Segmentation of lā -consonant and duláwā -consonant combination strokes in online handwritten Gurmukhi script recognition

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Software Engineering

Submitted by
Navneet Kaur Kaleka
Roll no. 801431014

Under the supervision of:
Mr. Karun Verma
Assistant Professor, CSED

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147001

July 2016

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, *"Segmentation of lā-consonant and duláwā -consonant combination strokes in online handwritten Gurmukhi script recognition"*, in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in the Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Mr. Karun Verma* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other university.

Navneet

Navneet Kaur Kaleka

801431014

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Karun

Mr. Karun Verma
Assistant Professor
Computer Science and Engineering Department
Thapar University

Countersigned by

Dr. Maninder Singh
Head
Computer Science and Engineering Department
Thapar University
Patiala

Dr. S.S.Bhatia
Dean (Academic Affairs)
Thapar University
Patiala

ACKNOWLWDGEMENTS

It has been an enriching experience for me to pursue my ME in this esteemed institute. I take this opportunity to thank all people who have helped me and supported me.

I am expressing my sincere gratitude to my guide, Mr. Karun Verma, for his able guidance and valuable inputs towards my research work. His conscientiousness and unabated pursuit of excellence have been sources of inspiration. I am thankful to him for the patiently helping me and for the encouragement he has provided me to carry on with the research work.

I thank Dr. Deepak Garg, Head of the department, for providing adequate resources in the department for research.

I would also like to thank all faculty members of this institute who have taught me many values in life apart from the regular curriculum. For the help that the other teachers and the non-teaching staff provided, I thank them.

My parents have always supported me and I am thankful to them for their blessings and their faith in me. I thank my brother for his love and support. He has always stood by my side.

I thank the Almighty for all I have and for showing me the path in life.

ABSTRACT

Online handwriting recognition has been spot of interest for research and has been worked upon for a long time now. Online Handwriting recognition has been done for some scripts all over the world. Great milestones have been reached in research on the online handwriting recognition for Indian scripts as well. I decided to work on *Gurmukhi* script. The fastness in writing Gurumukhi has led to the cursive nature of the script, there by leading to a combination of strokes written in a single stroke. These types of strokes are unrecognizable to the classifier. Segmentation algorithm has been proposed that use the slope calculation method at every point and find candidate points for segmenting the stroke into individual basic strokes. The algorithms demonstrated an accuracy of 95% in segmenting various stroke combinations when written in a single stroke.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

HCR - Handwritten Character Recognition

HMM - Hidden Markov Model

MRF - Markov Random Field

OCR - Optical Character Recognition

PDA - Personal Digital Assistants

SVM - Support Vector Machine

CHAPTER 1 INTRODUCTION

1.1 Introduction to Online Handwritten Character Recognition

Handwriting is widely used as a means of communication among humans. Writing is a text based visual means of communication. Writing refers to the spatial graphic marks on a surface. It has more explicit information content than the same information conveyed in speech along with gestures and face expressions. Writing has evolved from the pictographic symbols to represent objects to the use of complex symbols which also represent sound in a language. Input devices like keyboard and mouse are some of the means used to communicate with machines. Efforts are being made to make handwriting as an effective means of interaction between a man and the machine. Handwriting recognition by machine involves the training of machine based on rules. Handwriting recognition becomes complex when neural activity is involved.

Handwritten character recognition (HCR) is applicable to pen based input devices. Variability in handwritten text is its most important feature. The two paradigms of HCR are Offline HCR and Online HCR. Differences in Online and Offline handwriting recognition are based on the input modes, representation, and processing and recognition strategies. OHWR refers to the real time recognition of handwriting of an individual which means that the characters are recognized as they are typed. Offline recognition refers to a process of recognition performed later than handwriting capture. It consists of methods to extract information from the scanned images of the handwritten documents. Image processing methods are applied to get textual information.

Online HCR involves the use of pen based input devices to capture the sequence of co-ordinate points as the character is written. This gives information of the number, order and direction and writing speed of strokes. A stroke is the points collected along the track of the tip of the pen. The points are taken from a pen-down event until a pen-lift event occurs. Today there is a number of software and devices which are capable of providing handwriting based interfaces to the computer. Online Handwriting recognition engines for various scripts have been made. The work for Indian scripts is still in progress. In this direction many achievements have been reported where high accuracy has been achieved in presence of few constraints that included writing style, small vocabulary size and writer dependency.

Although work has been done for the online recognition for *Gurmukhi* handwritten text but some roadblocks have been encountered. Writing in a fast manner has led to the cursive nature of the *Gurmukhi* characters. *Gurmukhi* handwritten data is difficult to recognize online when more than one characters are written in one stroke. In this thesis, a method for segmenting the newly formed composite stroke classes into individual basic strokes is presented for reducing the number of classes formed and thereby increasing the accuracy of recognition engine.

1.1.1 Need of Online HCR for *Gurmukhi* script

The languages which have small character set such as English which is taken from Latin script have been used on handwriting interfaces by various tablet and PDA (Personal Digital Assistants) making companies. The use of English language in such devices depends on personal choice and convenience of the user. However in Indian scripts like *Gurmukhi*, where the character set is large the task of handwriting recognition becomes difficult. The data set in Indian scripts is large because the characters set consist of the Vowels and the Consonants as well as the composite characters consisting of the combinations of consonants and vowels. Typing in Indian languages needs an average of 3-4 keystrokes for a single character and the learning of keyboard mappings for characters is not an easy task either. Keyboards have been designed for typing in *Gurmukhi* script. Not everyone who wishes to type in Punjabi can use the keyboard for fast typing. Thus, speech and handwriting recognition have emerged as a propitious interfaces for the Indian languages. These means are used in mobile phones and tablets. They are preferred over the input method of keypads.

There are some limitations for speech based interface. It may not ensure privacy and thus making it difficult for the user to use it when others are around unless the system is used in a secluded place. Noise is another factor that affects the input through speech. There by making the option of inputting data by using handwriting, desired in case of Indian scripts.

Online handwriting recognition systems have a wide range of applications. Some of these applications are: online form filling in census data collection and land acquisition departments, dictation for official work in state government offices, teaching of a language in educational institutes, messaging and news, publication and writing, and online compliant filling for grievance handling in legal matters. This is worth

mentioning here that continuous efforts to enhance the recognizing accuracy and usability of handwriting recognition system is required because of ever increasing use of computer systems in various public and private sectors.

1.2 *Gurmukhi* writing system

Gurmukhi script is the basis to write Punjabi. It is the state language of Punjab state in India. More than 100 million human habitants speak in Punjabi language around the world.

Gurmukhi is a syllabic alphabet which consists of consonants and vowels. They can be represented using symbols. The consonants each have an inherent vowel which can be changed to another vowel or muted by means of diacritics or other modifications. Vowels can also be written with separate letters when they occur at the beginning of a word or on their own as shown in Table 1.3.

When two or more consonants occur together, special conjunct symbols are often used which add the essential parts of first letter or letters in the sequence to the final letter.

Gurmukhi script has 35 basic characters called consonants, 6 special consonants with /*pairi bindi*/, 9 vowel modifiers called /*matras*/, two symbols for nasal sounds /*bindi*//and /*ṭippi*/, and one symbol that copies the sound of a long consonant /*adhak*/ and 10 numerals.

One of the main features of *Gurmukhi* script is that it is not inherently cursive and written in left to right direction starting from top to down manner. Many characters of *Gurmukhi* script have a horizontal line called head line in the upper zone. Characters in a word are joined by the head line. So there is no gap between characters in the vertical direction in the letters of the word. In OHW *Gurmukhi* script text, stroke is considered as the smallest unit. A valid combination of strokes forms a *Gurmukhi* character vowel. Further, the character combination leads to a word or an *akshara*.

Table 1.1 *Gurmukhi* writing zones

Upper zone -	It contains headline, sub part of vowels /*sihari*/, /*bihari*/ and complete vowels /*dulaanv* /, /*horah*/, /*kanaura*/, /*tippi*/, and /*adhak*/.
Middle zone -	It contains all consonants and sub part of vowels /*sihari*/, /*bihari*/ and one complete vowel /*kanna*/.
Lower zone -	It contains two vowels /*aunkar*/, /*dulinkṛe*/ and three symbols called /*pairian*/ characters /*háhá*/, /*rárá*/, and /*vává*/ to form conjuncted consonants.

The combination of the individual strokes results in the formation of characters and the character combination lead to a word. A word in *Gurmukhi* script can be horizontally separated into three different zones, namely, busy zone, lower zone and upper zone. The part above the headline denotes the upper zone where the vowel modifiers and sound modifiers reside, while the busy zone represents the area between the upper and lower zones consisting of the consonants and the subparts of the vowel modifiers. Below the middle zone is the lower zone where various vowels and parts of characters are present. In Fig. 1.1 a word is written in *Gurmukhi*, pronounced as '*rangoon'*. In the figure /*ṭipí*/ is in the upper zone, there are three consonants present in the busy zone and */dulāinkṛe/* matra is in the lower zone. The headline, also called the *shirorekha* in *Gurmukhi* script is shown in the figure.

An analysis done statistically has shown zone distributed percentage distribution of *Gurmukhi* symbols in Punjabi corpus in printed text has been shown by researchers. Figure 1.1 illustrates these three zones of a *Gurmukhi* word.

The Gurmukhi character set is shown in Table 1.2.

Fig 1.1 *Gurmukhi* writing zone

Table 1.2 *Gurmukhi* Character set

ੳ	/*ooṛá*/	ਅ	/*āiṛá*/	ੲ	/*íṛí*/	ਸ	/*sassá*/
ਹ	/*háhá*/	ਕ	/*kakká*/	ਖ	/*khakkhá*/	ਗ	/*gaggá*/
ਘ	/*kaggá*/	ਙ	/*ṅaṅá*/	ਚ	/*chachchá*/	ਛ	/*chhachhá*/
ਜ	/*jajjá*/	ਝ	/*cajhá*/	ਞ	/*náñá*/	ਟ	/*ṭaīnká*/
ਠ	/*ṭhaṭhṭhá*/	ਡ	/*ḍaḍḍá*/	ਢ	/*taḍhá*/	ਣ	/*ṇáṇá*/
ਤ	/*tattá*/	ਥ	/*thththá*/	ਦ	/*daddá*/	ਧ	/*tadhá*/
ਨ	/*nanná*/	ਪ	/*pappá*/	ਫ	/*phaphphá*/	ਬ	/*babbá*/
ਭ	/*pabhá*/	ਮ	/*mammá*/	ਯ	/*yayyá*/	ਰ	/*rárá*/
ਲ	/*lallá*/	ਵ	/*vává*/	ੜ	/*ṛáṛá*/		

There are six additional consonants created by placing a dot /*bindí*/ at the foot /*pairi*/ of the consonant called /*pairi bindí*/ which are given in Table 1.3.

Table 1.3 *Gurmukhi* consonants with /*pairi bindí*/

ਸ਼	ਖ਼	ਗ਼	ਜ਼	ਫ਼	ਲ਼
/*sassá*/ pairi bindi	/*khakkhá*/ pairi bindi	/*gaggá*/ pairi bindi	/*jajjá*/ pairi bindi	/*phaphphá*/ pairi bindi	/*lallá*/ pairi bindi

Table 1.4 provides the list of ten *Gurmukhi* vowels. *Gurmukhi* supports two forms of vowels: independent form and dependent form. Independent vowels do not require a consonant, dependent vowels require it. All consonants use dependent form of vowel.

Table 1.4 *Gurmukhi* vowel modifiers

Independent	Dependent	Name	With consonant
ਅ	-	/muktá/	ਕ
ਆ	ਾ	/kanná/	ਕਾ
ਇ	ਿ	/sihárí/	ਕਿ
ਈ	ੀ	/bihárí/	ਕੀ
ਉ	ੁ	/āūnkaṛ/	ਕੁ
ਊ	ੂ	/dulāinkṛe/	ਕੂ
ਏ	ੇ	/lā /	ਕੇ
ਐ	ੈ	/duláwā /	ਕੈ
ਓ	ੋ	/hoṛá/	ਕੋ
ਔ	ੌ	/kanāūṛá/	ਕੌ

Nasalisation using /*ṭipí*/ and /*bindí*/ is done. /◌ੰ/*ṭipí*/ and / ◌ं /*bindí*/ are used for producing the sound from the back of the tongue. These are used as superscripts in the upper zone. Gemination using /*adhak*/ is done. The /◌ੱ/ /*adhak*/ tells that the following consonant is doubled in speech sound. While writing a particular character /*adhak*/ is introduced as superscript in the upper zone and is used between the two consonants.

1.2.1 Constrained and unconstrained handwriting

Handwriting styles can be constrained or unconstrained. Constrained handwriting is boxed discrete. It is also space discrete. Unconstrained handwriting is cursive or mixed cursive in nature. In boxed discrete handwriting, each character is written inside a special box. Fig. 1.2 illustrates the boxed discrete handwriting. When each character is written separately with spaces and no character touches other character is called spaced discrete handwriting. If each character is written separately and touches other characters, it is referred to as run-on discrete handwriting. When characters in one word are connected and strokes are used more than once in individual character, it is referred to cursive handwriting. It is observed that most of the people write in mixed cursive styles that include mixture of spaced, run on discrete and mixed handwriting styles are

illustrated in Fig. 1.2. Each writer has own speed of writing. They use different shapes to write characters. In cursive handwriting no clear boundaries are shown between characters for distinguishing among them.

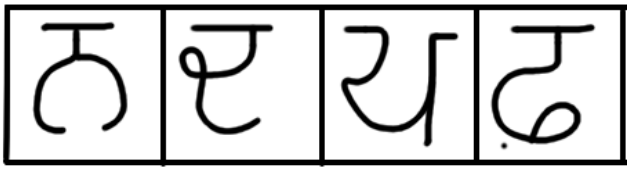


Fig. 1.2 Boxed discrete handwriting

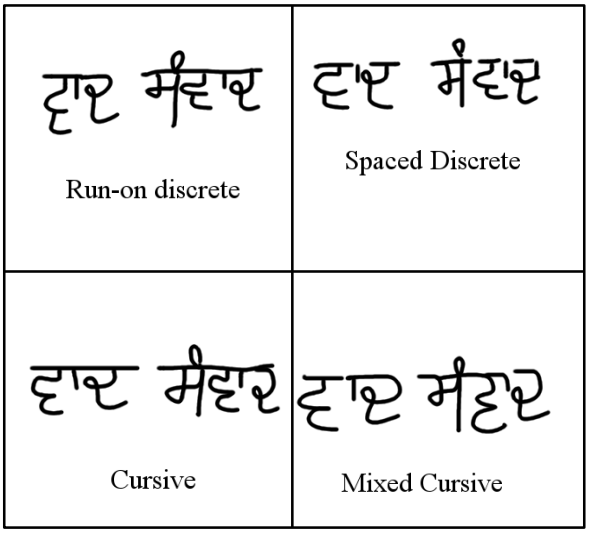


Fig 1.3 Different writing styles in *Gurmukhi* script

1.2.2 Difficulties in recognition of online handwritten *Gurmukhi* script

Recognition of Indian script poses various challenges because of the nature of Indian writing systems. The shape of Indian script characters is perceptually more complex than in many other writing systems. Its variation is accounted because of occurrence with a modifier. A character can be written using one or more strokes (In *Gurmukhi* a character can be written in a minimum of 1 or a maximum of 6 strokes). Some characters have common component stroke units. Hence it involves lesser count of classes to show the character set in *Gurmukhi* script. when using strokes compared to using character units.

The main issues encountered for online HCR with respect to *Gurmukhi* writing system include the following:

- Natural variability associated with handwriting:

  When a character is written by different writers or in a different situation or scenario it has different structure. The style of constructing words using the strokes is also different when different writers are involved.

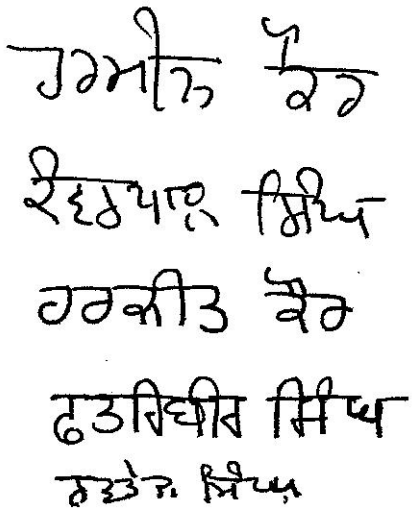


Fig. 1.4 Handwriting samples from 5 different people

- Number of stroke classes:

  The existence of compound characters in Indian writing systems contributes to a great count of stroke classes. They could represent consonants, vowels and modifiers or combinations of consonants and vowels. This and the inherent structurally complex strokes increase the complexity of the recognition system.

- Vertical connections of modifiers:

  Presence of consonant and vowel modifiers affects characters. Well more precisely the extent, horizontal or vertical. As a result of this, there is a significant variation in height in *Gurmukhi* characters which have vertical connections of vowel strokes.

- Choice of stroke representation:

  The handwritten stroke can be represented using various features. The features chosen for stroke representation should be capable of providing high discriminability across classes. Moreover, they should not be computationally

intensive, because online HCR systems are widely applicable in real-time systems.

- Recognition of characters and words from stroke level information:

  When a stroke level classification system is built for the writing system, it is necessary to propose methods for the identification of characters from strokes. The requirement of such methods is that they should not be time-consuming and should be flexible.

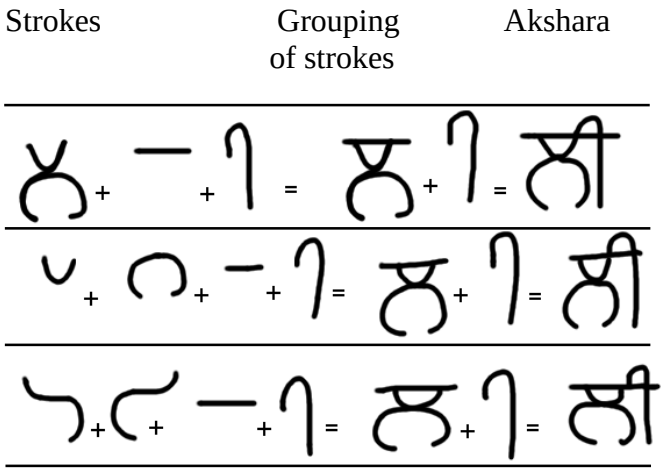


Fig 1.5 Illustration of grouping of single, and multiple strokes for formation of *Gurmukhi* character /ਲ/. (a) The akshara /*lee*/ /ਲੀ/ /*lallá + bihárí*/ composed of single character /*lallá* /, and /*bihárí*/. (b) and (c) The akshara /lee/ /ਲੀ/ /*lallá + bihárí*/ formed with multiple strokes for character /*lallá* / and single stroke for /*bihárí*/

- Variation of writing order of strokes in multi-stroke characters:

  At times the writer writes the strokes in a different way. By which we mean that the *matra* can be written first followed by consonant and vice versa. In writing a single character using multiple strokes can also lead to the inconsistencies in recognition.

- No pen-lifts while writing a stroke and random marks:

  These add to confusion and negatively affect the recognizing power of the engine.

- Inserting characters to the left of an already written character:

  This is an issue in Indian writing systems that are written from left to right. Such an issue never arises in speech recognition where continuity is

unambiguously maintained due to articulatory constraints or in offline character recognition which uses spatial information only. For online

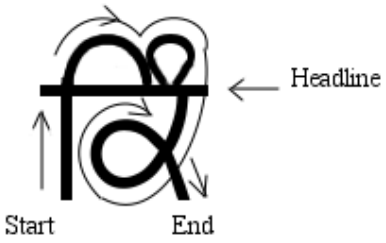


Fig 1.6 Illustrates the/*kakká*/ consonant with /sihárí/ and /*ṭipí*/ in one stroke.

handwriting, strokes are captured in the temporal order of writing. When strokes are inserted to the left of already written characters, they have to be spatially reordered.

- Horizontal extent of a character:

  There is inconsistency in the horizontal extent of characters when the character spans multiple non-overlapped horizontal units. Proximity analysis refers to the preprocessing step that is used to identify character and word units from the sequence of strokes captured by the pen-based input device.

1.2.3 External and internal segmentation

In the case of cursive writing, segmentation is imperative if character level identification is envisaged. When segmentation is performed in a separate step prior to recognition, it is called explicit or external segmentation. Implicit or internal segmentation involves the segmentation performed concurrent to the process of recognition. Segmentation is more critical in offline handwriting recognition but has become a much needed procedure in Online Handwriting recognition too. Segmentation into word units for holistic word recognition is the need. Criteria useful for segmentation include temporal and spatial separation or overlap across handwriting units. Morphological information or assumption from handwriting generation studies can also be used for segmentation.

1.3 Contribution to present work

The various ways in which a writer may write a word has been discussed in section 1. Among them there is a type where a person writes multiple characters in a single stroke. The combinational stroke is composed of different consonants together, may have two or more *matras* together or a combination of consonant and a *matra* together. The

combinational strokes are not recognized by the OHWR engine. The reason for writing to be that way may be due to a number of reasons. The reasons may include a person being in a hurry to write or it may also be the natural way of writing of a person, which in turn has led to the cursive nature of the *Gurmukhi* script.

A segmentation technique has been presented in this thesis which eases the recognition process and thus increases the efficiency of the recognition engine.

1.4 Outline of the thesis

This thesis is divided into seven chapters. A brief outline of the chapters included in the thesis is given in this section.

This chapter provides discussion on the motivation behind the proposed research that worked as a push to pursue the study on segmentation of strokes online handwritten *Gurmukhi* script. Further, this chapter comprises of an overview of broad features of *Gurmukhi* script; major issues in handwriting *Gurmukhi* script; and the complexities in recognition process due to these issues. In **second chapter**, a comprehensive review of literature on various stages involved in online handwriting recognition process has been presented. This chapter is divided into four sections. **Third chapter** explains the problem statement that has inspired me to take up the research on this topic. **Fourth chapter** explains the objectives and gives the outline of the proposed model. **Fifth chapter** consists of the proposed model and the system design for the problem statement stated in section 3.**Sixth chapter** shows the experimental results with screenshots and the statistical results of the study undertaken. **Seventh chapter** concludes the work and then future scope in the problem is stated keeping in mind what can be done further in the research area.

CHAPTER 2 LITERATURE REVIEW

Handwriting has helped humans to communicate with each other since early times. Efforts are being made to make handwriting as an effective means of communication between man and machine. Machines have to be trained to recognize the handwriting of individuals. Online Handwriting recognition (OHWR) for scripts like English and Chinese has been done (Plamondon, et al., 2000) (Wang, et al., 2012). With the advancement in technology Handwriting Recognition for Indian Scripts has gained significance. Handwriting recognition of Indian scripts is a tough task because they consist of large symbol sets and there is variability involved in the way a writer writes. Handwriting style differs from individual to individual. Many characters can be written in a single stroke (a stroke is made up of data points from when the pen is put down to write to pen up) thus posing as one more difficulty for online handwriting recognition. This problem arises because of the cursive nature of the handwriting of the users. Handwriting recognition model has a segmentation phase which is responsible for representing the data at stroke and character level. It helps in studying the stroke nature. Then they can be taken and studied individually.

2.1 Earlier work on recognition for Indian scripts

Designing a recognition system for handwritten characters of various Indic and non-Indic scripts has been discussed by various researchers (Bharath, et al., 2011) that included Arabic, Persian, Bangla (Bhattacharya, et al., 2012), *Devanagari*, Oriya, *Gujrati* and Kannada characters. Recognition of machine printed Indian scripts has works available for scripts, like Bangla, *Devnagari, Gurmukhi*, Tamil etc. Few works are there for segmentation of offline *Gurmukhi* handwriting.

Less work is present on the *Gurmukhi* character recognition and on online isolated *Gurmukhi* character/numeral recognition (Sharma, et al., 2008). Due to the vast variability of writing styles in *Gurmukhi*, for a single stroke various different sub strokes have been obtained as users have different handwritings. Lesser heed has been paid to the segmentation of online *Gurmukh*i handwritten text. A novel approach for segmentation of strokes for *Gurmukhi* handwritten online text is proposed. The algorithm works well for various types of stroke combinations and variations in shape.

Some rules were identified by analyzing the joining patterns of *Gurmukhi* characters. These rules were helpful in segmentation of text into strokes.

Some works are available for the machine printed recognition of some scripts (Zheng, et al., 2004). Zheng addresses to the problem of the identifying the text in noisy document images. They treat noise as different class. Then model noise based on the selected features. He treated segmentation and recognition techniques for the machine printed and handwritten text as different. Trained Fisher classifier was used to identify the text from noise. Markov Random Field- based (MRF) approach was used to model geometrical structure. Works well on text with noise. (Pal, et al., 2001) Pal gave a classification model to separate the handwritten and printed text from each other. It works for *Bangla* and *Devanagri* scripts. They may be present together in some documents. It hits the performance so separation is necessary. The model uses structural and statistical features of both types of the texts and gives an accuracy of 98.6%.

Work on the offline recognition of the *Gurmukhi* script has also be done in the recent past (Aggarwal, et al., 2015). Aggarwal uses the gradient and curvature feature set. Both of the features are then fused together. That is called a composite feature vector. SVM is used as the classifier. (Kumar, et al., 2011) Kumar gave a model for online HCR based on k-NN classifier. Character recognition model presented by Kumar uses the diagonal and transition features. It uses bitmap image. K-nearest neighbors are found using the Euclidean distance between testing point and reference point. Accuracy of 94.12% has been achieved.

The mostly and widely used process of recognizing online handwritten characters includes the sequential steps: data collection, pre-processing, feature extraction, segmentation, recognition and post processing (Jaeger, et al., 2001). These standard phases of recognition process are carried out in such a way that output obtained from the previous phase becomes the input for the next phase. Figure 2.1 describes the architecture of handwriting recognition process which shows the standard phases of the handwriting recognition process. Literature on the procedure of data collection and annotation is discussed in Section 2.1. Literature on the procedure of data collection and annotation is discussed in Section 2.1. Methods involved in pre-processing are reviewed in Section 2.2 of this chapter. Pre-processing includes the steps that are necessary to bring the input data into an acceptable form for feature extraction. In the

following sections the various phases of Online HWR are discussed as proposed by the researchers.

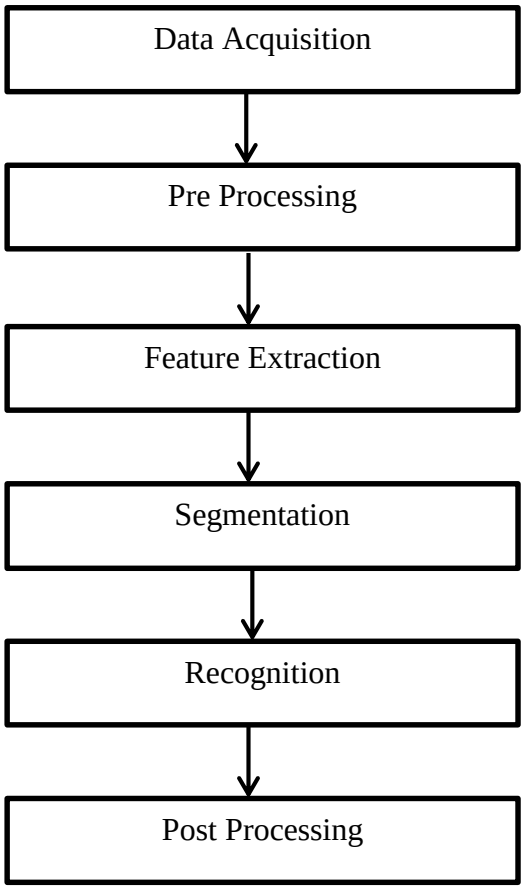


Fig 2.1 Phases of Online Handwriting Recognition System

2.2 Data Acquisition

First step towards the process of handwriting recognition is to collect data from a diversity of users. This step requires a gadget that will capture the handwriting. Electronic tablet or digitizer device that uses a digital pen (stylus) is used to collect handwriting data. Commercial products based on pen computing are available in the market. When we use a digital pen to write data, two movements are recorded, as Pen-Down and Pen-Up. These pen traces are sampled at constant rate. Thus these pen traces are evenly distributed in time and not in space. When a user moves digital pen on the writing pad, the time sequential signal captures the dynamic positions of the pen, *i.e.*, sequence of consecutive points on *x-y* plane in the form of coordinates with the help of sensors attached to the pad. Personal Digital Assistants (PDA) are an example of the Digitizers.

(a) **(b)**

Fig 2.2 Commercial products based on pen computing are available in the market. a) Digitizer b) A Tablet PC

In order to capture the full essence of this phase effort must be made in order to capture the full essence of the variation in the handwriting of the writers. (Jaeger, et al., 2003) here the data collection process described is the one for Japanese language. The input data of the handwritten Japanese characters are captured in boxes. This eliminates the need of segmentation in their work.

Tools are available that assist in data collection and annotation of handwritten data. The first PDA, Organizer was released in 1984 by Psion. Common personal digital assistants available in the market are Acer N Series, Apple Newton, Dell Axim, HP iPAQ, HP Jornada Pocket PC, iPAQ, Philips Nino, Sony Magic link with the Magic Cap operating system, Toshiba e310, Palm Taxi aka Palm-Pilot, Handsrping Visor, Tungsten E2, PalmPilot Personal and PalmPilot Professional.

2.3 Pre-processing

Few limitations like noise and distortion in input data makes the recognition process difficult when data is input. Pre-processing makes the segmentation and recognition process smooth and easy. It is used for removing noisy parts from handwriting and also helps in correcting imperfections. Pre-processing gives a uniform representation. It aims at enhancing of improving the classifying accuracy of recognition process. During online handwriting recognition process, pre-processing phase is helpful in removing noise or distortions present in input data. Such limitations of noise or distortions include abnormal size of text, missing points due to fast writing with the pen, jitter in text, slant handwriting and uneven distance of points from adjoining positions. In addition to this,

pre-processing is also useful to reduce the information content of the online handwritten signal.

The way a writer holds the pen, the emotions of the writer at the moment of writing and the variation in the writing of the writers are the causes of the noise. Various examples that show the presence of the noise in the input text include sharp edges, irregular sizes of text, non-centered text and missing points in text trajectories (Beigi, et al., 1995). Pre-processing is an important phase of recognition process. It improves recognition rate.

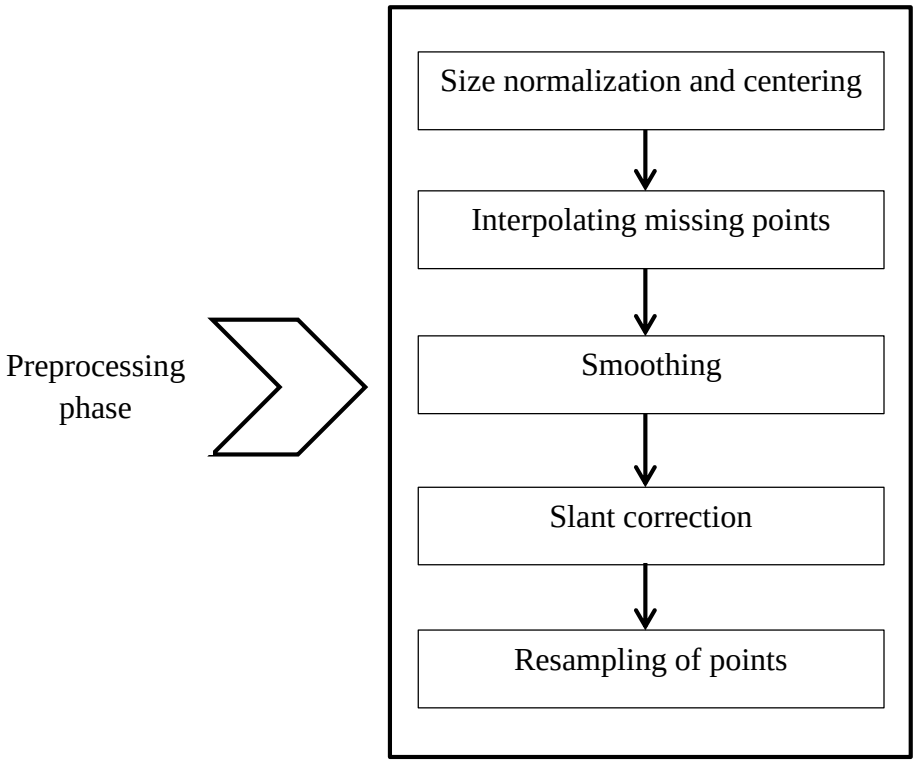


Fig 2.3 Preprocessing common steps

2.3.1 Pre-processing for removal of noise

The most common problem present or faced in handwriting recognition is the presence of the noise. The reason for the problem to occur may be many. For instance, hardware limitations, limitations of the digitizing process, crooked motion of the hand and inconsistent pen down indications. Various noise removal steps have been discussed here like, de- hooking and smoothing.

The skidding of pen in a particular direction gives the hooks problem, usually found at the start and the end of the handwritten character. It arises due to the habit of a person to write in a certain way. Inconsistent pen lifts also lead to the pen and paper contact

detection. De-hooking is the solution. Jitter is caused by the rough tip of the pen (Kavallieratou, et al., 2002). Smoothing is done to remove jitter. Low pass filter algorithm used by (Daifallah, et al., 2009) to reduce the noise for online Arabic Handwritten text. Wild points are formed due to the hardware problems. (Tappert, et al., 1990) explained use of fuzzy features to identify wild points.

2.3.2 Size normalization and centering

There is variability in handwritten characters. Normalization aims at the elimination of intra-class variability. To obtain uniformity in representation, process of normalization involves regularization of position, size and shape. Normalization is required as the size of handwritten character differs in handwriting from writer to another and even with same writer. Reason for this is that output of handwritten character depends on the trace of the pen, the writing pad and also on the style of written text. Size normalization normalizes every stroke to the same size and centered to a constant frame and places the text at a fixed distance from the origin. The use of size normalization techniques in online handwriting recognition has been discussed by (Beigi, et al., 1995). (Sharma, et al., 2010) Jhajj and Sharma also adopted size normalization where they used 48×48 size normalized image of *Gurmukhi* script characters and created 64 (8×8) zones. (Yamada, et al., 1990) discussed entirely different approach for normalization. They introduced non-linear normalization that aims to equalize the line density in order to exploit the area within the bounding box of the character. Centering is needed when pen is moved along the margin of the writing pad. A different approach to normalization, termed as non-linear normalization attempts to equalize the line density so as to utilize the area within the bounding box of the character more effectively. Here dense sectors of the writing area are expanded and sparse sectors are contracted, effectively adapting the size of the feature cell to the character pattern.

2.3.3 Slant correction and resampling

Cursive handwriting style creates the issue of the slanted handwriting. It may arise in other types of handwritings too. With an aim to reduce the variation in script and to increase the quality of the segmentation researchers have focused on slant correction with an aim to reduce variations in a script and enhance the quality of segmentation. Slant correction is applied for size normalization of the input words that usually occur due to variation in handwriting style. Existing literature provides deskewing algorithms for slant correction. Deskewing restricts the projection of non-diacritic characters on

the *x*-axis to be spatially separable and involves the absolute segmentation of characters especially in case of cursive word (Tappert, 1982). Histogram of directions of pen movement while writing a character gives the basic estimation of slant of handwriting. However, the main techniques for slant estimation and correction are projection method, run length based technique, generalized chain code estimator and extreme method .Resampling of points involves the points in the list to be equidistant from neighboring points where data points are considered from the original points in the list. Equidistant resampling is done to bring each stroke at equal intervals in space along with its trajectory. (Aparna, et al., 2004) described one dimensional linear interpolation technique that can be used as a procedure for uniform resampling. Existing literature provides detail on resampling techniques with emphasis on retaining information about corner points (Plamondon, et al., 2000).

2.4 Segmentation

Character or stroke level representation of data that leads to the individual study is achieved by the use of the segmentation phase of handwriting recognition. Segmentation becomes necessary when a writer does not write the basic level stroke to form a character. Much work has not been done for the segmentation of online *Gurmukhi* handwritten data. Combinational strokes are the ones that undergo the segmentation. The segmentation of the offline handwritten text is a very challenging task and has been a topic of research for years. Segmentation is of two types: external segmentation and internal segmentation. External segmentation is performed prior to recognition phase. External segmentation provides greater interactivity and saves computation time (Tappert, et al., 1990). Casey (Casey, et al., 1996) reviewed the research on segmentation and suggested that segmentation methods can be classified into four categories, *viz.* classical, dissection, hybrid, and holistic. They have inferred that segmentation is dependent on local topological as well as global contextual information. In addition to this, many researchers have classified segmentation methods based on their findings and experiences. Casey (Casey, et al., 1996) divided their survey on recognition into four categories as dissection technique, recognition based segmentation, over segmentation and holistic segmentation.

Work on the segmentation of Bangla handwritten text has been done (Pal , et al., 2003). This technique is for offline text. The Technique uses the document to be divided into vertical stripes. Horizontal histograms are made of the stripes. The vertical projection

lines formed by the minimum values in histogram are used to segment the lines to words. Another segmentation technique for Bangla script text has been proposed that uses the rules in the joining patterns of the strokes (Bhattacharya , et al., 2012). Lehal and Singh (2001) discussed text segmentation of machine printed *Gurmukhi* script and provided solutions for the difficulties related to connectivity of the characters on the headline, crossing between two or more characters in a word, multi-component characters and touching characters (Jindal, et al., 2009). Study on segmentation for offline handwriting recognition system is valuable to comprehend segmentation in online handwriting recognition system because a common task of word level segmentation exists in both offline and online handwriting recognition systems. Ha *et al.* (1995) studied mathematical expressions and proposed the use of *x-y* cuts as bounding boxes to separate the symbols. Elnagar and Alhajj (2003) discussed a new approach of segmentation based on separating single touching handwritten digit strings (Shi, et al., 1997). Important segmentation points are estimated based on decision line that is assumed from the deepest or highest valley in the image. The partitioning track is determined exactly and then the numerals are separated before restoration is applied. Shin (2004) provided a structural analysis algorithm that searches globally for important points of segmentation on a character unit level (Shin , 2004). Most important benefit offered by his approach is that it can cope with large variations in stroke shape and position. The search for key segmentation points is performed in a very systematic and logical manner with the help of two-level dynamic programming matching algorithm in conjunction with syntax control of composition characteristics. Jindal *et al.* (2006) proposed some new algorithms for segmentation of horizontally overlapping lines and applied this technique on Indian scripts (Jindal , et al., 2006). For this purpose they divided the whole document into strips and then proposed an algorithm for segmentation of horizontally overlapping lines and associating small strips to their respective lines. Their algorithms achieved 96.5% to 99.8% accuracy for different scripts.

Thus this section discusses the literature regarding the OHWR systems and phases.

CHAPTER 3 PROBLEM STATEMENT

Handwriting recognition has been an ever evolving topic. It has been talked and worked upon for the past four decades. Language when exercised in speech or hand written or typed for that matter gives it different dimensions. There is uniqueness in the way individuals write. That is what gives it a personal touch. A need for the handwritten data to be a means of communication between man and machine gave a push in the direction of the making of handwriting recognition systems. It is wonderful to have seen the system evolve and work for a lot of scripts in the world.

Technological advancement in India has given a humble reason for a handwriting recognition system in Indian scripts too. Smartphones have become a popular choice among the new generation and the older people too. Applications have turned out to be a boon for the people and there is an application for almost everything from finding places to dine to find places to buy. Collecting data has also been a herculean task in itself in the past but with recognition systems it has become less tedious. One's mother tongue is close to heart and sometimes it just is the only language that a person uses in his or her life.

In order to achieve the objective of a machine to recognize and understand the handwritten data of different individuals several roadblocks have been encountered. This made me into choose one of them as the problem statement and contribute to the ongoing study of OHWR for *Gurmukhi* script.

3.1 Objective

The various ways in which a writer may write a word has been discussed in section 1. Among them there is a type where a person writes multiple characters in a single stroke. The combinational stroke is composed of different consonants together, may have two or more *matras* together or a combination of consonant and a *matra* together. The combinational strokes are not recognized by the OHWR engine. The reason for writing to be that way may be due to a number of reasons. The reasons may include a person being in a hurry to write or it may also be the natural way of writing of a person, which in turn has led to the cursive nature of the *Gurmukhi* script.

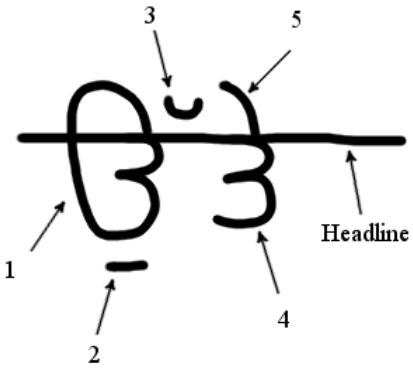


Fig. 3.1 An example where the numbers represent the order in which the strokes are written when writing *utte* in *Gurmukhi* script.

Fig 3.1 shows the way the basic non combination strokes are written on the interface and hence makes it possible to be identified by the classifier. Sometimes the order in which these strokes are written to be recognized plays a very important role. Fig 3.2 shows the basic strokes used to write the word '*utte*'.

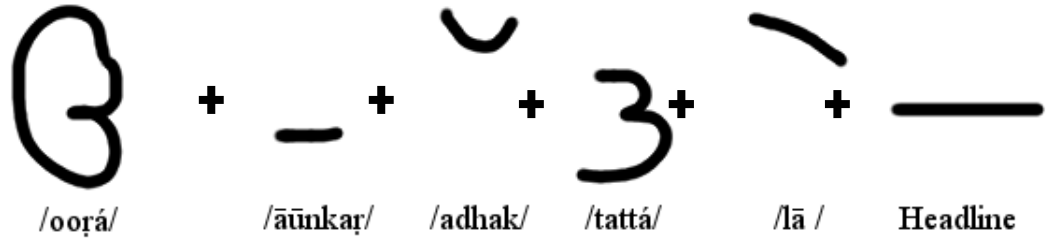
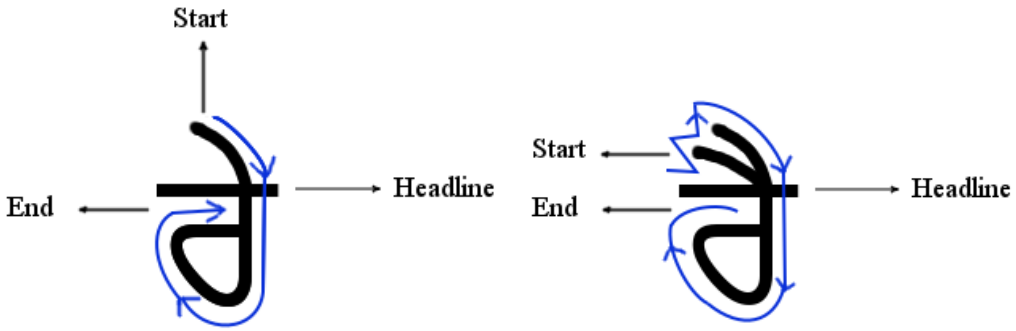


Fig. 3.2 The basic strokes using which the word *utte* is written in *Gurmukhi.*



(a) (b)

Fig. 3.3 /*rárá*/ consonant is here by written with two matras. (a) shows /*rárá*/ with */lā/* matra. (b) shows /*rárá*/ with /*duláwā* / matra.

Sometimes that is not case as explained. Fig 3.3 shows the way a person can write combination stroke in one stroke. Writing in one combination stroke means that the writer does not pick his pen up to write the *matra* or the vowel and the consonant individually. Instead he writes it in one single stroke. This poses as a problem to an extent that the word is not recognized by the classifier. The newly found strokes with combinations are put into a new class of strokes with different stroke ids. This makes the whole point of recognition a little absurd as new classes have to be fed to train the engine. This reduces efficiency and reliability to a great extent. A method for dividing the combinational stroke into different basic strokes has been presented in this work. The basic strokes are then fed to the recognition engine. As discussed in the Literature Review this phase of OHWR is segmentation.

In the case of cursive writing, segmentation is imperative if character level identification is envisaged. When segmentation is performed in a separate step prior to recognition, it is called explicit or external segmentation. Implicit or internal segmentation involves the segmentation performed concurrent to the process of recognition. Segmentation is more critical in offline handwriting recognition but has become a much needed procedure in Online Handwriting recognition too. Segmentation into word units for holistic word recognition is the need. Criteria useful for segmentation include temporal and spatial separation or overlap across handwriting units. Morphological information or assumption from handwriting generation studies can also be used for segmentation.

3.2 Preliminary analysis of stroke data

Online handwriting data for *Gurmukhi* script is a collection of characters. A character represents a syllable containing at most three consonant units and one vowel unit and is written as a combination of one or more strokes. A stroke based HCR system is used because of the large number of characters in Indian scripts with strokes common across the characters.

3.2.1 Data collection and annotation

I have created a Punjabi dictionary containing 150 often used Punjabi words. These 150 words cover all the basic characters, consonants and vowel modifiers. Handwritten data was collected from a group of 10 writers belonging to the age group of 15-45 years. The group comprised of school students, college students and government officials who

can write Punjabi language. I built an interface for collecting handwritten data. The data was collected on a Dell XPS tablet. Each one of the 10 writers had to write the 150 words from the dictionary, therefore a data set of 1500 words was collected. No restriction was imposed on writing.

In the collected dataset the average length of word is four characters and average number of strokes per word is six. Hence the data set comprises of near about 7860 strokes. Input data set consists of co-ordinates traversed across the tracks of pen, together with position of pen-down (start of stroke) and pen up (stroke end). The stroke data corresponding to a word is stored in a text file. In the file every row consists of the stroke id and the x-y co-ordinates corresponding to a single stroke. The points must be equal to or more than 128 points. It means that 64 or more for x and y each. Thus, I was able to grasp the pen pressure, pen down/up events and sequence of consecutive points on the *x-y* plane in the form of coordinates.

In this process, attributes of a handwritten character were captured like index, total number and the x-y coordinates of the stroke.

Table 3.1 Categories of *Gurmukhi* script writers used for data collection

Category	Category Name (Number of People)	Background
1	Highly Proficient (2)	Punjabi users such as Govt. employees / Language experts.
2	Moderately Proficient (3)	Users who have acquired formal Punjabi education, up to high school; but they do not write Punjabi frequently.
3	Low Proficient (3)	Users who have acquired formal Punjabi education, up to primary; but they write Punjabi frequently.
4	Ignorant (2)	Users who have no formal education in Punjabi; but can write Punjabi on the basis of structure.

3.2.2 Stroke level annotation

I did stroke level annotation and found that 10% of the data was not recognized by the SVM classifier. The unrecognized strokes were the ones where two or more characters were written in a single stroke. These strokes were taken as a problem set and a segmentation algorithm is proposed here. The segmentation algorithm divides the

unrecognized strokes into the basic independent strokes which are again sent for the recognition process.

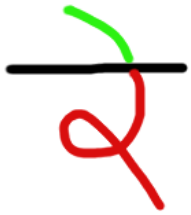


Fig. 3.4 /*kakká*/ consonant is written using three strokes, one stroke in black color, second in red and third in color green.

3.3 Problem Definition

3.3.1 Problem Description 1

/*lā/* + Consonant

In this case the writer has written /*lā /* and consonant in one stroke. For example /*rárá*/ consonant with /*lā / matra* are there and it is pronounced as‘re’ in the English word ‘prey’.

3.3.2 Problem Description 2

/duláwā / + Consonant

In this case the writer has written */duláwā /* and consonant in one stroke. For example /*rárá*/ consonant with */duláwā / matra* are there and it is pronounced as ‘ra’ in the English word ‘rat’.

A segmentation algorithm is proposed here in the text. The algorithm divides the combination strokes into basic strokes. The basic strokes are then fed to the classifying engine. The technique to recognize the *Gurmukhi* handwritten text uses the external segmentation technique, which means that the segmentation is performed prior to the recognition.

CHAPTER 4 METHODOLOGY

The outline of the proposed method is present in figure 4.1. The objectives are also outlined.

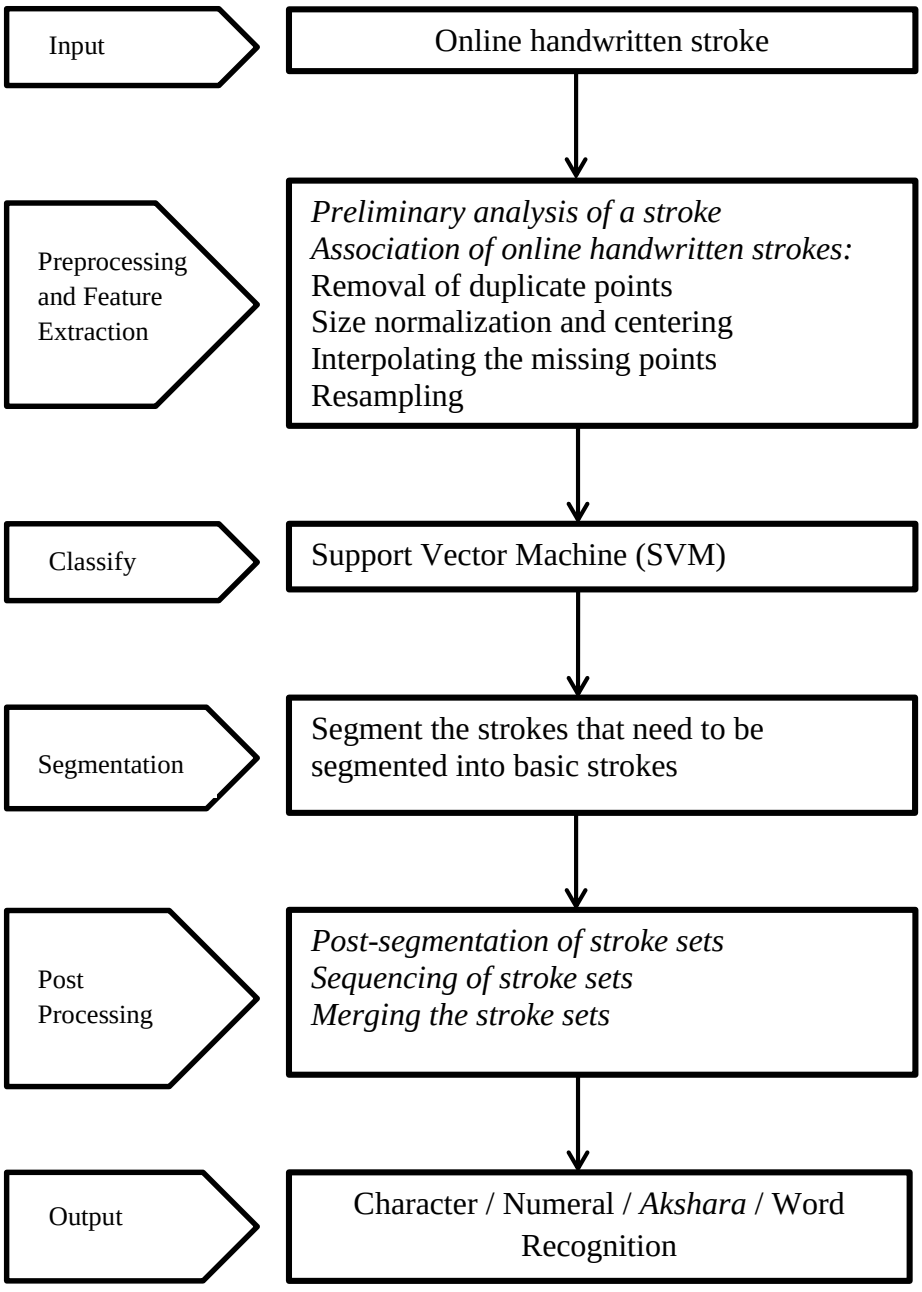


Fig 4.1 Steps in the proposed model

Existing framework for handwriting recognition clearly defines active areas for the development of such a system. From existing literature, it is evident that research in this area has been worked upon for more than four decades. Nevertheless, it is still admitted to be a challenging area which is encouraging researchers world-wide to refine existing methods.

Research in this area demands developing new techniques for segmentation that can achieve higher recognition accuracy. This study presents an algorithm for segmenting the strokes written in one stroke for online handwritten *Gurmukhi* script recognition.

4.1 Objectives

The goal of this research is to build and test the segmentation model for cutting and recognizing the handwritten strokes. The objectives are stated as follows:

- To categorize the strokes as combination strokes written in one single stroke.
- To give segmentation module to cut the combination strokes and then recognize them.
- To test the accuracy of the presented segmentation module.

CHAPTER 5 PROPOSED WORK

The segmentation phase is important phase as it helps in the recognition of the non-basic strokes or combination strokes. It increases the efficiency of the recognition process. It is made to find the candidate points for the optimal cut and thereby dividing the combination stroke into basic strokes. The application has been develop in Netbeans IDE.

5.1 Basic Theory

The proposed segmentation algorithm is based on the slope method. It means that it finds the slope between every two adjacent points of the stroke in order which they traced. Then taking decisions on finding candidate points based on the value of the slope.

5.1.1 Slope

Slope is the description of both direction and steepness of a line. It is measured as follows:

$$m = \frac{a_2 - a_1}{b_2 - b_1}$$

Here (b1 , a1) and (b2 , a2) are two points with the given x and y coordinates. The letter m can also be replaced by tanθ. θ Theta is the angle of incline. We use the increasing and decreasing slope fundamentals.

5.1.2 Assumptions

For the algorithm some assumptions have been made. The assumptions facilitate in the process of segmentation.

- Topline: It is provided to the writers already in the interface for them to write. It is synonymous to the *shirorekha* used in writing *Gurmukhi* characters and words.
- Bottom line: It is provided to give the writer an idea for the space area to use for writing.

- The input points for the stroke must be 64 or greater than 64. Failure to do so leads the segmentation module to not consider the stroke. By 64 or more points we mean that 64 or more values for x and 64 or more values for y.

5.2 Proposed Segmentation Algorithm

Algorithm to segment the stroke which consists of *lā -consonant* and *duláwā - consonant*

1. d[ ] = points captured in array of the input stroke
2. if values in d[ ] are less than 64
3. then return false and exit
4. else
5. i = total number of points captured.

   assumption = topline and bottom-line is known.
6. set k=0;
7. if (d[start].y > topline)
8. then repeat until k<i and increment k by 1.
9. do set angle = FindAngle (d[k].x , d[k].y , d[k+1].x , d[k+1].y)
10. if value in angle is negative and not equal to zero and lies between 60 and 90 degrees
11. then segment the stroke where it touches the topline with negative angle.
12. if (after touching the topline the angle is positive and is above top line)
13. segment the stroke where it again touches the top line with negative angle
14. else End.

The algorithm takes a variable k which is used to get the points in array. FindAngle method finds the slope between two points. Then the value of the slope is calculated and assessed.

5.2.1 Collection of the raw points of the input

The writer writes the stroke on the interface using the stylus or the pen. The trace points of stroke are then stored in an array. The points in the input stroke must be greater or equal to 64 points.

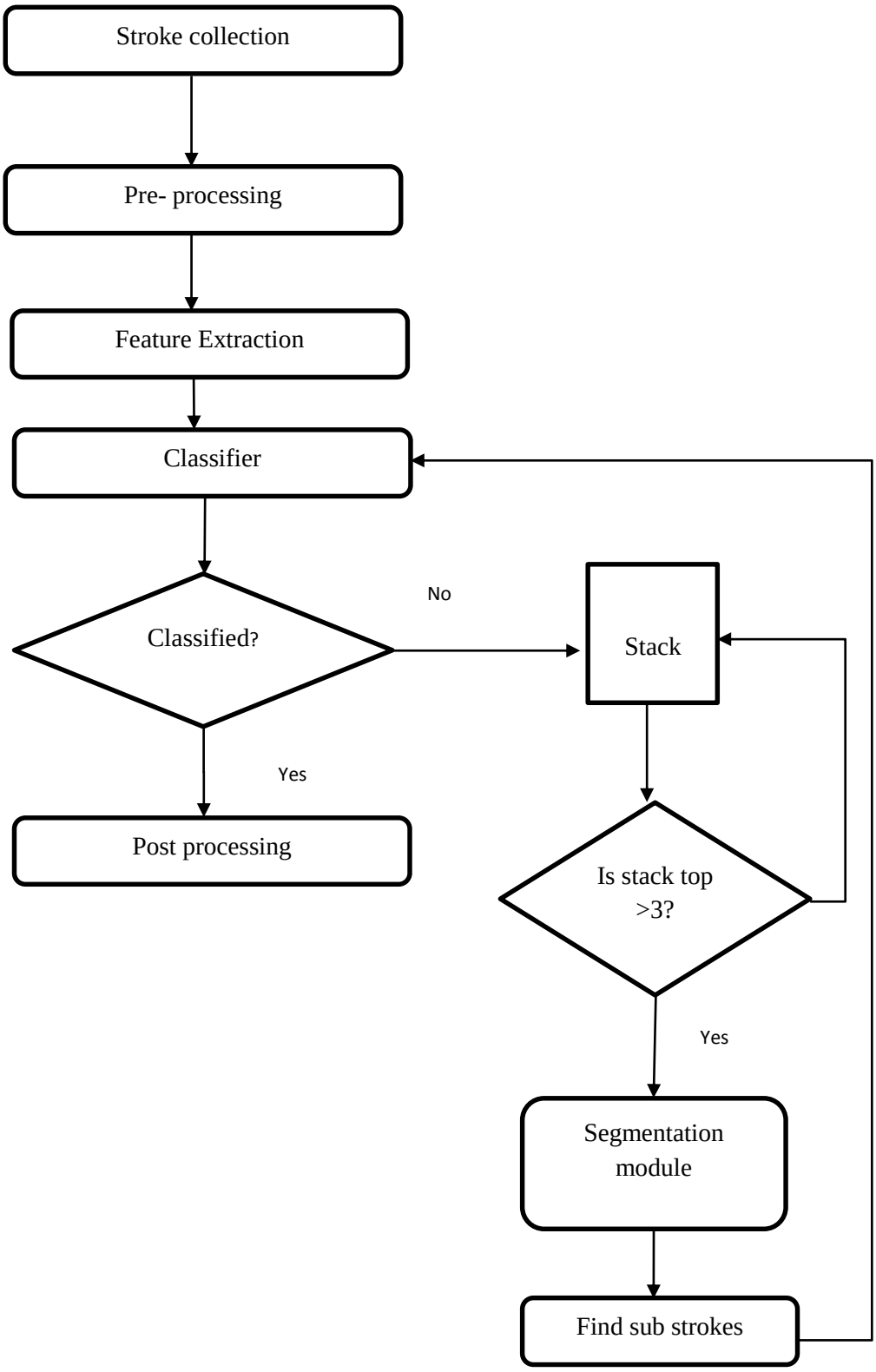


Fig. 5.1 System Design for segmentation module

D = {($xp1$, $yp1$), ($xp2$, $yp2$), … , ($xpn-1$, $ypn-1$), (xpn , ypn)}

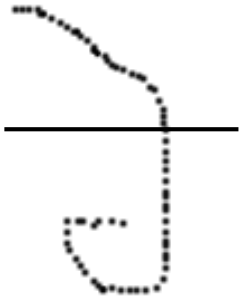


Fig.5.2 Raw x-y points of a stroke

5.2.2 Preprocessing

Pre-processing, being a preliminary step, serves various purposes that make the recognition process smooth and easy. It is used for removing noisy artifacts from handwriting and also helps in correcting imperfections. The pre-processing phase is helpful to remove noise and distortions present in input data due to hardware and software limitations during the online handwriting recognition process. A total of 64 points are stored.

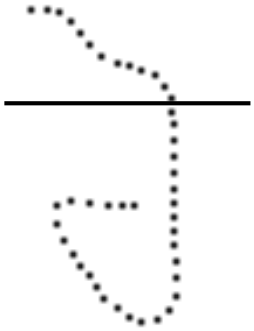


Fig.5.3 Preprocessed stroke points

5.2.3 Support Vector Machine classifier

Support Vector Machine (SVM) is the state of the art classification technique that has been used for recognition of various scripts. SVMs are a set of related supervised learning methods used for classification and regression. Application of SVM has achieved excellent recognition results for numerous pattern recognition problems. The standard SVM classifier takes the set of input data and predicts to classify them into one of the two distinct classes. To train an SVM classifier, a particular data set is given and a model is prepared for classification process. The base of this classification is normally the defined decision boundaries where, two or more distinct classes are

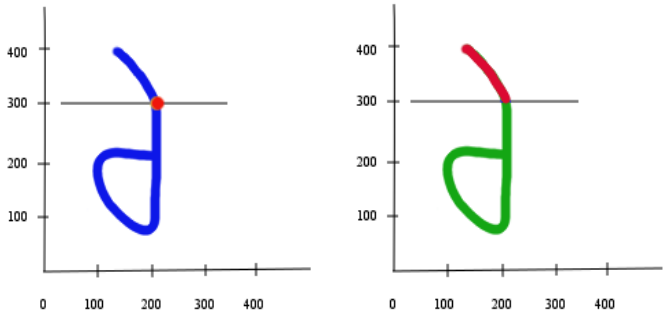
defined and a boundary is framed to distinct them. SVMs are binary classifiers that separate linearly any two classes by finding a hyper plane to the closest data points. The output of SVMs is based only on the data points that are at the margin and called support vectors.

The unrecognized combination strokes are taken and the array points are stored in a stack. The size of the stack is taken to be 3. When the stack is full then the stack top is popped and sent to the segmentation module. The strokes that are combination strokes and not recognized by the classifier are sent to the segmentation module.

5.2.4 Segmentation module

The module firstly checks the point and position from the headline of that point to examine the starting point of the stroke. If the stroke starts from above the Headline and has a negative slope and intersection with headline is there then the candidate point is found. There may be 0 or more candidate points for the stroke segmentation.

Figure shows the candidate point in the part a where combination stroke spelled as re is shown in blue color. The stroke after segmentation is shown in part b. the *matra* is of color red and the consonant is in color green. After successful segmentation the strokes are again sent to the classifier and when they are recognized properly then post processing steps follow.



(a) (b)

Fig.5.4 The stroke segmentation input and results. (a) shows the candidate point for the cut. (b) shows the stroke after segmentation

CHAPTER 6 EXPERIMENTS AND RESULTS

6.1 Experiments

The data collection of points has been illustrated in the following figure.

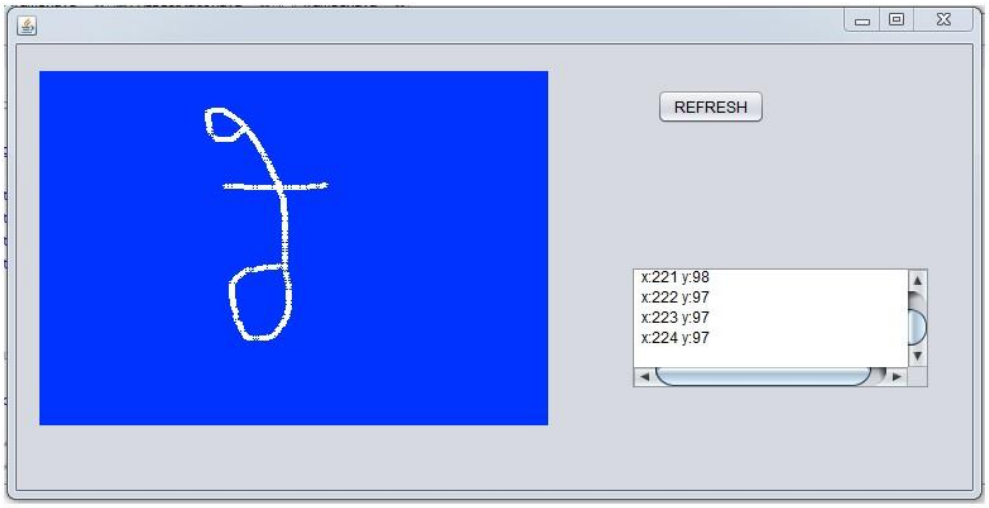


Fig. 6.1 The stroke is written on the blue background and on the right the points along the path are taken. They are represented as x and y coordinates.

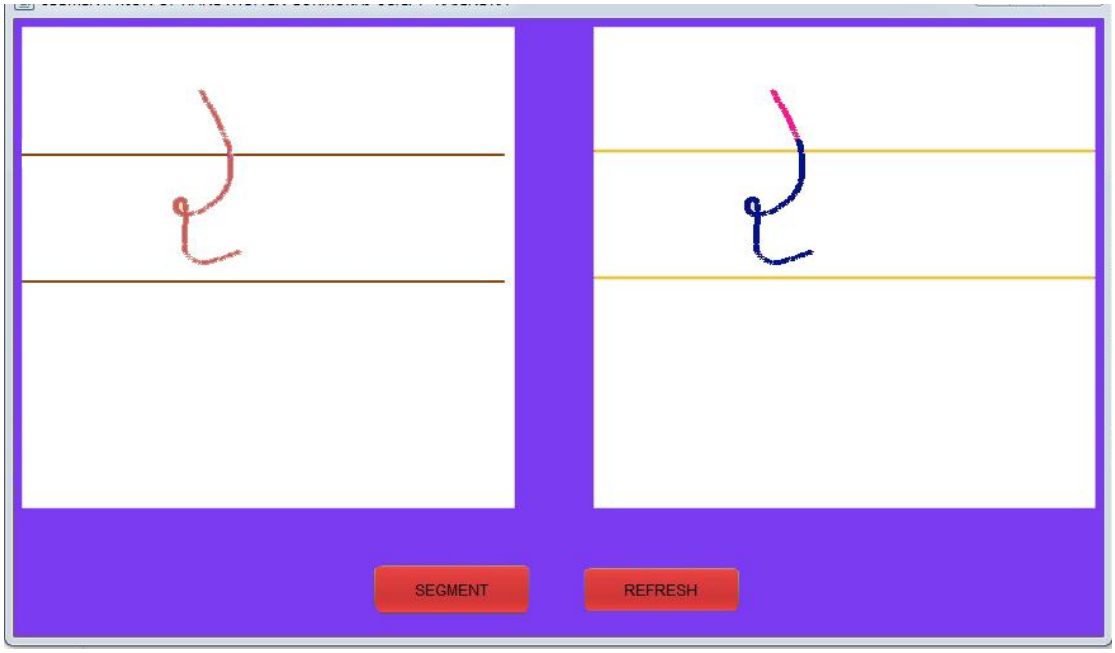


Fig. 6.2 illustrates the segmentation of stroke segmentation of /*daddá*/ and /lā /. The right represents the single stroke and the left is the segmented stroke.

Fig. 6.2 shows the segmented stroke in two different colors.

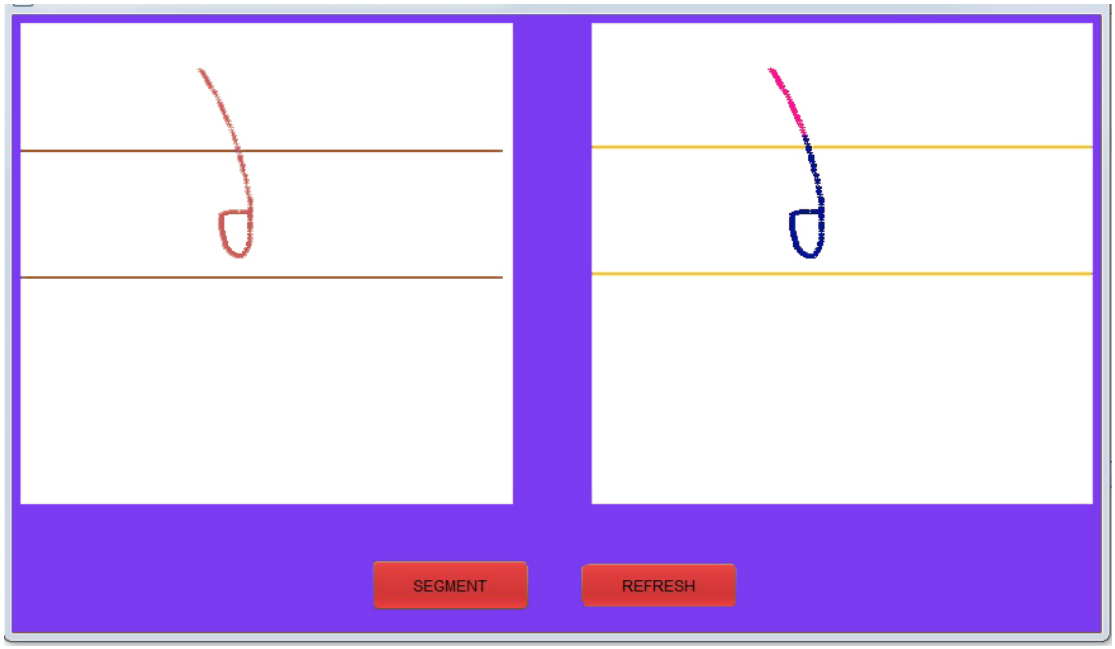


Fig. 6.3 shows the segmentation of the character pronounced as‘re’ in the English word ray. The stroke consists of the consonant /*rárá*/ and */lā /* vowel.

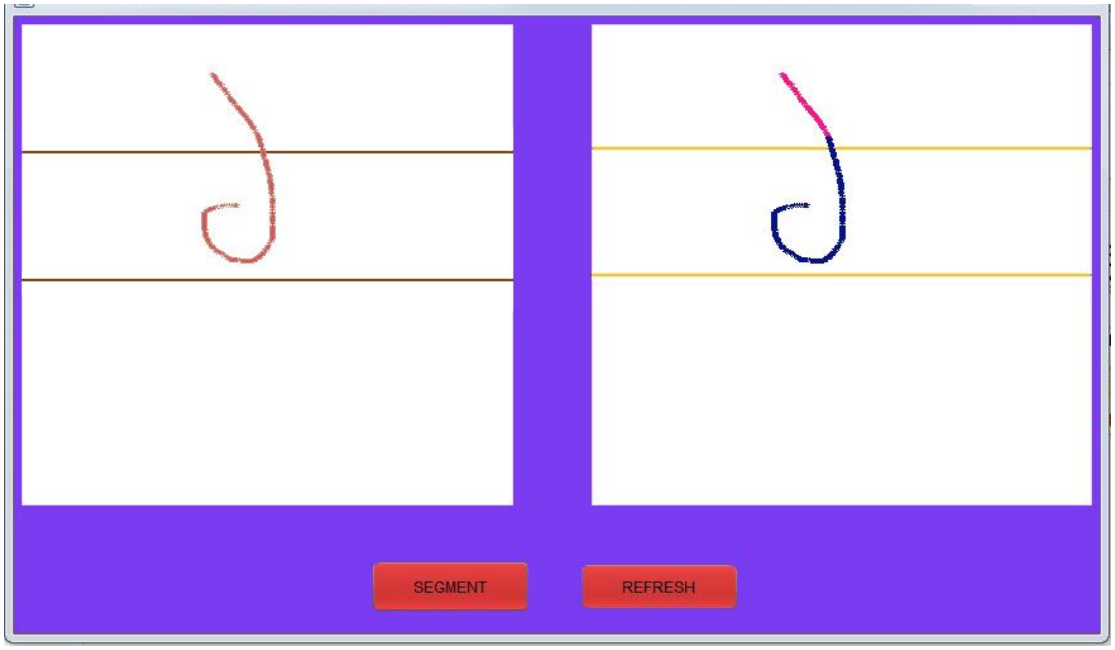


Fig. 6.4 shows the segmentation of the character pronounced as‘hey’ in the English word. The stroke consists of the consonant /*háhá*/ and */lā /* vowel.

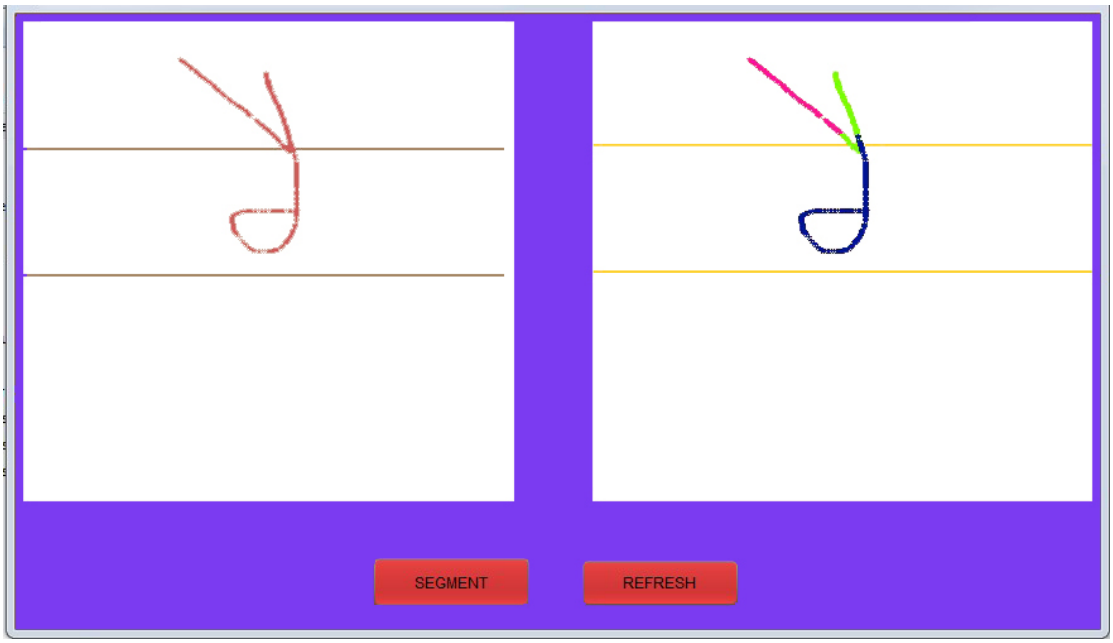


Fig. 6.5 shows the segmentation of the character pronounced as‘ra’ in the English word “rapid”. The stroke consists of the consonant /*rárá*/ and */duláwā /* vowel.

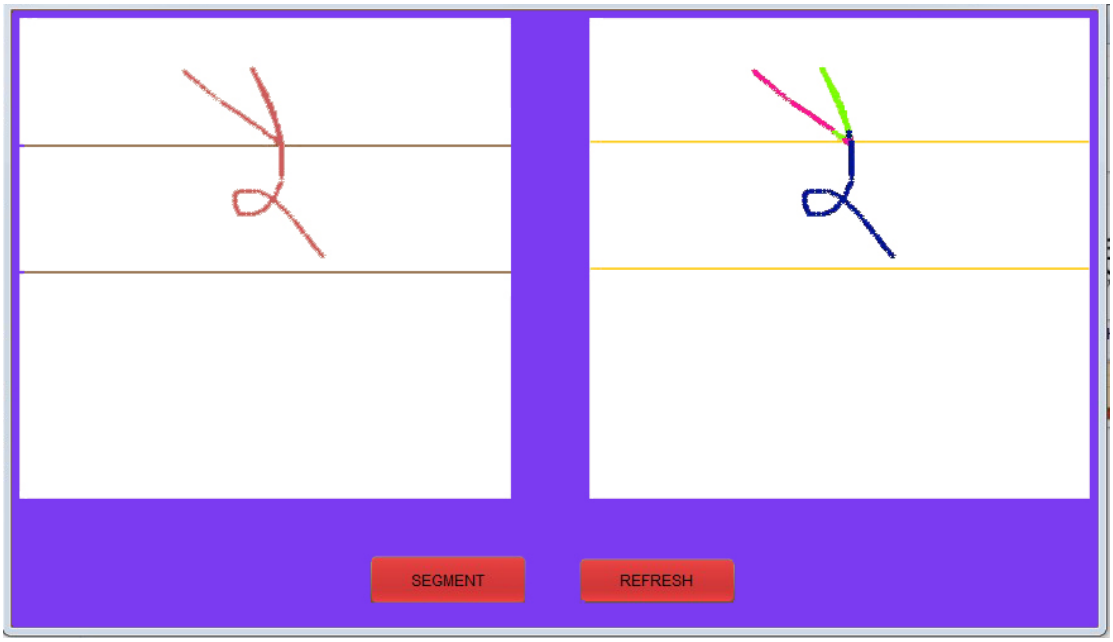


Fig. 6.6 shows the segmentation of the character pronounced as‘ka’ in the English word “cat”. The stroke consists of the consonant /*kakká*/ and */duláwā /* vowel.

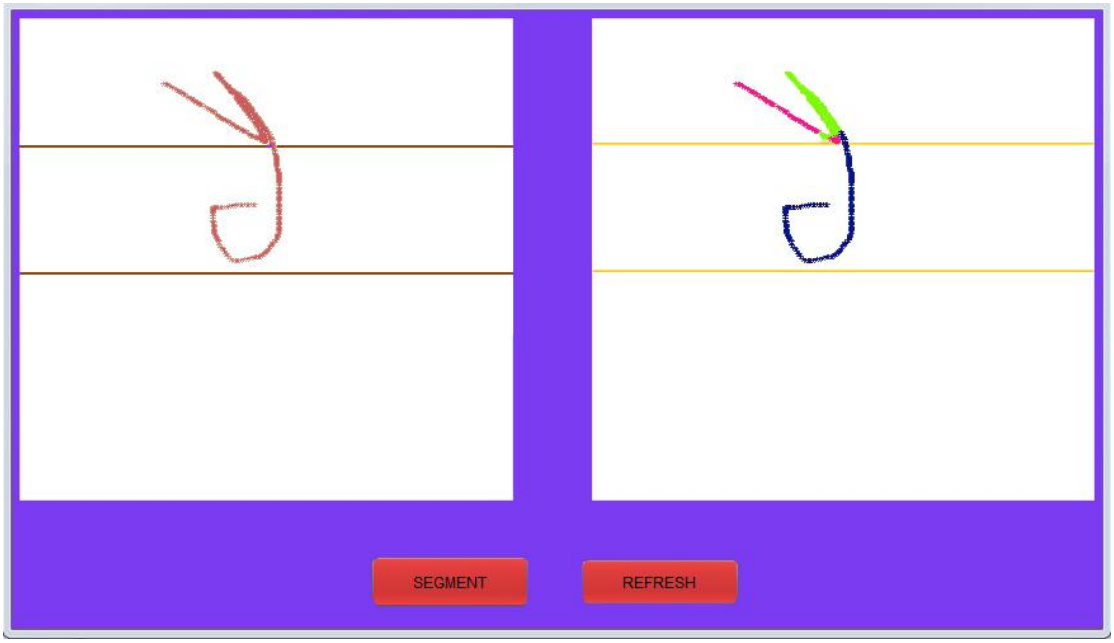


Fig. 6.7 shows the segmentation of the character pronounced as‘ha’ in the English word. The stroke consists of the consonant /*háhá*/ and */duláwā /* vowel.

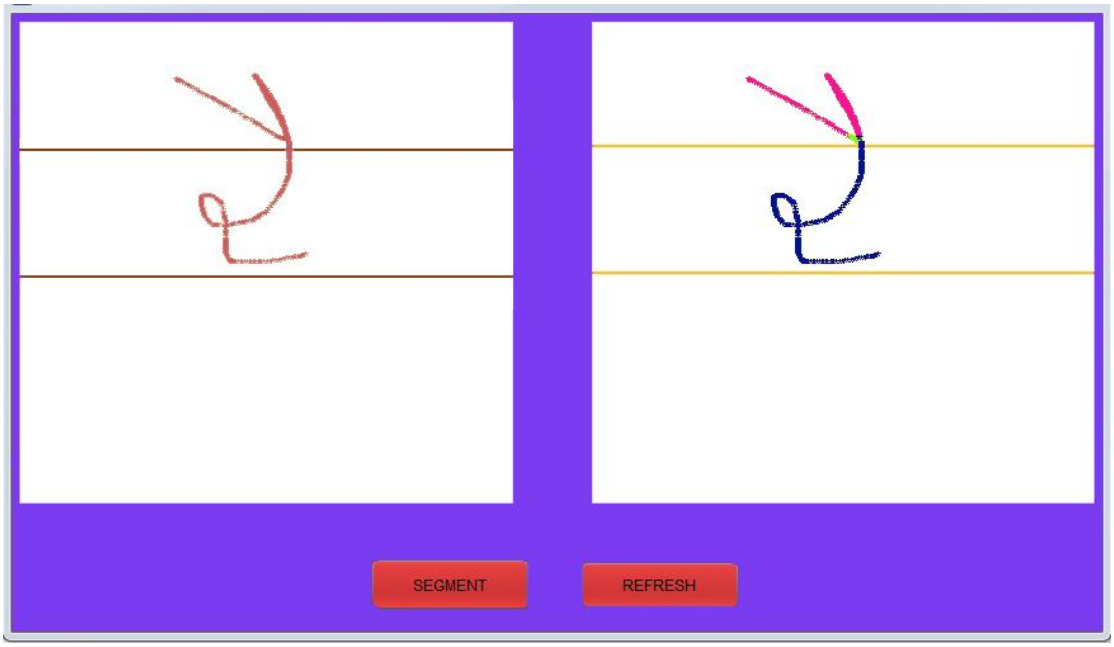


Fig. 6.8 shows the incorrect segmentation of the stroke that contains consonant /*daddá*/ and /*duláwā /* vowel. The writer does not touch the headline when first line of */duláwā /* is written and then it touches one time to write the consonant.

6.2 Results and Discussion

It is observed that apart from the successfully segmented strokes, the strokes that were incorrectly segmented consisted of the pen movement where the writer does not touches the headline when writing the */duláwā / matra*.

I have taken 30 input strokes written by 10 writers each to write cursively on the interface developed. This gives a total of 300 input strokes, out of which 285 were correctly segmented by the application developed. And 15 of the input strokes were incorrectly segmented. Hence it gives us an accuracy of 95% with an error rate of 5% as shown in Table 6.1 and Table 6.2.

Table 6.1 Results of the segmentation with different writers

Writer ID	Number of strokes written	Correctly segmented	Incorrectly segmented	% of Accuracy
A	30	27	3	90
B	30	28	2	93.33
C	30	28	2	93.33
D	30	29	1	96.66
E	30	30	0	100
F	30	28	2	93.33
G	30	28	2	93.33
H	30	29	1	96.66
I	30	30	0	100
J	30	28	2	93.33

Table 6.2 Results for the total strokes segmented

Total number of strokes	Correctly segmented	Incorrectly segmented	Total Accuracy(%)	% error Rate
300	285	15	95	5

CHAPTER 7 CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this paper, a novel approach for segmentation of online handwritten *Gurmukhi* script has been presented. The external segmentation method where the segmentation is done before the recognition is used. Firstly the data points of the stroke to undergo segmentation are collected in array and then after applying the preprocessing methods on them these points are sent for segmentation. The algorithm is directed towards the strokes containing the *matra* in the upper part of the combination character handwritten by the writer. The data points can be used in character recognition and also for shape recognition. The algorithm presented in the thesis in is based on the slope calculation between two consecutive points in the stroke. The method of the positive and negative slope is used about the headline. Then the candidate points are found on the basis of change of slope. After segmentation the data points of the strokes found are then again stored in an array. The points in the array are sent for the recognition. The proposed segmentation algorithm is robust in segmenting the combinations of characters present in a single stroke.

7.2 Future work

The proposed approach has given good results. Apart from that, certain limitations like the strokes with the *matra* or the vowel in the busy zone and the lower zone of the character or word were not segmented. In future work, the strokes which consist of the characters in the lower zone for example, pairi /*háhá*/*and pairi* /*rárá*/, can be taken for segmentation. In the strokes that have the matra in the busy zone can be taken up for segmentation, for example, words with */kanná/* can be taken up. Various new classes can be found and various new combinations of strokes can be segmented.

REFERENCES

Aggarwal Ashutosh and Singh Karamjeet Handwritten Gurmukhi character recognition [Conference] // Computer, Communication and Control (IC4), 2015 International Conference on. - Indore : IEEE, 2015. - Vol. 10. - pp. 1-5.

Aparna K. H., Subramanian Vidhya and Kasirajan M. Online Handwriting Recognition for Tamil [Conference] // IWFHR '04 Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition. - [s.l.] : IEEE, 2004. - pp. 438-443.

Basu Subhadip [et al.] A Fuzzy Technique for Segmentation of Handwritten Bangla Word Images [Conference] // Computing: Theory and Applications, 2007. ICCTA '07. International Conference on. - Kolkata : [s.n.], 2007.

Beigi H. S. M. and Nathan K. S. Real-time on-line unconstrained handwriting recognition using statistical methods [Conference] // Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on. - [s.l.] : IEEE, 1995. - Vol. 4. - pp. 2619-2622.

Bharath A. and Madhvanath Sriganesh HMM-Based Lexicon-Driven and Lexicon-Free Word Recognition for Online Handwritten Indic Scripts [Journal] // IEEE Transactions on Pattern Analysis and Machine Intelligence. - December 2011. - 4 : Vol. 34. - pp. 670-682.

Bhattacharya U. and Chaudhuri B. B. Offline recognition of handwritten Bangla characters: an efficient two-stage approach [Journal] // Pattern Analysis & Applications. - November 2012 . - 4 : Vol. 15. - pp. 445-458.

Bhattacharya Nilanjana and Pal Umapada Stroke Segmentation and Recognition from Bangla Online Handwritten Text [Conference] // ICFHR '12 Proceedings of the 2012 International Conference on Frontiers in Handwriting Recognition. - [s.l.] : IEEE, 2012. - pp. 740-745.

Casey R. G. and Lecolinet E. A survey of methods and strategies in character segmentation [Journal] // IEEE Transactions on Pattern Analysis and Machine Intelligence. - [s.l.] : IEEE, 1996. - 7 : Vol. 18. - pp. 690-706.

Daifallah Khaled, Zarka Nizar and Jamous Hassan Recognition-Based Segmentation Algorithm for On-Line Arabic Handwriting [Conference] // 10th International Conference on Document Analysis and Recognition. - [s.l.] : IEEE, 2009. - Vol. 10. - pp. 886-890.

Garg Naresh Kumar, Kaur Lakhwinder and Jindal MK Segmentation of handwritten Hindi text [Journal] // International Journal of Computer Applications. - 2010. - Vol. 1. - pp. 19-22.

Jaeger S. [et al.] Online handwriting recognition: the NPen++ recognizer [Journal]. - [s.l.] : International Journal on Document Analysis and Recognition, Springer, 2001. - 3 : Vol. 3.

Jaeger S., Liu C. L. and Nakagawa M. The state of the art in Japanese online handwriting recognition compared to techniques in western handwriting recognition [Journal]. - [s.l.] : Springer, Document Analysis and Recognition, 2003. - 2 : Vol. 6. - pp. 75-88.

Jindal M. K., Sharma R. K. and Lehal G.S. Segmentation of Horizontally Overlapping lines in Printed Gurmukhi Script [Conference] // 2006 International Conference on Advanced Computing and Communications. - [s.l.] : IEEE, 2006. - pp. 226-229.

Jindal M. K., Sharma R. K. and Lehal G. S. Segmentation of touching characters in upper zone in printed Gurmukhi script [Conference] // Proceedings of the 2nd Bangalore Annual Compute Conference. - Bangalore, India : [s.n.], 2009.

Jindal M. K., Sharma R. K. and Lehal G. S. Segmentation of touching characters in upper zone in printed Gurmukhi script [Conference] // COMPUTE '09 Proceedings of the 2nd Bangalore Annual Compute Conference. - 2009. - Vol. 9.

Jindal Munish, Jindal MK and Sharma RK Segmentation of Isolated and Touching Characters in Offline Handwritten Gurmukhi Script Recognition [Journal] // International Journal of Information Technology and Computer Science. - 2014. - Vol. 6. - p. 58.

Kavallieratou E., Fakotakis N. and Kokkinakis G An unconstrained handwriting recognition system [Journal] // International Journal on Document Analysis and Recognition. - [s.l.] : Springer, 2002. - 4 : Vol. 4. - pp. 226-242.

Koerich Alessandro L. and Sabourin Robert Lexicon-driven HMM decoding for large vocabulary handwriting recognition with multiple character models [Journal] // International Journal on Document Analysis and Recognition. - [s.l.] : Springer, 2003. - 2 : Vol. 6. - pp. 126-144.

Kumar Munish, Jindal M. K. and Sharma R. K. k-nearest neighbor based offline handwritten Gurmukhi character recognition [Conference] // Image Information Processing (ICIIP), 2011 International Conference on. - Himachal Pradesh : IEEE, 2011. - Vols. 1-4.

Lecun Yann and Bengio Yoshua Word normalization for on-line handwritten word recognition [Conference] // Proceedings of the International Conference on Pattern Recognition. - Jerusalem : IEEE, 1994. - Vol. 2. - pp. 409-413.

Pal U. and Datta S. Segmentation of Bangla unconstrained handwritten text [Conference] // Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on. - [s.l.] : IEEE, 2003. - pp. 1128-1132.

Pal U. and Chaudhuri B.B. Machine-printed and hand-written text lines identification [Journal]. - [s.l.] : Elseviar, 2001. - 3-4 : Vol. 22. - pp. 431-441.

Pal Umapada, Jayadevan Ramachandran and Sharma Nabin Handwriting Recognition in Indian Regional Scripts: A Survey of Offline Techniques [Journal] // ACM Transactions on Asian Language Information Processing . - 2012. - 1 : Vol. 11. - pp. 1-35.

Plamondon Réjean and Srihari Sargur N. On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey [Journal] // IEEE Transactions on Pattern Analysis and Machine Intelligence. - January 2000. - 1 : Vol. 22. - pp. 63-84.

Plamondon Réjean and Srihari Sargur N. On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey [Journal] // IEEE Trans. Pattern Anal. Mach. Intell.. - 2000. - 1 : Vol. 22. - pp. 63-84.

Roy Partha Pratim [et al.] HMM-based Indic handwritten word recognition using zone segmentation [Journal] // Pattern Recognition. - [s.l.] : Elsevier, 2016.

Sharma Anuj, Kumar Rajesh and Sharma RK Online Handwritten Gurmukhi Character Recognition Using Elastic Matching [Journal] // Image and Signal Processing, 2008. CISP '08. Congress on. - May 2008. - Vol. 2. - pp. 391 - 396.

Sharma Anuj, Kumar Rajesh and Sharma R. K. HMM-based online handwritten gurmukhi character recognition [Journal]. - [s.l.] : Machine Graphics & Vision International Journal, 2010. - 4 : Vol. 19. - pp. 439-449.

Sharma Dharamveer and Jhajj Puneet Recognition of Isolated Handwritten Characters in Gurmukhi Script [Journal] // International Journal of Computer Applications. - 2010. - 8 : Vol. 4. - pp. 9-17.

Shi Zhixin and Govindaraju Venu Segmentation and recognition of connected handwritten numeral strings [Journal] // Pattern Recognition. - 1997. - 9 : Vol. 30. - pp. 1501-1504.

Shin Jungpil On-line cursive hangul recognition that uses DP matching to detect key segmentation points [Journal] // Pattern Recognition. - [s.l.] : Elseviar Science Inc., November 2004. - 11 : Vol. 37. - pp. 2101-2112.

Tappert C. C. Cursive script recognition by elastic matching [Journal]. - [s.l.] : IBM Journal of Research and Development, 1982. - 6 : Vol. 26. - pp. 765-771.

Tappert Charles C., Suen Ching Y. and Wakahara Toru The State of the Art in On-Line Handwriting [Journal] // IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE. - 1990. - pp. 787-808.

Verma Karun and Sharma Rajendra Kumar Comparison of HMM- and SVM-based stroke classifiers for Gurmukhi script [Journal] // Neural Computing and Applications. - [s.l.] : Springer, 2016. - pp. 1-13.

Wang Da-Han, Liu Cheng-Lin and Zhou Xiang-Dong An approach for real-time recognition of online Chinese handwritten sentences [Journal] // Pattern Recognition. - October 2012. - 10 : Vol. 45. - pp. 3661-3675.

Yamada Hiromitsu, Yamamoto Kazuhiko and Saito Taiichi A nonlinear normalization method for handprinted kanji character recognition—line density equalization [Journal] // Pattern Recognition. - 1990. - 9 : Vol. 23. - pp. 1023-1029.

Zheng Yefeng, Li Huiping and Doermann D. Machine printed text and handwriting identification in noisy document images [Journal]. - [s.l.] : IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004. - 3 : Vol. 26.

PAPER PUBLICATION STATUS

Paper has been communicated to Mr. Karun Verma for correction to be further communicated to a publisher under the title of "*Segmentation of strokes for recognition of Gurmukhi script*".

PLAGIARISM REPORT

turnitin **Turnitin Originality Report**

Thesis801431014 by Navneet Kaleka

From Thesis (Faculty Accounts)

Processed on 10-Aug-2016 12:41 IST
ID: 694733660
Word Count: 11360

Similarity Index	Similarity by Source	
14%	Internet Sources:	10%
	Publications:	9%
	Student Papers:	0%