find the nearest k neighbors [9,14,26].

KNN algorithm can be well understood using Figure 3.8. Green object is the unknown pattern and it is required to find the class to which it belongs. Classification is performed with k-3 (solid circle) and k=5 (dotted circle). The class of unknown pattern is altered in each case. For k=3, unknown pattern is classified as red triangular shape pattern and for the other case it is classified as blue square pattern.

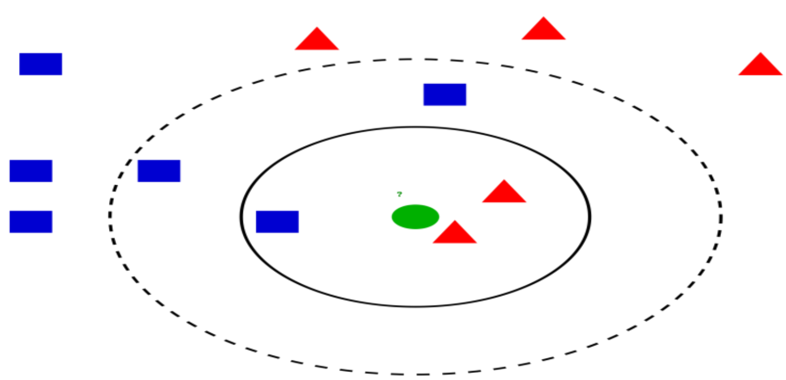


Figure 3.8: Implementation of KNN algorithm.

It calculates classification function by examining the labeled training points as nodes in n dimensional space where n is feature size. In the proposed work, KNN is used as one of the classifier to recognize different classes. For the thesis work, classification using KNN is done at k = 1,3,5,7. Comparison of accuracies obtained at these values has also been done.

3.4.2 Multilayer Perceptron (MLP)

MLP is very popular feed forward artificial neural network architectures used in online handwriting recognition system. It uses supervised learning technique called backpropagation through which network is trained. MLP is composed of three layers: Input layer, hidden layer and output layer and each layer is fully connected to the next layer. Modification to linear perceptron has made multilayer perceptron. It can classify data which are not linearly separable. MLP has property that some of its neurons use non linear activation function like sigmoid. Common issues faced while using MLP are selection of its architecture (number of hidden neurons) and values of different learning parameters such as learning rate and momentum factor of backpropagation network [26,28]. Figure 3.9 represents the architecture of single hidden layer multilayer perceptron.

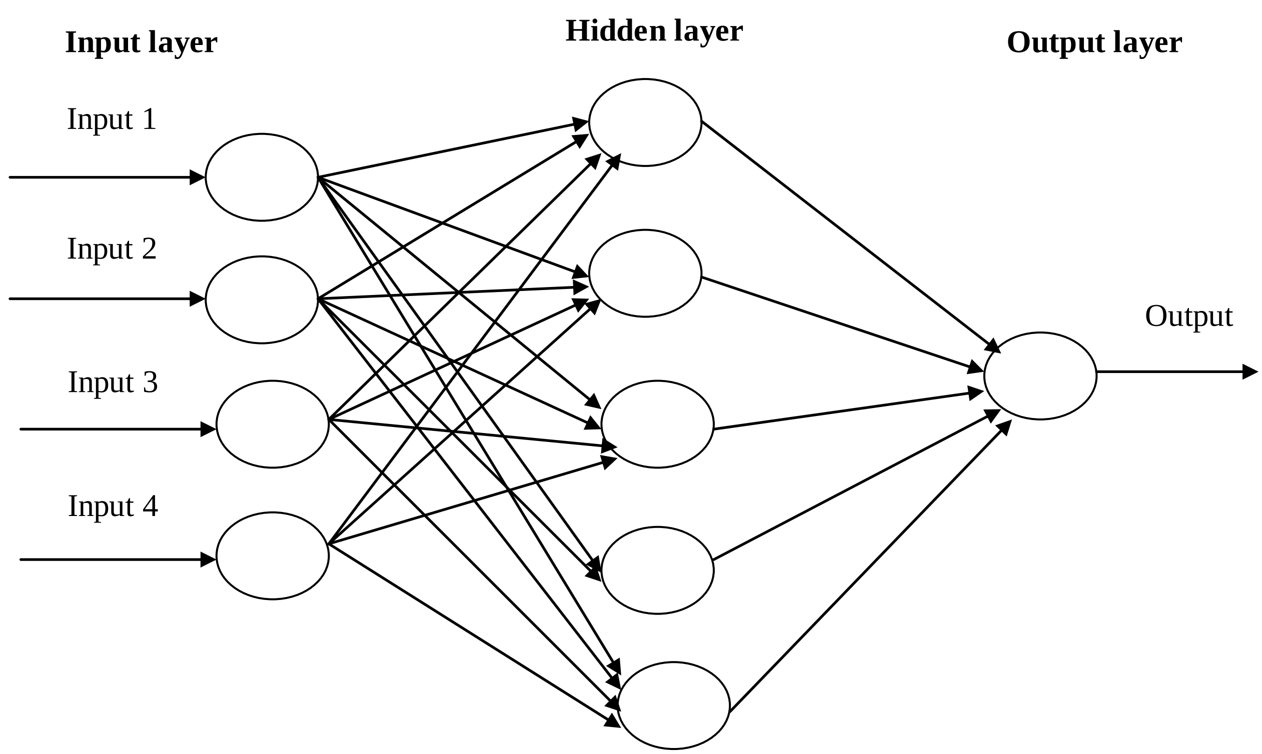


Figure 3.9: Basic architecture of multilayer perceptron.

The activation of the neuron is calculated as weighted sum of the inputs subtracted from the threshold value of the neuron. The network generates its input by passing the activation signal through the activation or transfer function. Input layer is responsible for introducing the value of input variables.

This model is used because of its parallel execution capability which helps in dealing with complexities associated with the Gurmukhi script characters [28]. Also, neural network have high noise tolerance due to which it is preferred [24]. Parameters of MLP are empirically optimized. The number of neurons in the output layer is 32 which are equal to the number of stroke classes. Learning rate of backpropagation algorithm is set at 0.3 which is its default value. Learning rate can be between 0 and 1. Momentum rate of backpropagation algorithm can take values between 0 and 1. Its value is set at 0.2 which is the default value. In this work, four types of feature sets are used to train the MLP. For both of the feature sets P_{xy} and S_{xy}, size of hidden layer is 48 in the architecture. Size of hidden layers is decided using formula: (attributes +classes)/2. For hybrid of P_{xy} and S_{xy}, size of hidden layer is 80. For hybrid of P_{xy} and T_{xy}, size of hidden layer is 80. Optimum value of parameters has been taken.

3.4.3 Support Vector Machines (SVM)

SVM is supervised learning method that can be applied to classification or regression. SVM requires fixed length of feature vector of the stroke, irrespective of instance or class it belongs. It is a binary classifier which takes the input data and classifies it to one of the two distinguished classes by constructing the maximum margin hyper plane. In case classes are not linearly separable, SVM uses kernel functions in order to map data to higher dimensional space to increase the separability. One such kernel function is polynomial kernel and it has been observed that it results in good accuracies for online handwriting recognition system. For the thesis, polynomial kernel function is used. If the problem consists of more than two classes, then algorithm breaks the multiclass into multiple binary classification problems and multiple binary SVM is designed [36-39].

SVM works upon the principle of decision planes which separates between the set of samples according to their class membership. Figure 3.10 represents the linear classification performed by the SVM. This example consists of two linearly separable classes: dots and triangles. The line separating the two classes is defined as decision plane. Unknown sample is classified according to region in which it falls of the decision plane.

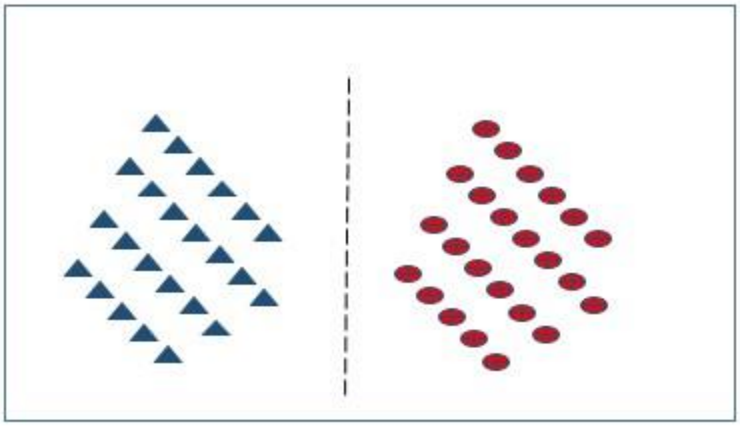


Figure 3.10: Linear classification done using SVM.

Sometimes complex separations are required when classes are not linearly separable which is done by complex functions rather than linear function. Figure 3.11 represents that classes are not linearly separable and they require a curve to distinguish between the classes and how the input space is mapped to feature space. For classes which are not linearly separable, non linear classifiers called as hyperplane classifiers are used by applying kernel function to maximum margin hyperplane. With the help of this algorithm, maximum margin hyperplane fits to the transformed feature space. The transformation from input to feature space is nonlinear and the feature space is high dimensional space. The classifier is the hyperplane in the transformed feature space.

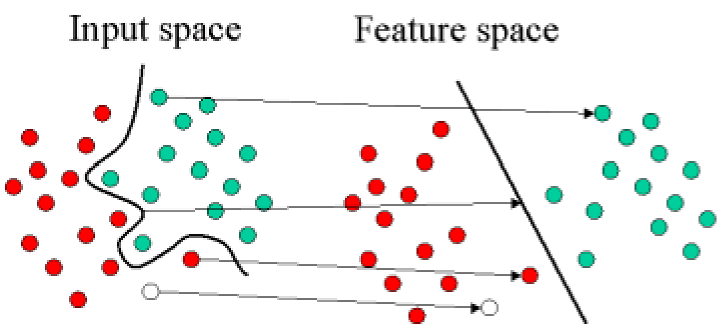


Figure 3.11: Mapping of input space to feature space in SVM classification.

These hyperplane classifiers linearly separate the two classes by finding maximum margin between these classes. The margin is defined as minimum distance of the closest data points from the separating hyperplane. These data points are known as 'Support Vectors' and also illustrated in Figure 3.12 [27].

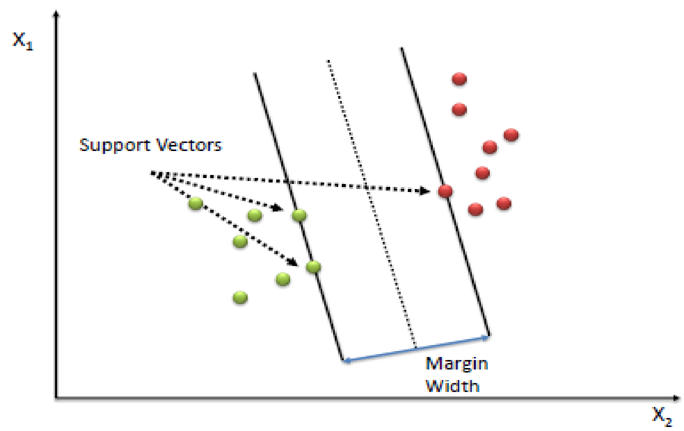


Figure 3.12: Feature space in SVM showing the support vectors and hyperplane classifier.

For the thesis work, SVM is used as one of the classifier for recognition. SVM is chosen for the thesis work because of its speed, high flexibility and scalability [24]. Polykernel is used for the work. Other parameters which are optimized: learning rate (γ) and penalty parameter (C).

CHAPTER 4

RESULTS AND DISCUSSIONS

Recognition using three types of features: Spatiotemporal, Spectral and Tangential features have been performed in this chapter. For the recognition three types of classifiers are used: KNN, MLP, SVM and their parameters are optimized to obtain better accuracies. Two methods are adopted for recognition: k-fold cross validation and percentage split. Cross validation is performed at two values of k = 5 and k = 10. Percentage split is performed at 66% split of dataset. Parameters of KNN, MLP and SVM are given in Table 4.1.

Table 4.1: Parameters of KNN, MLP and SVM.

Classifier	**Features**	**Parameters**	**Options/Values**
K-Nearest Neighbor (KNN)	P_{xy}, S_{xy}, hybrid of P_{xy} and S_{xy}, hybrid of P_{xy} and T_{xy}	K	1,3,5,7
Multilayer Perceptron (MLP)	P_{xy}, S_{xy}	Size of hidden layer (a)	48
	hybrid of P_{xy} and S_{xy}, hybrid of P_{xy} and T_{xy}	Size of hidden layer (a)	80
	P_{xy}, S_{xy}, hybrid of P_{xy} and S_{xy}, hybrid of P_{xy} and T_{xy}	Learning rate (0-1)	0.3
		Momentum (0-1)	0.2
		Neurons in output layer	32
		Training time	500
Support Vector Machines (SVM)	P_{xy}, S_{xy}, hybrid of P_{xy} and S_{xy}, hybrid of P_{xy} and T_{xy}	Penalty parameter (C)	50,100,150
		Kernel	Polykernel
		Tolerance parameter (ε)	0.001

4.1 Classification performance of Spatiotemporal Features (P_{xy}).

Table 4.2 represents the recognition rates for the spatiotemporal features obtained using KNN, MLP and SVM using 5-fold cross validation technique. Table 4.3 represents the recognition rates with 10-fold cross validation technique. Table 4.4 represents recognition rates using percentage split method in which dataset is split at 66%. Parameters optimized are provided in table 4.1. For KNN classifier, accuracy at k = 1, 3, 5 and 7 has been obtained and compared for best accuracy. In case of MLP, learning rate and momentum of backpropagation are set at default value of 0.3 and 0.2 respectively. Also, 48 neurons in hidden layer are used with resulting 32 targets in the output layer. In case of SVM, accuracy for c = 50, 100 and 150 has been obtained and compared. Kernel function used is polynomial function and tolerance parameter is set at default value of 0.001

Table 4.2: Recognition rates with 5-fold cross validation technique.

Classifier		**5-fold Cross Validation**			
		Recognized Instances	**Misclassified Instances**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1792	227	0.013	88.75
	K=3	1791	228	0.006	88.70
	K=5	1781	238	0.003	88.21
	K=7	1768	251	0.003	87.56
Support Vector Machines (SVM)	**C=50**	1805	214	3.38	89.40
	C=100	1806	213	3.47	89.45
	C=150	1805	214	3.92	89.40
Multilayer Perceptron (MLP)		1800	219	288.12	89.15

Table 4.3: Recognition rates with 10-fold cross validation technique.

Classifier		10-fold Cross Validation			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1801	218	0.02	89.20
	K=3	1804	215	0.02	89.35
	K=5	1795	224	0.02	88.90
	K=7	1772	247	0.04	87.76
Multilayer Perceptron (MLP)		1815	204	290.48	89.89
Support Vector Machines (SVM)	**C=50**	1810	209	3.58	89.64
	C=100	1798	221	4.15	89.05
	C=150	1805	214	3.95	89.40

Table 4.4: Recognition rates with percentage split method.

Classifier		Percentage Split (66%)			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	606	80	0	88.33
	K=3	606	80	0	88.33
	K=5	597	89	0	87.02
	K=7	560	96	0	86
Multilayer Perceptron (MLP)		612	74	341.71	89.21
Support Vector Machines (SVM)	**C=50**	613	73	3.48	89.35
	C=100	614	72	3.51	89.50
	C=150	614	72	4.53	89.50

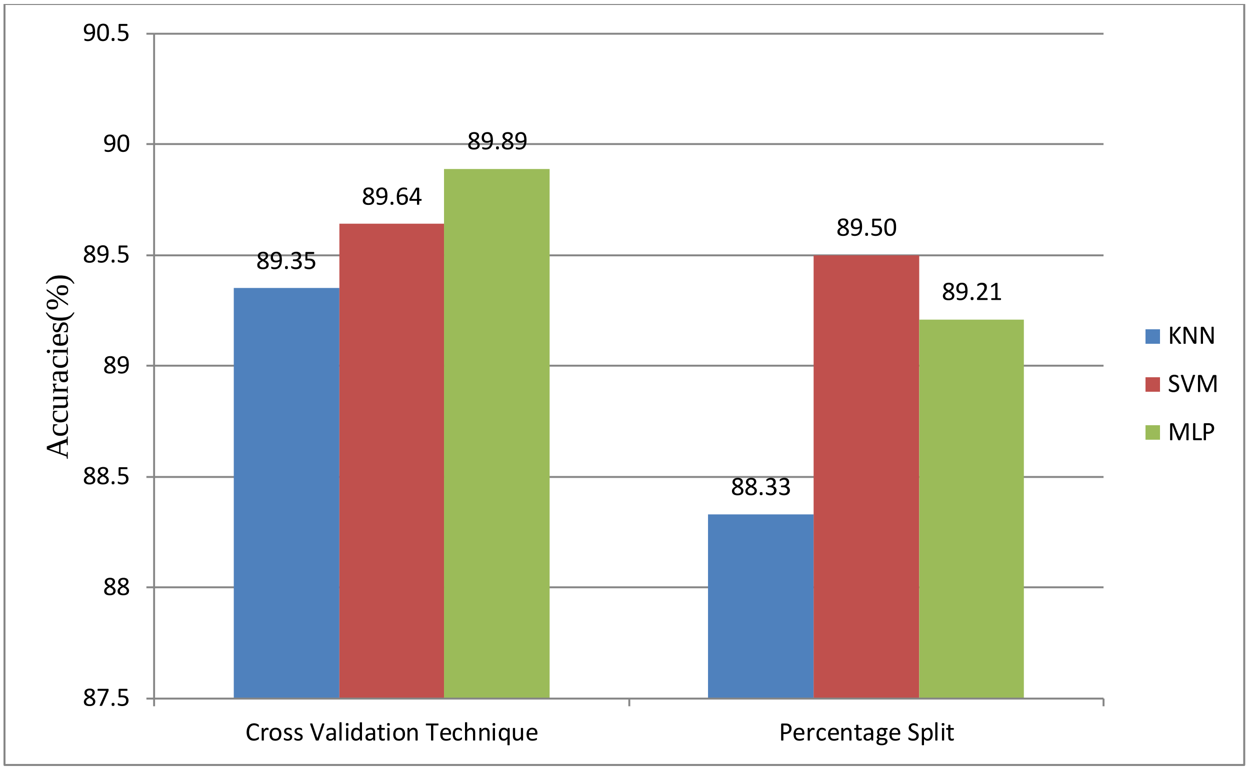


Figure 4.1: Comparison of highest accuracies obtained from cross validation and percentage split technique.

Figure 4.1 depicts the comparison of highest accuracies obtained using cross validation and percentage split for KNN, MLP and SVM. It can be seen from the graph that in cross validation part, MLP provides highest accuracy and in case of percentage split SVM provides the highest accuracy. Also it can be seen the results are better with cross validation technique.

From the above results of recognition for spatiotemporal features, it has been inferred that for KNN with 5-fold cross validation maximum accuracy is achieved for k =1 which is 88.75%, KNN with 10-fold cross validation provides maximum accuracy at k = 3 which is 89.35%, KNN with percentage split method provides maximum accuracy at k = (1,3) which is 88.33%. In case of multilayer perceptron with 10 fold cross validation technique, accuracy is 89.89% and with percentage split method, accuracy is 89.21%. For the case of SVM, maximum accuracy is obtained for c = 100 that is 89.45% with 5-fold cross validation technique, maximum accuracy is obtained for c=100 that is 89.64% with 10-fold cross validation technique, maximum accuracy is obtained for c = (100,150) that is 89.50% with percentage split method. It is observed that good accuracy rates have been obtained for these classifiers using spatiotemporal features and highest accuracy is obtained using multilayer perceptron in 10-fold validation that is 89.89%.

4.2 Classification performance of the Spectral features (S_{xy}).

Table 4.5 depicts the results of recognition rates obtained from the Fourier descriptor features with three different classifiers namely KNN, MLP and SVM using 5-fold cross validation technique. Table 4.6 represents the recognition rates for 10 fold cross validation. Table 4.7 represents the recognition rates for the percentage split method in which dataset is split at 66%. For KNN classifier, accuracy at k = 1, 3, 5 and 7 has been obtained and compared for best accuracy. In case of MLP, learning rate and momentum of backpropagation are set at default value of 0.3 and 0.2 respectively. Also, 48 neurons in hidden layer are used with resulting 32 targets in the output layer. In case of SVM, accuracy for c = 50, 100 and 150 has been obtained and compared. Kernel function used is polynomial function and tolerance parameter is set at default value of 0.001.

Also, comparison among the accuracies obtained using these classifiers with cross validation and percentage split method have been depicted using graphical representation.

Table 4.5: Recognition rates using 5-fold cross validation technique.

Classifier		5-fold Cross Validation			
		Recognized Instances	**Misclassified Instances**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1621	398	0.02	80.28
	K=3	1527	492	0.02	75.63
	K=5	1502	517	0.04	74.39
	K=7	1474	545	0.03	73
Support Vector Machines (SVM)	**C=50**	1704	315	4.41	84.39
	C=100	1700	319	4.53	84.20
	C=150	1694	325	4.65	83.90
Multilayer Perceptron (MLP)		1738	281	295.85	86.08

Table 4.6: Recognition rates with 10-fold cross validation technique.

Classifier		10-fold Cross Validation			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1629	390	0.02	80.68
	K=3	1536	483	0.04	76.07
	K=5	1506	513	0.03	74.59
	K=7	1484	535	0.02	73.50
Multilayer Perceptron (MLP)		1754	265	300.55	86.87
Support Vector Machines (SVM)	**C=50**	1715	304	4.78	84.94
	C=100	1711	308	4.09	84.74
	C=150	1712	307	4.06	84.79

Table 4.7: Recognition rates with percentage split method.

Classifier		Percentage Split (66%)			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	536	150	0.02	78.13
	K=3	510	176	0.03	74.34
	K=5	492	194	0.03	71.72
	K=7	480	206	0.02	69.97
Multilayer Perceptron (MLP)		585	101	303.63	85.27
Support Vector Machines (SVM)	**C=50**	576	110	4.49	83.96
	C=100	572	114	4.11	83.38
	C=150	572	114	4.01	83.38

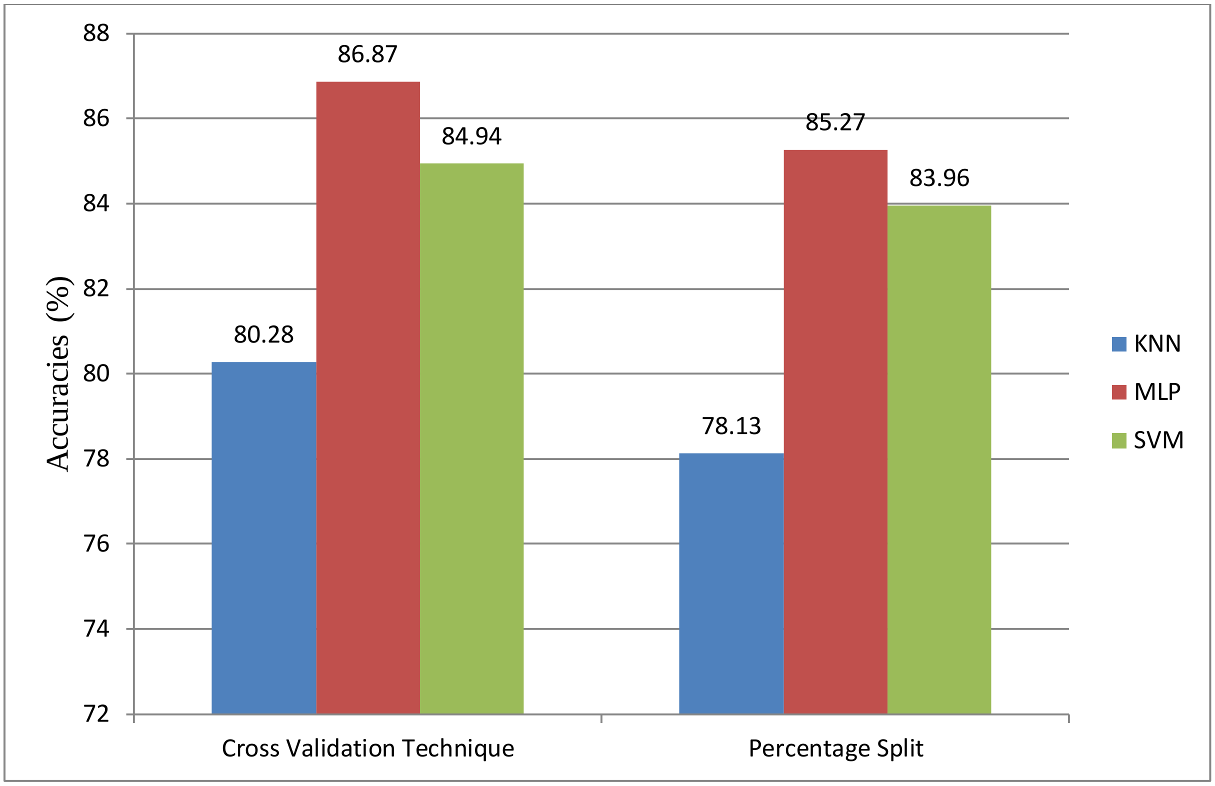


Figure 4.2: Comparison of highest accuracies for cross validation and percentage split method.

Figure 4.2 depicts graphical representation of accuracies obtained by different classifiers using cross validation and percentage split method. It can be clearly seen in the graph that cross validation results is better than the percentage split method. Highest accuracy is obtained with MLP using cross validation method.

From the above results of recognition for spectral features, it has been observed that for KNN with 5-fold cross validation maximum accuracy is achieved for k =1 which is 80.28%, KNN with 10-fold cross validation provides maximum accuracy at k = 1 which is 80.68%, KNN with percentage split method provides maximum accuracy at k = 1 which is 78.13%. In case of multilayer perceptron with 10 fold cross validation technique, accuracy is 86.87% and with percentage split method, accuracy is 85.27%. For the case of SVM, maximum accuracy is obtained for c = 50 that is 84.39% with 5-fold cross validation technique, maximum accuracy is obtained for c =50 that is 84.94% with 10-fold cross validation technique, maximum accuracy is obtained for c = 50 that is 83.96% with percentage split method. It is observed that average accuracy rates have been obtained for these classifiers using spectral features and highest accuracy is obtained using multilayer perceptron in 10-fold validation that is 86.87%.

4.3 Performance comparison using Hybrid of Spatiotemporal (P_{xy}) and Tangential (T_{xy}) features.

Table 4.8 depicts the accuracies obtained with different classifiers when Spatiotemporal and Tangential features are combined for classification. Classification is performed using two techniques: cross validation and percentage split method. In the above mentioned table, 5-fold cross validation technique is used. Table 4.9 depicts the results with 10-fold cross validation technique. Table 4.10 depicts the results with percentage split method in which dataset is split at 66%. For KNN classifier, accuracy at k = 1, 3, 5 and 7 has been obtained and compared for best accuracy. In case of MLP, learning rate and momentum of backpropagation are set at default value of 0.3 and 0.2 respectively. Also, 80 neurons in hidden layer are used with resulting 32 targets in the output layer. In case of SVM, accuracy for c = 50, 100 and 150 has been obtained and compared. Kernel function used is polynomial function and tolerance parameter is set at default value of 0.001. Also, comparison among the accuracies obtained using these classifiers with cross validation and percentage split method have been depicted using graphical representation.

Table 4.8: Recognition rates using 5-fold cross validation technique.

Classifier		5-fold Cross Validation			
		Recognized Instances	**Misclassified Instances**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1671	348	0.04	82.76
	K=3	1597	422	0.03	79.09
	K=5	1550	469	0.03	76.77
	K=7	1508	511	0.02	75.97
Support Vector Machines (SVM)	**C=1**	1800	219	4.54	89.15
	C=50	1786	233	4.49	88.45
	C=100	1786	233	4.65	88.45
	C=150	1786	233	5.64	88.45
Multilayer Perceptron (MLP)		1804	215	725.84	89.35

Table 4.9: Recognition rates with 10-fold cross validation technique.

Classifier		10-fold Cross Validation			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1689	330	0.013	83.65
	K=3	1618	401	0.04	80.13
	K=5	1559	460	0.02	77.21
	K=7	1534	485	0.04	75.97
Support Vector Machines (SVM)	**C=1**	1801	218	5.72	89.20
	C=50	1786	233	4.61	88.45
	C=100	1786	233	4.46	88.45
	C=150	1786	233	5.05	88.45
Multilayer Perceptrron (MLP)		1826	193	892.64	90.44%

Table 4.10: Recognition rates with percentage split method.

Classifier		Percentage Split (66%)			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	564	122	0.02	82.21
	K=3	533	153	0.03	77.69
	K=5	519	167	0.02	75.65
	K=7	505	181	0.02	73.61
Support Vector Machines (SVM)	**C=1**	603	83	4.48	87.90
	C=50	606	80	4.66	88.33
	C=100	606	80	5.37	88.33
	C=150	606	80	5.37	88.33
Multilayer Perceptron		608	78	781.28	88.62

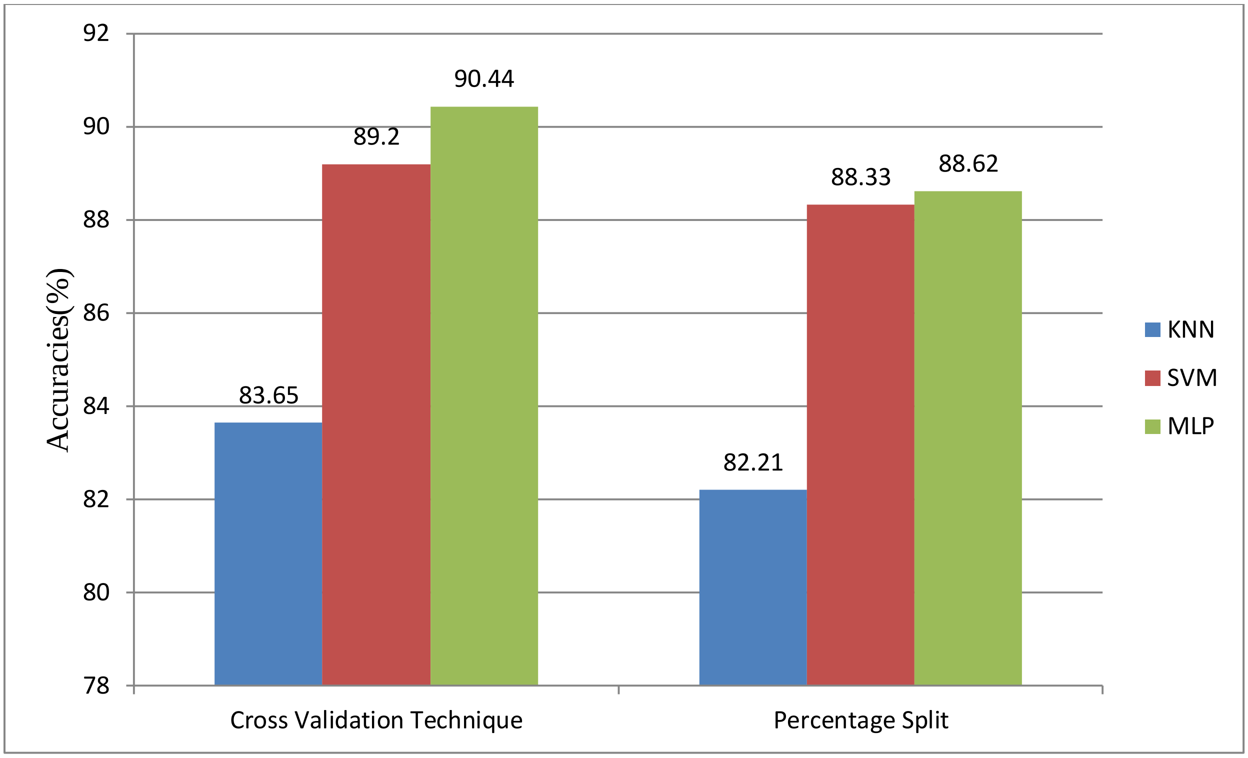


Figure 4.3: Comparison of results obtained from cross validation and percentage split method.

Figure 4.3 is a graphical representation of the results obtained using cross validation and percentage split technique when the combination of Spatiotemporal and Tangential features are used for classification. From graph, it can be observed that better results are produced using cross validation technique in comparison to the percentage split method. Highest accuracy is achieved using MLP with cross validation technique that is 90.44%.

From the above results of recognition for Spatiotemporal and Tangential features, it has been observed that for KNN with 5-fold cross validation maximum accuracy is achieved for k =1 which is 82.76%, KNN with 10-fold cross validation provides maximum accuracy at k = 1 which is 83.65%, KNN with percentage split method provides maximum accuracy at k = 1 which is 82.21%. For the case of MLP, maximum accuracy obtained with cross validation method is 90.44%. For the case of SVM, maximum accuracy is obtained for c = 1 that is 89.15% with 5-fold cross validation technique, maximum accuracy is obtained for c =1 that is 89.20% with 10-fold cross validation technique, maximum accuracy is obtained for c = 50, 100, 150 that is 88.33% with percentage split method. It is observed that good accuracy rates have been obtained for these classifiers using these features and highest accuracy is obtained using MLP in 10-fold validation that is 90.44%.

4.4 Performance comparison using Hybrid of Spatiotemporal (P_{xy}) and Spectral features (S_{xy}).

Table 4.11 depicts the accuracies obtained using different classifiers when Spatiotemporal and Spectral features are combined. Classification is done using three classifiers: KNN, MLP and SVM. Proposed classifiers are applied on database using cross validation and percentage split method. The above mentioned table uses 5 fold cross validation technique. Table 4.12 uses tenfold cross validation technique. Table 4.13 shows result using percentage split method in which dataset is split at 66%. For KNN classifier, accuracy at k = 1, 3, 5 and 7 has been obtained and compared for best accuracy. In case of MLP, learning rate and momentum of backpropagation are set at default value of 0.3 and 0.2 respectively. Also, 80 neurons in hidden layer are used with resulting 32 targets in the output layer. In case of SVM, accuracy for c = 50, 100 and 150 has been obtained and compared. Kernel function used is polynomial function and tolerance parameter is set at default value of 0.001. Also, comparison among the accuracies obtained using these classifiers have been depicted using graphical representation.

Table 4.11: Recognition rates with 5-fold cross validation technique.

Classifier		5-fold Cross Validation			
		Recognized Instances	**Misclassified Instances**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1896	123	0.05	93.90
	K=3	1867	152	0.013	92.47
	K=5	1846	173	0.02	91.43
	K=7	1836	183	0.02	90.93
Multilayer Perceptron (MLP)		1895	124	664.71	93.85
Support Vector Machines	**C=50**	1918	101	4	94.99
	C=100	1917	102	3.22	94.94
	C=150	1917	102	4.05	94.94

Table 4.12: Recognition rates with 10-fold cross validation technique.

Classifier		10-fold Cross Validation			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	1898	121	0.02	94.00
	K=3	1873	146	0.02	92.76
	K=5	1856	163	0.03	91.92
	K=7	1836	183	0.02	90.93
Multilayer Perceptron (MLP)		1919	100	683.43	95.04
Support Vector Machines (SVM)	**C=50**	1918	101	4.06	94.99
	C=100	1916	103	3.38	94.89
	C=150	1912	107	4.13	94.70

Table 4.13: Recognition rates with percentage split method.

Classifier		Percentage Split (66%)			
		Recognized	**Misclassified**	**Classification Time(s)**	**Accuracy (%)**
K-Nearest Neighbor (KNN)	**K=1**	637	49	0.04	92.85
	K=3	621	65	0.03	90.52
	K=5	618	68	0.02	90.08
	K=7	607	79	0.02	88.48
Multilayer Perceptron (MLP)		649	37	680.62	94.60
Support Vector Machines (SVM)	**C=50**	652	34	3.14	95.04
	C=100	650	36	3.7	94.75
	C=150	650	36	3.68	94.75

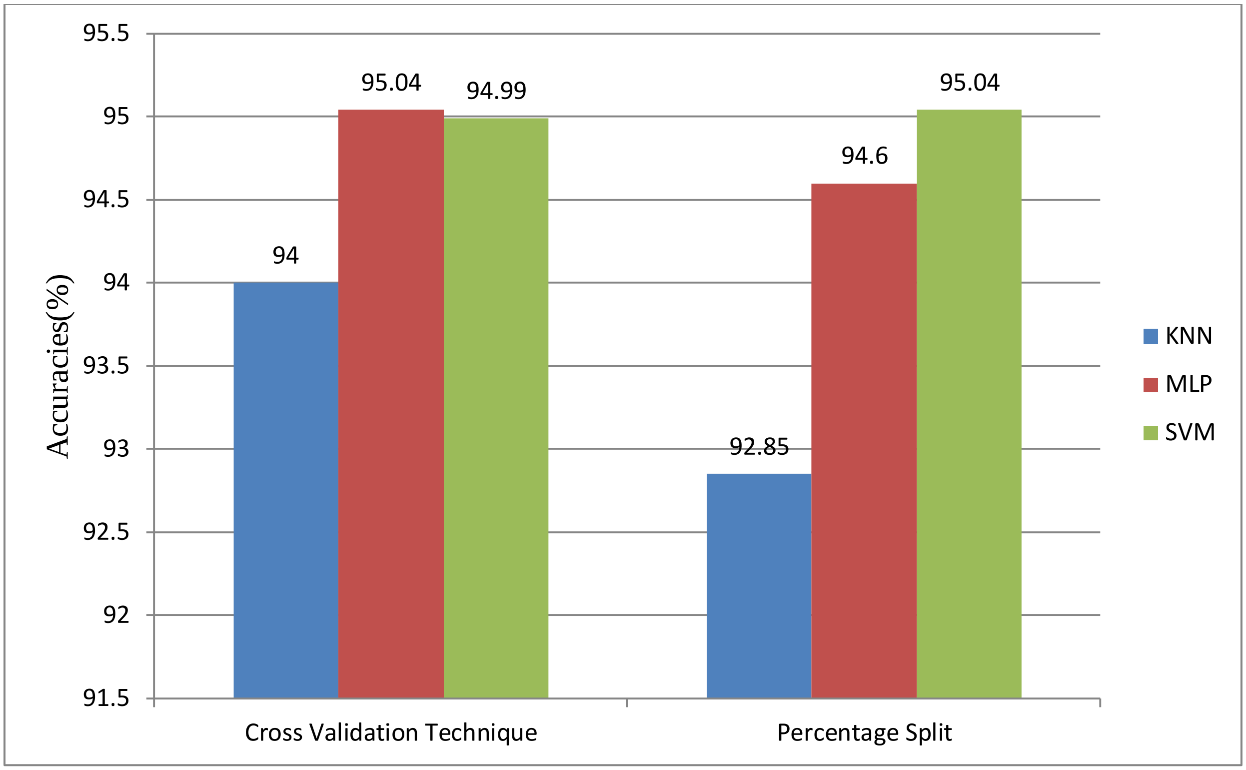


Figure 4.4: Comparison of results obtained from cross validation and percentage split method.

Figure 4.4 depicts the comparison of accuracies obtained with different classifiers using cross validation and percentage split method. Better accuracies are achieved using cross validation technique in comparison to percentage split method as it can be observed from graph also. It can also be seen from the graph that maximum accuracy is obtained using MLP with cross validation technique that is 95.04% and SVM with percentage split that is 95.04%.

From the above results of recognition for combination of Spatiotemporal and Spectral features, it has been observed that for KNN with 5-fold cross validation maximum accuracy is achieved for k =1 which is 93.90%, KNN with 10-fold cross validation provides maximum accuracy at k = 1 which is 94.00%, KNN with percentage split method provides maximum accuracy at k = 1 which is 92.85%. In case of multilayer perceptron with 10 fold cross validation technique, accuracy is 95.04% and with percentage split method, accuracy is 94.6%. For the case of SVM, maximum accuracy is obtained for c = 50 that are 94.99% with 5-fold cross validation technique, maximum accuracy is obtained for c =50 that are 94.99% with 10-fold cross validation technique, maximum accuracy is obtained for c = 50 that are 95.04% with percentage split method. It is observed that pretty good accuracy rates have been obtained for these classifiers using these features and highest accuracy is obtained using MLP in 10-fold validation that is 95.04% and SVM in percentage split that is 95.04%.

Overall observation tells that accuracy rate using cross validation technique in comparison to percentage split method is high. Also, it can also be inferred from the results that tenfold cross validation give better accuracy in comparison to fivefold cross validation. It is observed from the overall results that MLP classifier achieved higher accuracies in most of the cases and also overall good accuracies are obtained with Spatiotemporal and Spectral features when combined. So, highest accuracy which is achieved is 95.04% for Spatiotemporal and Spectral features combined using multilayer perceptron applied using tenfold cross validation technique. It has also been observed that SVM also provide good accuracies and the highest accuracy obtained is 95.04% for the Spatiotemporal and Spectral features combined for percentage split method. KNN also shown good results and the highest accuracy obtained by it using tenfold validation is 94.00% for the Spatiotemporal and Spectral features combined. Also in case of KNN classifier, it has been observed that accuracies are higher when value of k is set at 1 and afterwards it decreases.

CHAPTER 5

CONCLUSION

In this thesis, online handwriting recognition system has been developed for Gurmukhi script. Gurmukhi script is widely spoken language across the globe and consists of large character set. The characters in Gurmukhi script are formed using one or more strokes. Therefore, stroke based approach is followed in the thesis because strokes are unique. Stroke classification is very important step to recognize the characters.

Input data is collected in the form of handwritten strokes. For the work, 32 stroke classes are considered and samples which are belonging to each class are given in tabular form in the chapter 3. After data acquisition phase, stroke is preprocessed to remove the variations produced due to software and hardware limitations. Then from the preprocessed stroke, features are extracted. Spatiotemporal, spectral and tangential features are extracted. Hybridization of these features is then further used with different types of classifiers to achieve recognition. Three types of classification algorithms are used namely KNN, MLP and SVM. These algorithms are applied using two techniques: cross validation and percentage split technique.

From the results, it has been observed that cross validation technique provides better results than percentage split technique. Also, for tenfold cross validation results are better than fivefold cross validation. It is because cross validation technique provides following advantages over percentage split method:

1. It improves the generalization ability of the classifier having optimized parameters by averaging the classification accuracies of k classifiers.
2. It makes full use of limited data by ensuring that all of the dataset is involved in training and validation for classification.
3. It uses every data point exactly once as test set and trained k-1 times irrespective of how the data is divided.

But the problem with the cross validation is that it takes k times computation to make an evaluation.

Highest accuracies are obtained using MLP classifier in most of the cases and also, other two classifiers SVM and KNN are producing good accuracies. It has been observed that SVM and KNN are better than MLP in terms of speed but MLP has high noise tolerance compared to them. It has also been observed from the experimental results that KNN classifier works best when the value of k is kept at 1 and also optimum value of k depends upon the data. It could be if test data have high similarity with the training data. Among the features extracted, highest accuracy has been achieved using hybrid of spatiotemporal and spectral features and other features are also providing good results.

Table 5.1: Overall conclusion on basis of experimental results.

Features	**Accuracies (%)**					
	KNN		**MLP**		**SVM**	
	Cross validation	**Percentage split**	**Cross validation**	**Percentage split**	**Cross validation**	**Percentage split**
Spatiotemporal features	89.35	88.33	89.89	89.21	89.64	89.50
Spectral features	80.28	78.13	86.87	85.27	84.94	83.96
Hybrid of Spatiotemporal and Tangential features	83.65	82.21	90.44	88.62	89.2	88.33
Hybrid of Spatiotemporal and Spectral features	94	92.85	**95.04**	94.6	94.99	**95.04**

Table 5.1 depicts that cross validation has produced better accuracies than percentage split method and these accuracies are represented by red values. Also, the highest accuracy of 95.04% represented by blue value has been obtained using MLP and SVM classifier. But SVM is faster and flexible than MLP.

5.1 Future Scope

This work is limited to stroke recognition. So, it can be extended to recognize the characters or words in Gurmukhi script for handwritten character recognition (HCR). Low recognition can be due to some confusing strokes in the Gurmukhi script which can be improved by incorporating zone detection algorithm. Also, performance rate can also be increased by adding more features and using their combinations. To improve the computational complexity of the algorithms, dimensional reduction can be used. Linear discriminant analysis (LDA) classification model can also be explored by using different combination of the features. Hybridization of different classifiers can also be explored.

REFERENCES

[1] Verma K, Sharma RK: Comparison of HMM- and SVM- based stroke classifiers for Gurmukhi script. Neural Computing and Applications, pp. 1-13, 2016.

[2] Gupta N, Gupta M, Agrawal R: Preprocessing of Gurmukhi Strokes in Online Handwriting Recognition. 3rd International Conference on Information Security and Artificial Intelligence (ISAI), vol. 56, pp. 163-168, 2012.

[3] Thapar University Digital Repository. Available at http://dspace.thapar.edu:8080/jspui/. Accessed 15 May 2016.

[4] Rekha A: Offline Handwritten Gurmukhi Character and Numeral Recognition using Different Feature Sets and Classifiers - A Survey. International Journal of Engineering Research and Applications (IJERA), vol. 2, pp. 187-191, 2012.

[5] Safdar QTA, Khan KU: Online Urdu Handwritten Character Recognition: Initial Half Form Single Stroke Characters. IEEE 12th International Conference on Frontiers of Information Technology (FIT), pp. 292-297, 2014.

[6] Parui SK, Guin K, Bhattacharya U, Chaudhuri BB: Online Handwritten Bangla character recognition using HMM. IEEE 19th International Conference on Pattern Recognition (ICPR), pp.1-4, 2008.

[7] Stroke based cursive character recognition. Available at https://arxiv.org/pdf/1304.0421.pdf. Accessed 22 March 2016.

[8] Patel SR, Jha J: Handwritten Character Recognition using Machine Learning Approach – A Survey. IEEE International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO), pp. 1-5, 2015.

[9] Kumar M, Jindal MK, Sharma RK: *k* -Nearest Neighbor Based Offline Handwritten Gurmukhi Character Recognition. IEEE International Conference on Image Information Processing (ICIIP), pp. 1-4, 2011.

[10] Singh G, Sachan M: A Framework of Online Handwritten Gurmukhi Script Recognition. International Journal of Computer Science and Technology (IJCST), vol. 6, pp. 52-56, 2015.

[11] Liu CL, Jaeger S, Nakagawa M: Online Recognition of Chinese characters: the state-of-the-art. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, pp.198-213, 2004.

[12] Mahto MK, Bhatia K, Sharma RK: Combined Horizontal and Vertical Projection Feature Extraction Technique for Gurmukhi Handwritten Character Recognition. IEEE International Conference on Advances in Computer Engineering and Applications (ICACEA), pp. 59-65, 2015.

[13] Singh G, Kumar CJ, Rani R: Performance Analysis of Different Classifiers for Recognition of Handwritten Gurmukhi Characters using Hybrid Features. IEEE International Conference on Computing, Communication and Automation (ICCCA), pp. 1091-1095, 2015.

[14] Siddharth KS , Jangid M, Dhir R, Rani R: Handwritten Gurmukhi Character Recognition Using Statistical and Background Directional Distribution Features. International Journal on Computer Science and Engineering (IJCSE), vol. 3, pp. 2332-2345, 2011.

[15] Gurmukhi script Overview. Available at http://www.sikhiwiki.org/index.php/Gurmukhi. Accessed 22 May 2016.

[16] Sharma A, Kumar R, Sharma RK: Rearrangement of Recognized Strokes in Online Handwritten Gurmukhi Words Recognition. IEEE 10th International Conference on Document Analysis and Recognition, pp. 1241-1245, 2009.

[17] Sharma A, Kumar R, Sharma RK: Online Handwritten Gurmukhi Character Recognition Using Elastic Matching. IEEE proceedings of International Congress on Image and Signal Processing, pp. 391-396, 2008.

[18] Gurmukhi script zones. Available at http://www.discoversikhism.com/images/punjabi/zones.jpg. Accessed 20 June 2016

[19] Renuka R, Suganya V, Kumar AB: Online Hand Written Character Recognition using Digital Pen for Static Authentication. IEEE International Conference on Computer Communication and Informatics (ICCCI), pp. 1-5, 2014.

[20] Almuallim H, Yamaguchi S: A Method of Recognition of Arabic Cursive Handwriting. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9, pp. 715-722, 1987.

[21] Lehal GS, Singh C: A Gurmukhi script recognition system. IEEE 15th International Conference on Pattern Recognition, vol. 2, pp. 557-560, 2000.

[22] Aparna K, Subramanian V, Kasirajan M, Prakash GV, Chakravarthy V, Madhvanath S: Online handwriting recognition for Tamil. IEEE Ninth International Workshop on Frontiers in Handwriting Recognition, pp. 438-443, 2004.

[23] Joshi N, Sita G, Ramakrishnan A, Madhvanath S: Comparison of elastic matching algorithms for online Tamil handwritten character recognition: IEEE Ninth International Workshop on Frontiers in Handwriting Recognition, pp. 444–449, 2004.

[24] Jayaraman A, Chandra SC, Srinivasa CV: Modular approach to recognition of strokes in Telugu script. IEEE Ninth International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp 501–505, 2007.

[25] Bhattacharya U, Gupta BK, Parui SK: Direction code based features for recognition of online handwritten characters of Bangla. IEEE ninth international conference on document analysis and recognition (ICDAR), vol. 1, pp. 58–62, 2007.

[26] Mondal T, Bhattacharya U, Parui SK, Das K: Online handwriting recognition of Indian scripts – the first benchmark. IEEE 12th International Conference on Frontiers in Handwriting Recognition, pp. 200-205, 2010.

[27] Wadhwa D, Verma K: Online Handwriting Recognition of Hindi Numerals using SVM. International Journal of Computer Application (IJCA), vol. 48, pp. 13-17, 2012.

[28] Singh G, Sachan M: Multi-Layer Perceptron (MLP) Neural Network Technique for Offline Handwritten Gurmukhi Character Recognition. IEEE International Conference on Computational Intelligence and Computing Research, pp. 1-5, 2014.

[29] Haider I, Khan KU: Online Recognition of single stroke handwritten Urdu characters. IEEE 13th International Multitopic Conference, pp. 1-6, 2009.

[30] Joseph SM, Hameed A: Online handwritten Malayalam character recognition using LIBSVM in matlab. IEEE National Conference on Communication, Signal Processing and Networking (NCCSN), pp. 1-5, 2014.

[31] Dinesh M, Sridhar MK; A Feature Based on Encoding the Relative Position of a Point in the Character for Online Handwritten Character Recognition. IEEE Ninth International Conference on Document Analysis and Recognition (ICDAR), vol. 2, pp. 1014-1017, 2007.

[32] Joshi N, Sita G, Ramakrishnan AG, Deepu V., Madhvanath S: Machine Recognition of Online Handwritten Devanagiri Characters. IEEE International Conference on Document Analysis and Recognition (ICDAR), vol. 2, pp. 1156-1160, 2005.

[33] Shape Descriptors. Available at http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/boundary-rep-desc.pdf. Accessed on 13 March 2016.

[34] Gupta A, Srivastava M, Mahanta C: Offline Handwritten Character Recognition Using Neural Network. IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE), pp. 102-107, 2011.

[35] Kubatur S, Ahmed MS, Ahmadi M: A Neural Network Approach to Online Devanagari Handwritten Character Recognition. IEEE International Conference on High Performance Computing and Simulation (HPCS), pp. 209-214, 2012.

[36] Hassiem AA, Sudirman R, Khalid PI: Handwriting Classification Based on Support Vector Machine with Cross Validation. 2013.

[37] Kumar M, Jindal MK, Sharma RK: Classification of Characters and Grading Writers in Offline Handwritten Gurmukhi Script. IEEE International Conference on Image Information Processing (ICIIP), pp. 1-4, 2011.

[38] Pal U, Wakabayashi T, Kimura F: Comparative Study of Devanagiri Handwritten Character Recognition using Different Features and Classifiers. IEEE 10th International Conference on Document Analysis and Recognition, pp. 1111-1115, 2009.

[39] Kumar M, Sharma RK, Jindal MK: Offline Handwritten Gurmukhi Character Recognition: Study of Different Feature – Classifier Combinations. Proceeding of the workshop on Document Analysis and Recognition, pp. 94-99, 2012.

PUBLICATIONS

1) Ramandeep Kaur, Mandeep Singh , "Stroke Based Online Handwritten Gurmukhi Character Recognition", accepted in IEEE Fourth International Symposium on Intelligent Informatics (ISI'16), 2016.
2) Ramandeep Kaur, Mandeep Singh, "Hybrid feature set for Online Handwritten Gurmukhi Character Recognition", submitted in Elsevier Journal, Pattern Recognition Letters (SCI).

Raman_Thesis

ORIGINALITY REPORT

12%	5%	9%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.thapar.edu:8080 Internet Source	2%
2	Advances in Intelligent Systems and Computing, 2013. Publication	1%
3	Verma, Karun, and Rajendra Kumar Sharma. "Comparison of HMM- and SVM-based stroke classifiers for Gurmukhi script", Neural Computing and Applications, 2016. Publication	1%
4	Hidalgo, Melisa J., Diana C. Fechner, Eduardo J. Marchevsky, and Roberto G. Pellerano. "Determining the geographical origin of Sechium edule fruits by multielement analysis and advanced chemometric techniques", Food Chemistry, 2016. Publication	<1%
5	Anuj Sharma. "Rearrangement of Recognized Strokes in Online Handwritten Gurmukhi Words Recognition", 2009 10th International	<1%