SEGMENTATION OF CEHARI+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

Computer Science and Engineering

Submitted By

Rajendra Singh Verma

(Roll No. 801432023)

Under the supervision of:

Mr. Karun Verma

Assistant Professor

(CSED)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

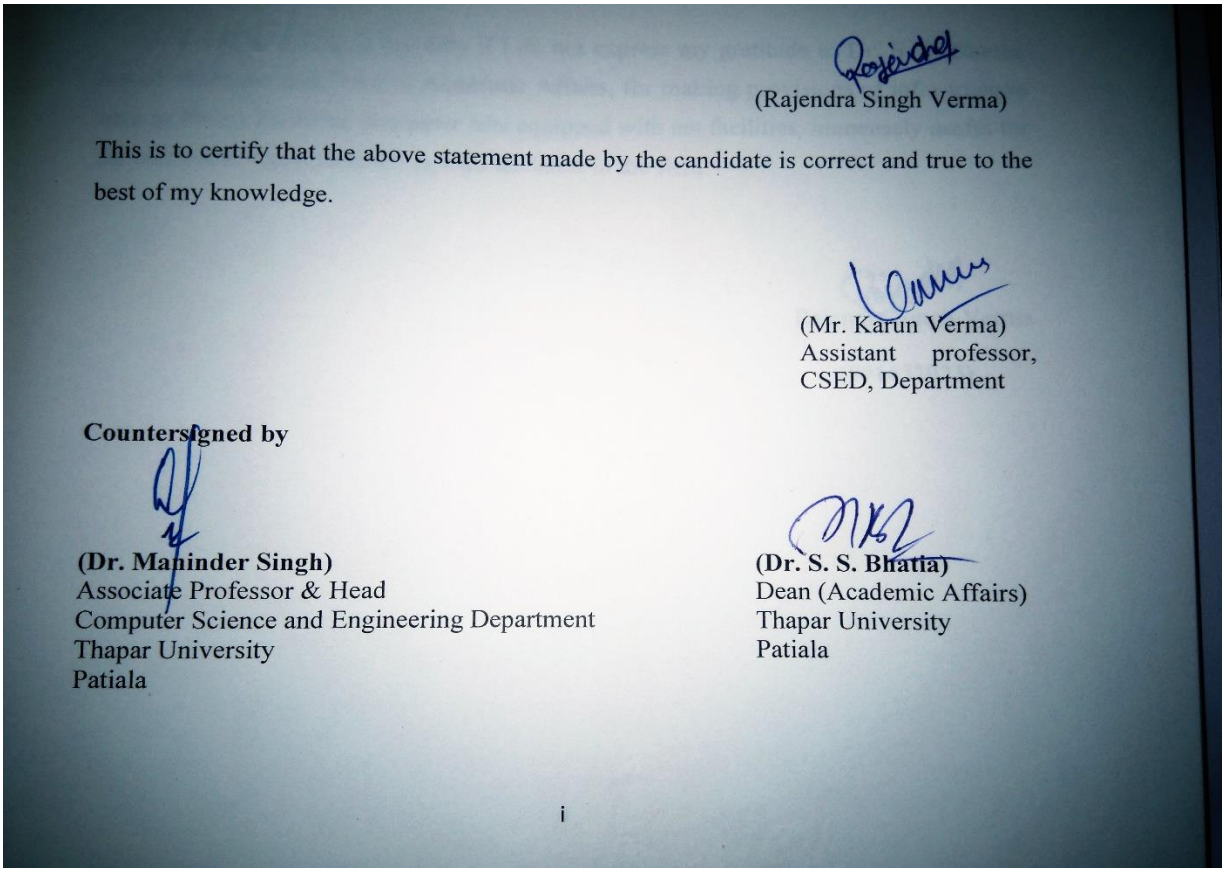
PATIALA – 147004

June 2016

Certificate

I hereby certify that the work which is being presented in the thesis entitled, **"***Segmentation of cehari+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 *Computer science and Engineering* submitted in 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.



(Rajendra Singh Verma)

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

(Mr. Karun Verma)
Assistant professor,
CSED, Department

Countersigned by

(Dr. Mahinder Singh)
Associate Professor & Head
Computer Science and Engineering Department
Thapar University
Patiala

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

i

Acknowledgement

First of all, I am extremely thankful to my respective guide **Mr. Karun Verma,** Assistant Professor Computer Science & Engineering Department of Thapar University for his valuable guidance, advice, motivation, encouragement, moral support, sincere effort and positive attitude with which he solved my queries and provide delightful ambiance for learning, exploring and making this thesis possible. It has been a great pleaser and experience to work under his sanctuary. I am also heartily thankful to **Dr. Deepak Garg**, Professor and **Dr. Maninder Singh,** Associate Professor and Head, Computer Science & Engineering Department and **Dr. Ashutosh Mishra**, PG coordinator, for motivation and providing uncanny guidance and support throughout the preparation of the thesis report.

I will be failing in my duty if I do not express my gratitude to **Dr. S. S. Bhatia**, Senior Professor and Dean of Academic Affairs, for making provisions of infrastructure such as library facilities, computer labs equipped with net facilities, immensely useful for the learners to equip themselves with the latest in the field.

Rajendra Singh Verma

(**801432023)**

Abstract

Online handwriting recognition has been worked upon for a long time now. It has been done for various scripts all over the world. Great milestones have been reached in research for the online handwriting recognition for Indian scripts as well. We 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 algorithms are proposed that use the slope calculation method at every point and find candidate points for segmenting the stroke into individual basic strokes. The candidate points is a segmenting points, more than one candidate points may be present in the text. The algorithms demonstrated an accuracy of 96.5% in segmenting various stroke combinations when written in a single stroke.

Table of Content

List of Figures

List of Tables

Chapter 1: Introduction

Nowadays, computers have a great influence on us and we perform all the essential work using computer. Everyone wants to complete our task as easily as possible. Exchange of information between computer and user is done by handwriting and is the easiest way of communication. In general, computer takes the input through keyboard or mouse, but mouse or keyboard have some limited functionality. Writing Indian scripts using keyboard and mouse is time consuming task because Indian scripts have large symbol sets and modifiers. But we can perform both input and output on the single compact device and other advantage that it is very easily to given an input by users. The data can be converted into characters or digits which can be entered the data at a fast rate without using keyboard and any additional devices. A natural way of entering text is by writing. Handwriting has continued to persist as a means of communication among humans [1]. Computer has become more and more widespread in recent years. Gurmukhi script have a various similarities to other Indian script such as Devnagari and Bangla [2]. Efforts are being made to make handwriting as an effective means of communication between man and machines. Machines have to be trained to recognize the handwriting of every individual.

Handwriting recognition of character is a system for convert the handwritten text into actual text. Segmentation of handwritten text is most important process for character recognition in which the stream of continuous handwritten characters are segmenting into more than one stroke for correctly identified by recognition system. Incorrect segmentation goes to incorrect feature extraction and incorrect recognition [3]. Isolating of characters and modifiers in the handwritten text of Gurmukhi script, the recognition efficiency largely depends on success of full segmentation of handwritten characters. Online Handwriting recognition (OHWR) for Indian scripts like Bangle, English and Chinese has been done [4].With the advancement in technology Handwriting Recognition for Indian languages has gain importance.

Handwriting recognition of Indian Scripts is a very difficult task because they consist of large symbol sets and there is inconsistency involved in writing styles of different person to person.

1.1 Property of Gurmukhi script

Gurmukhi script is used to write Punjabi language in India and all over the world which is the world's 10th most widely spoken language. More than 100 million people speak in Punjabi all over the world. Gurmukhi script is cursive and written in top-down approach and left to right direction [9]. The characters have a horizontal line on the upper part which is connected the characters. This is called headline (shirorekha). Gurmukhi script has similar feature to other Indian scripts as Bangla and Devnagari. The Gurmukhi alphabet has 9 vowels and 41 consonants. In addition to them, there are 3 special sound modifier symbols. Upper case and lower case characters concept is not present in this script. Below table 1.1 and 1.2 shows the characters and vowels Gurmukhi script. Combination of the consonant and the vowels produces different sound. For example dudda consonant and kanna matra gives the sound of da as in dada. Gurmukhi script is not case sensitivity.

Table 1.1 Character set of Gurmukhi script:

S. No.	Character	Character name	S. No.	Character	Character name
1	ੳ	URHA	2	ਅ	ERHA
3	ੲ	EERHI	4	ਸ	SUSSA (Sa)
5	ਹ	HAHA (Ha)	6	ਕ	KUKKA (Ka)
7	ਖ	KHUKHA (Kha)	8	ਗ	GUGGA (Ga)
9	ਘ	GHUGGA (Gha)	10	ਙ	UNGGA (Nga)

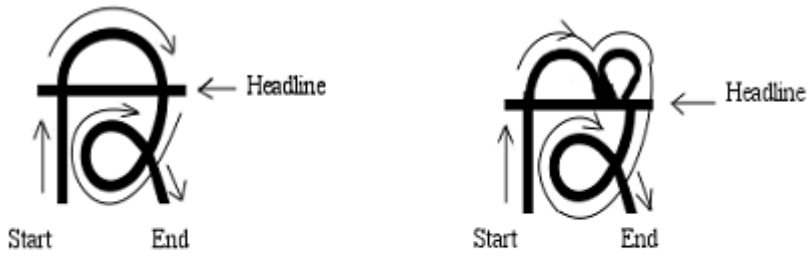
11	ਚ	CHUCHA (Ca)	12	ਛ	CHHUCHHA (Cha)
13	ਜ	JUJJA (Ja)	14	ਝ	JHUJHA (Jha)
15	ਞ	YANZA (Nya)	16	ਟ	TAINKA (Tta)
17	ਠ	THUTHA (Ttha)	18	ਡ	DUDDA (Dda)
19	ਢ	DHUDDA (Ddha)	20	ਣ	NAHNHA (Nna)
21	ਤ	TUTTA (Ta)	22	ਥ	THUTHA (Tha)
23	ਦ	DUDA (Da)	24	ਧ	DHUDDA (Dha
25	ਨ	NUNNA (Na)	26	ਪ	PUPPA (Pa)
27	ਫ	PHUPHA (Pha)	28	ਬ	BUBBA (Ba)
29	ਭ	BHUBBA (Bha)	30	ਮ	MUMMA (Ma)
31	ਯ	YAIYYA (Ya)	32	ਰ	RARA (Ra)
33	ਲ	LULLA (La)	34	ਵ	VAVA (Va)
35	ੜ	RARA (Rha)	36	ਸ਼	SUSSA PAIR BINDI (Sha)
37	ਜ਼	JAJJA PAIR BINDI (Za)	38	ਖ਼	KHUKHA PAIR BINDI (Khha)
39	ਫ਼	PHUPHA PAIR BINDI (Faa)	40	ਗ਼	GUGGA PAIR BINDI (Ghha)
41	ਲ਼	LALLA PAIR BINDI (Lla)			

Table 1.1

Table 1.2 Vowel set of Gurmukhi script:

S. No.	Vowel	Vowel name
1	◌ਾ	KANNA (aa)
2	◌ਿ	SIHARI (i)
3	◌ੀ	BIHARI (ii)
4	◌ੇ	LAVAN (ee)
5	◌ੈ	DULAVAN (ai)
6	◌ੁ	ONKAR (u)
7	◌ੂ	DULANKAR (uu)
8	◌ੋ	HORA (oo)
9	◌ੌ	KANAURA (au)

When writing two or more characters or a character and modifiers in a single stroke (a stroke is a collection of consecutive coordinate points from pen down to pen up) is one more problem for the online handwriting recognition is shown in Figure 1.1.



(a) **(b)**

Figure. 1.1 The arrows from start to end show one stroke, pen down to pen up. (a) Kukka consonant with *cehari* matra (b) Kukka with *cehari* and *tippi*.

Cehari in the Gurmukhi script is a vowel (*matra)* as "ਿ" shown in table 1.2. *Tippi* in Gurmukhi Scrip is a "ੰ" and character set of Gurmukhi script shown in Table 1.1. In Figure 1.1 the arrows display the direction of pen moment from pen down to pen up it is a single stroke. The problem comes because of the cursively nature of handwriting, when writer has written text very fast then complete a text by free hand. When finding newly identified classes which are not recognized by the character recognition so that newly identified class can be compared with the existing classes of handwritten character. Existing classes may be containing multiple variations of the single class (stroke). All the variation of any single stroke is represented by single identity which is called stroke ID.

1.1.1 Zone in Gurmukhi script

Gurmukhi word is divided into three zones, upper zone, middle zone, lower zone as display in Figure 1.2.

Upper Zone: It is the area above the topline (shirorekha), where vowels exist.

Middle Zone: In middle zone contain the area where the consonant and little part of vowels exist. *i.e.*, the portion below the headline and above the lower zone.

Lower Zone: It shows the area below the middle zone where halant or certain half character lie at the foot of the consonants and some vowels.

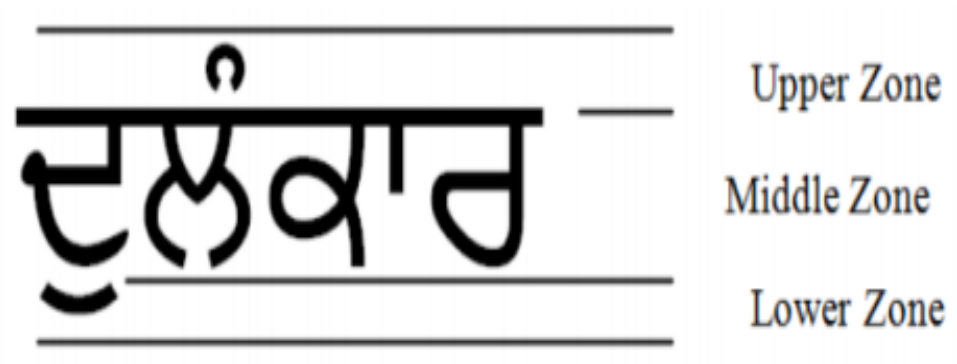


Figure 1.2 Zone of Gurmukhi script.

1.2 Classification of Handwriting Recognition

Character recognition field, it is an active research area nowadays. Handwritten character recognition system deals with the character recognition which written by users on an electronic pad or on a paper. The handwritten character recognition is mainly classified in two major categories: Offline character recognition and online character recognition show in Figure 1.3.

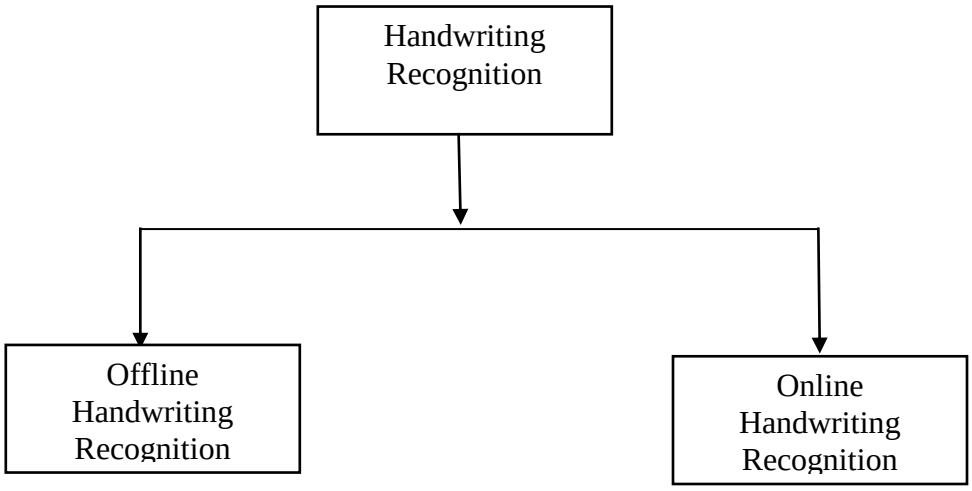


Figure 1.3 Classification of Handwriting Recognition.

1.2.1 Offline Handwriting Recognition

In offline handwritten recognition system, handwritten text or characters are available in the form of images which is captured with the help of a camera or a scanner, which capture the writing text by optionally. Offline handwritten recognition system is the process to converting offline Gurmukhi script into machine processable format [5]. In offline character recognition system, text is not recognize at the same time of text written.

In this process text written on the paper and provide as input to computer in the form of image. This is performed with the help of many processes: firstly we store image in form of grey scale format and further process done on it for better recognition accuracy. This type of recognition process is called Optical character recognition (OCR). This recognition system is not suitable for communication between men and machine because it is not real time activity. Offline character recognition system is more suitable for converting the paper document into electronic documents through computer. OCR can be used in various applications like data entry for document e.g. check, passport, Barcode, ID cards, bank statement and automatic number plate recognition etc.

Figure 1.4 Input sample for offline character recognition.

1.2.2 Online Handwriting Recognition

Online handwriting recognition (OHR) is real time activity, it is recognized text at the time of handwritten text produced by writer. Digital tablet is an electronic device which is used for writing and it uses a pen for writing on digital pad which is called as “Stylus”. Online handwriting recognition system will captured the two dimension coordinate points from writing pad on the direction of pen movement. Online handwriting recognition is collected a dynamic information for each stroke. Stroke is a collection of consecutive coordinate points from pen down to pen up. In this system characters are captured as a sequence of strokes. In case of OHR system text is correctly recognize because if any character is not properly recognized, then user can change writing style and again apply for recognition. Therefore segmentation and recognition are very easier in online handwriting recognition as compared to offline handwriting recognition. In Figure 1.5 shows the process of online handwriting recognition system.

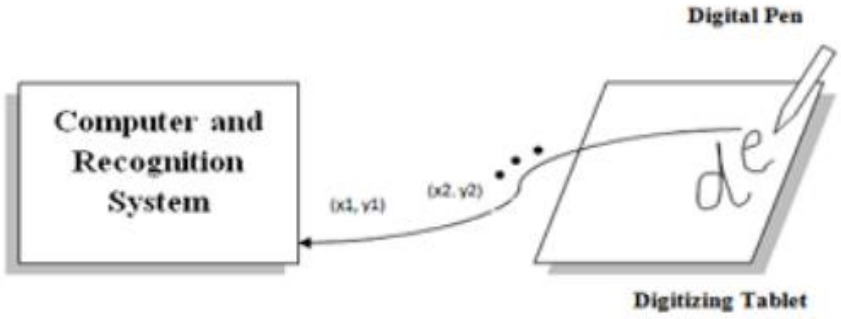


Figure 1.5 Online handwritten input sample.

Online handwritten recognition system perform all the processes which is shown in Figure 1.5. Data collection is a first phase of the handwriting recognition system in which we collected data in form of x-y coordinate points with the help of personal digital assistant, tablet PC and pen tablet. A usually pen perform to action Pen-Up and Pen-Down. The pen trace between Pen-Down and pen-Up is called a stroke [6].

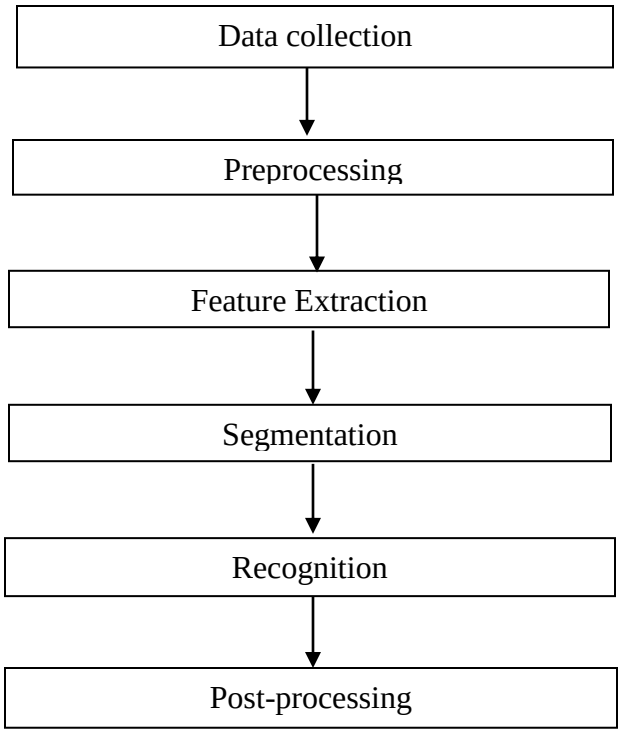


Figure 1.6 Phases of online recognition.

Preprocessing phase in online handwriting recognition system is use to eliminate distortion and noise present in the character due to software and hardware limitation. Preprocessing in online handwriting recognition system performing five common task size normalization, duplicate point removing, centring, missing points interpolation and resampling of points [7]. Feature extraction is important to classify correct feature. Feature extraction is important for recognition and segmentation. Segmentation is a phase of recognition system which divides the text into stroke level. Segmentation has two types: internal segmentation and external segmentation [4]. Recognition phase uses various classifier for recognize the stroke *i.e.* SVM, K-NN and HMM etc. Post-processing is processing to combine stroke after recognition and generate a complete meaning full word.

1.3 Handwritten character segmentation

Segmentation of handwritten text is most important process for character recognition. The accuracy of recognition system is depends on correct segmentation, in which the

stream of continuous handwritten text are divided into more than one stroke for correctly identified by recognition system.

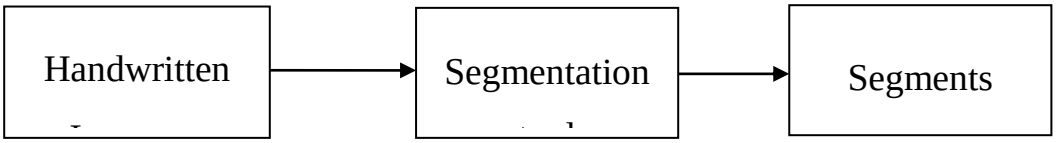


Figure1.7 Block diagram of segmentation.

In this thesis work, we have mainly attention on improving the accuracy of recognition system by improving the process of segmentation of handwritten Gurmukhi script [8].

Chapter 2: Review of Literature

The basic idea of doing this study is to discover the knowledge that how many various types of methods have been applied for the recognition and to see the recognition accuracy in percentage of their respective methods. This survey is done for both offline handwriting recognition and online handwriting recognition system.

Bhattacharya *et.al* [2] presented technique for online recognition and segmentation from Bangla text. Here, firstly a divide cursive text into strokes. Stroke represents a part of character. Stroke is collection of consecutive coordinate points capture from pen down to pen up which is written on digital pad. Unconstrained Bangla handwriting is usually cursive. After that researchers have selected set of Bengali text which contain basic characters, consonant and vowel. For text segmentation into stroke they discovered rules by analyzing various joining patterns for Bengali text. Here for segmentation offline and online information was used. This methods also deals with recognition of stroke segmentation in which directional feature are used.

Jindal *et.al* [9] suggested a method for segmentation of printed Gurmukhi words. They segmented the whole documents into strips and segmenting horizontally overlapping lines and their respective lines.

Lehal *et.al* [7] proposed a technique to encounter adverse situation of English and Gurmukhi script. In this paper they performed experiment on multi-fond and multi-sized character using Gradient feature and Gabor feature based on gradient information and directional frequency of an individual character to find it as Gurmukhi or English. In this paper they focused on single character script identification and presented SVM based method for the identification of multi-fund and multi-sized character.

Sharma *et.al* [6] Presents rearrangement system of recognized online handwritten Gurmukhi word. In this paper rearrangement includes stroke identification, stroke position and combination of the stroke for the recognition of character. In Gurmukhi

script having number of stroke after segmentation arrangement the stroke for completing the character.

Kumar and Sharma *et.al* [5] presented K-NN based classifier using transition feature and diagonal feature for offline Gurmukhi character. Proposed method firstly create skeleton of character for the extraction of feature information about the character. Feature of character have been calculated using distribution of point of character bitmap image. They also calculated reference and testing points to identified k-NN.

Fink et.al [4] suggested novel technique online Bangla handwriting recognition that includes cursively written words. In this paper proposed approach uses the concept of sub-stroke level for the feature represent the characters and Hidden Markov model based on the writing model which is applied for the online and offline text recognition. For building writing model researchers have consider various model structure that is major aspect in Hidden Markov model based methods.

Verma and Sharma [10] in this paper, proposed work for online system of handwriting recognition for Gurmukhi script, proposed technique to selected feature from handwritten text for recognition and applied SVM and HMM based stroke classifier model. Which is used a comparative based analysis for character recognition. In this work, they used voting-based classifier and they have attained a recognition rate of 96.7 % for character recognition of Gurmukhi script.

Plamondon *et.al* [3] suggested approach for offline and online handwriting whose meaning to expressing the thoughts and way of communication in the form of written like text. They have mentioned in this that when we try to take handwritten text on the recognition of machine of that time we try to explore the nature of handwritten text actually processed to how you can convert into electronic data. In this paper automated handwritten text recognition sees since long decades. Now these days clearly we are able to recognize the machine written language documents using lot of various software's. Out of these software's the most reliable software is off the shelf OCR software.

Vyas and Verma [11] suggested to segment the handwritten text into number of sub-strokes which is pass through recognition system. In this paper it is mentioned that there is not a single base where we can segment the all type of Gurmukhi script. In online recognition system take input from PDA, mobile and tablets, it is told that we cannot segment the Gurmukhi script on the basis of their shape instead of we divide the Gurmukhi script on the basis of coordinate points. They also told that if we divide on the basis of their shape there may be chance that it may create ambiguity so finally the authors have proposed that we draw the Gurmukhi script on the interface which is created by user and then system segments the Gurmukhi script. What it do that it tries to find the correctly segmented coordinate points. In these points there is no chance of an ambiguity and we can correctly recognize our Gurmukhi script.

Jibu *et.al* [12] in this paper deals with generally on the different segmentation methods existing for handwritten Indian script for recognition system. Image processing is very difficult task for Indian scripts, where image segmentation contain meaning full object. In this proposed work image is divided into homogeneous pixel groups, and segmentation methods can be divided as Region based and Edge based.

Djeddi *et.al* [13] suggested methods for text- independent recognition in multi script environment, in this study considered written text for English and Greek languages. Writing style is same of individual's writer. In this paper working on the short pieces handwritten text to recognize a given writer. In this paper proposed technique was based on a set off run length feature. K-NN and SVM classifier were used for classification. Suggested process to handwriting based writer identification. The writing style persists to same across the various scripts.

Roy *et.al* [14] suggested a novel method for Indian handwritten text by zone-wise information. It is a very complex task due to modifier, multi stroke character and touching to another character or shirorekha. Proposed approach for classification is a Support Vector machine and Hidden markov model to classify the stroke which is written by writers. Suggested to segmenting to handwritten text image horizontally into three zone like lower, upper, middle zone and the corresponding zones are recognized. In this

segmentation approach improve the accuracy of recognition system. Proposed technique firstly recognize the middle zone because middle zone contain most of the part of characters are in middle zone which is recognition by Hidden markov model, and other upper and lower zone are defined the boundary of the character. At the end of recognition all components are combined with the text to complete the final level of recognition. In this paper they proposed water reservoir-based property have used to increasing the boundary detection and zone segmentation. Suggested approach based on zone-wise recognition to increasing the accuracy of Indian script recognition. In zone wise segmentation system is reduce to possible number of combinations in character set.

Pal *et.al* [15] in this paper discussed different technique for offline character recognition for Indian script. They shows the comparative analysis of different methods which is previously done. They discussed novel technique for classification is support vector machine which is used for multi-class classification. In this paper they presented a review of optical character recognition work which is done in Indian scripts.

Plamondon *et.al* [16] suggested a new algorithm that estimate importance of each signature points for segmentation of cursive handwritten text. Presented algorithm create a centered vertex for each signature points with the help of its neighboring points. In this algorithm the first step is to create a function that weights each signature points according to its neighbor points and second step to find out the different local maxima of a created function in step one.

Saabne *et.al* [17] presented a real time method for segmenting Arabic handwritten script in which segmentation was performed at stroke level. Stroke induced by segmenting point was classified with help of K-NN classifier and classification data was stored in a scoring matrix and finally with the help of scoring matrix researcher was find out final segmentation points.

Chapter 3: Problem Statement

Online handwriting recognition has an area of active research since 1960s. Handwriting recognition of Indian Scripts is a difficult task because it consists of large symbol sets and inconsistency involved in the writing styles. Writing style is varied from person to person. When two or more characters or character + modifier are written in the single stroke shown in Figure 3.1. It is another problem for handwriting recognition. Many researchers have been done in offline segmentation of handwritten Gurmukhi script but in online segmentation of handwritten Gurmukhi script has little work is done. The purpose of this thesis to segment newly identifies classes which are not recognized by recognizer. Character segmentation is one of the most crucial phase for handwriting recognition. It is segment single stroke which is not recognized by recognizer into multiple strokes which stroke class is already studied.

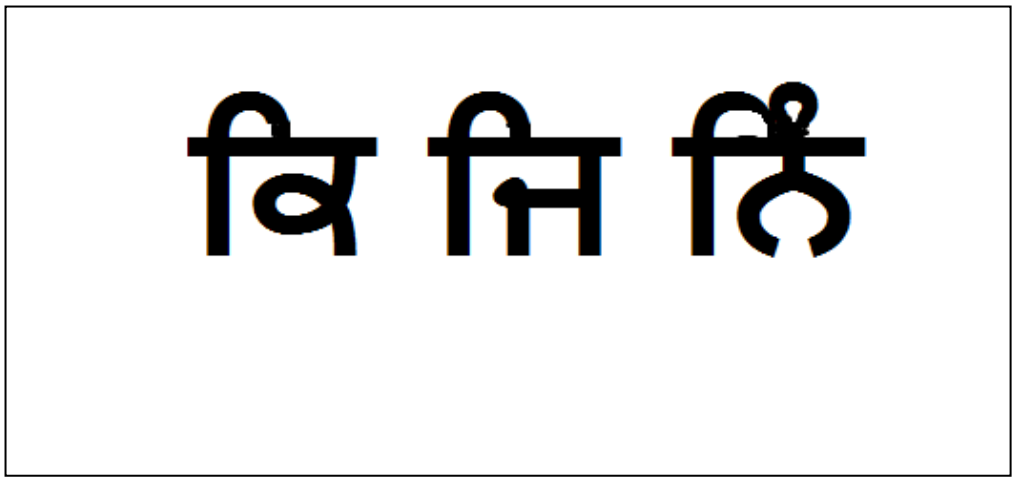


Figure 3.1 Multi stroke character of Gurmukhi script.

The aim of this thesis work is to segment words when written *cehari* and character in a single stroke which is not identify because newly identified classes is not exist in the data set. *Cehari* is a "matra"ਿ". So we segment the stroke in existing classes and identified it.

Developing the online segmentation system for handwritten Gurmukhi script is a greater challenge because following reasons.

- **Character sets are very large.**

In Gurmukhi script has a consonant (vianjan), vowel (matra), *bindi*, *tippi* and many more symbols are present.

- **Similarity between characters is high.**

Many similarities between the shapes of character so it generates many problems in segmentation of word and recognition to desired output. This vagueness generates a problem in the segmentation of Gurmukhi script.

Chapter 4: Data collection, Pre-processing, and Feature extraction

Data collection, Pre-processing and Feature extraction phases have been discussed in previous chapter 1. In this chapter we will broadly explain all the phases and algorithms which are used in this thesis work. In this section we elucidate the process of data collection of online handwritten Gurmukhi script. In section 4.1 explain the process of data collection, in section 4.2 explain the process of pre-processing and in section 4.3 explain the process of feature extraction.

4.1 Data collection

In this thesis study, we have collected 150 sample of handwritten Gurmukhi script. These 150 words cover all the basic characters, consonants and vowel modifiers. We have collected handwritten data in three categories from group of 40 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. We built an interface for collecting handwritten data by using java language. In the collected dataset the average length of word is four characters and average number of strokes per word is three and maximum is six. Hence the data set comprises of near about 4000 strokes. Input data set consists of co-ordinates along the trajectory of pen and the position of pen-down and pen up. 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. We are taking stream of continuously point of X-Y plan in form of coordinate points of the moving pen point. The example of a testing stroke is given in Figure 4.1. A stroke T = $\{(x_0, y_0), (x_1, y_1) \ldots\ldots\ldots (x_n, y_n)\}$ is shown the sequence of co-ordinate points of stroke. In Figure 4.1 shows the data generation for Gurmukhi script. In Figure 4.1 which (x_0, y_0) denotes the starting point of the stroke and (x_n, y_n) denotes the end point of the stroke. The stroke is a collection of the co-ordinate points which is generated from pen down to pen up.

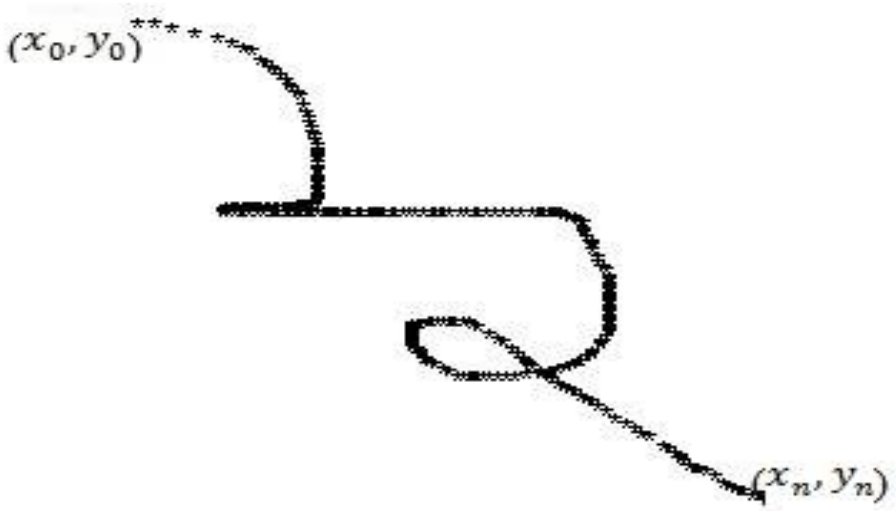


Figure 4.1 Stroke of Gurmukhi character.

In **Category 1** we have considered twenty writers and each writers is written ten times of each word of Gurmukhi script. Each writer is written Gurmukhi script in different-different writing style and in category 1 we have collected total number of word is two hundred and per words has an average number of strokes is six so total number of strokes is twelve hundred in category 1.

In **category 2** we have considered ten writers and each writers is written twenty times per word of Gurmukhi script. Each writer is written Gurmukhi script in varied writing style and in category 2 we have collect total number of word is two hundred and per words has an average number of strokes is six so total numbers of strokes is twelve hundred in category 2.

In **category 3** we have considered ten writers and each writers is written thirty times per word of Gurmukhi script. Each writer is written Gurmukhi script in varied writing style and in category 3 we have collect total number of word is three hundred and per words

has an average number of strokes is six so total number of strokes is eighteen hundred in category 3.

Table 4.1 Handwritten data collection.

Category	**Number of writers**	**Number of words written by each writer**	**Total number of words written by all writer**
Category 1	20	10	200
Category 2	10	20	200
Category 3	10	30	300
Total	**40**	**60**	**700**

A stroke can be defined the sequence of coordinate points between pen-down and pen-up position. In Gurmukhi script has also single stroke as well as multi stroke both type of characters is present. In Figure 4.2 we represent Gurmukhi character **'ਕ**'. We can discover that **'ਕ**' is a two stroke character. Stroke 1 is represent the character and stroke 2 is represent *shirorekha* (head bar). Writing speed is a big issue in recognition process as well as segmentation process.

If writing speed is too slow down then sampling rate and resolution should be very higher and writing speed is high then sampling rate and resolution should be lower. If writing

speed is very lower and very higher then it cannot appropriate for recognition process as

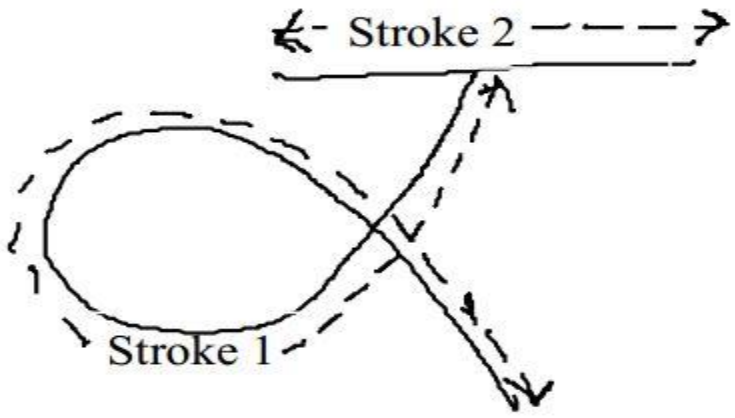


Figure 4.2 Character '**व**' written with two strokes.

Well as segmentation process. We select a suitable writing speed for sample generation which generates 64 samples (coordinate points) per stroke.

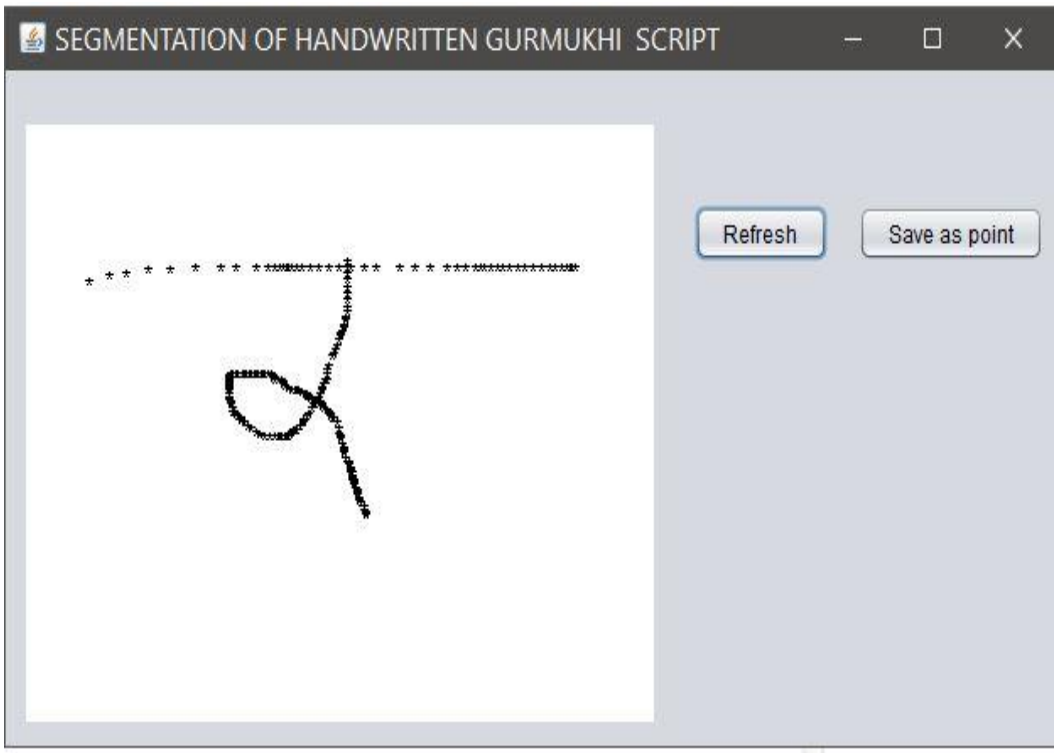


Figure 4.3 User interface to display pen movement.

If stroke contains 64 coordinate points then recognizer can easily detect the stroke. We have developed an online segmentation system for Gurmukhi script which provides user interface to display pen movement then capture coordinate points.

In Figure 4.3 shows the graphical interface for user. This software takes input from pen movement and stores the coordinate points in text file. Text file stores original coordinate points that are required at later for pre-processing, it is next step of segmentation process. Text file is help to identify the input stroke.

4.2 Pre-Processing

Pre-processing is the first stage of character recognition and segmentation of any languages. In online handwriting recognition and segmentation system this phase is used to removing noise existing in the input text due to software and hardware limitation like: different size of the stroke, centring of the stroke, missing point in the stroke due to character is hast pen movement.

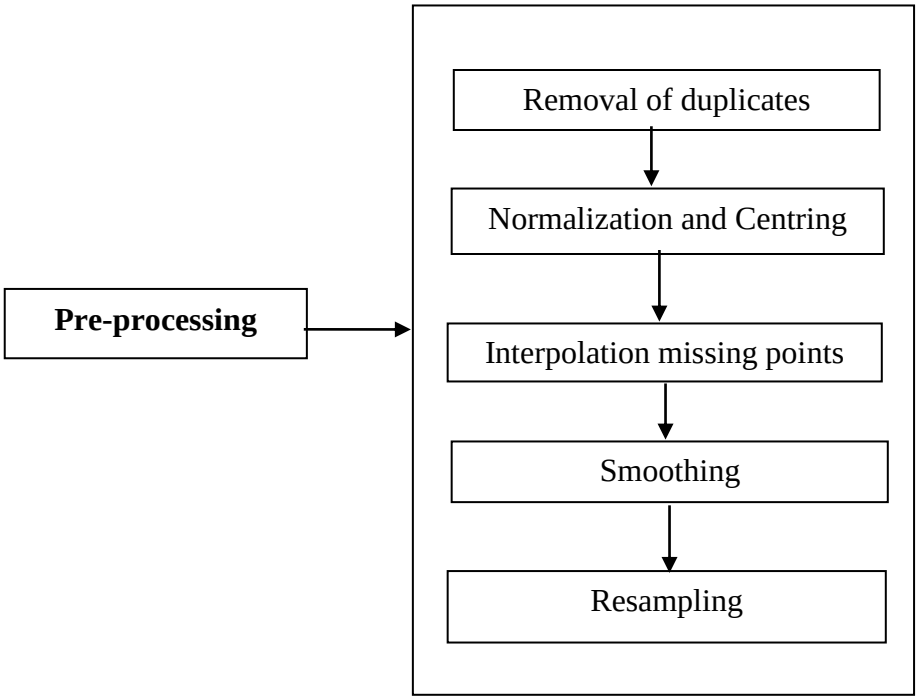


Figure 4.4 Phases of pre-processing.

We discussed in chapter 1, before extracting feature we applied preprocessing on input text for achieving higher accuracy for recognition as well as segmentation. In Figure 4.4 shows all the process of pre-processing phase for Gurmukhi script like removal of duplicate point, normalization and centring, interpolation missing point, smoothing and Resampling.

4.2.1 Removal of duplicate point

When writing the stroke then some time coordinate points are coincided to other coordinate points due to shape of the stroke. Duplicate point shows in Figure 4.5. Duplicate points are reducing the accuracy of recognition. When finding duplicate point then one of them is removed. In Algorithm 4.1 used for removing duplicate points.

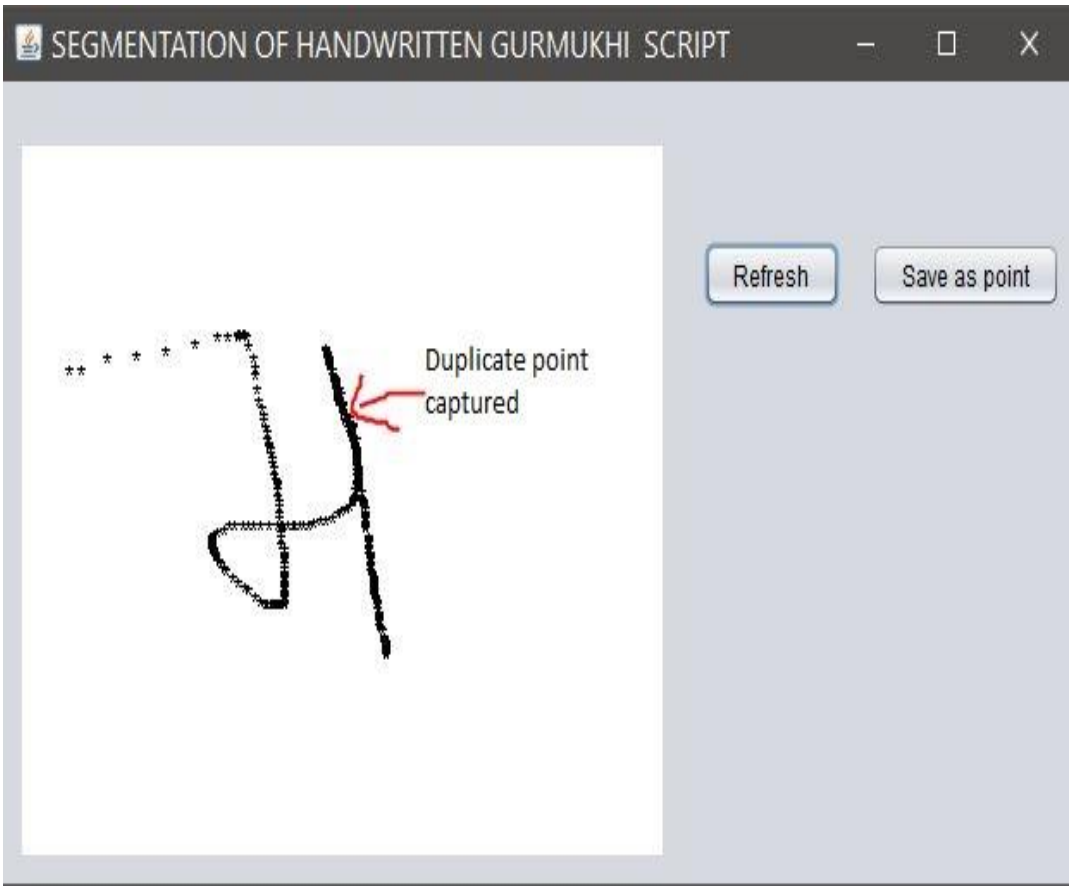


Figure 4.5 A stroke showing duplicate points.

Algorithm 4.1 Duplicate points removing:

1. p[]=points captured in array.
2. i= total number of points captured.
3. Set j=0 and v=0.
4. Repeat until j<i and increment j by 1.
5. Set v=j+1 and repeat until v<i and increment v by 1.
6. If(p[j].x==p[v].x and p[j].y==p[v].y)
7. Delete p[v].x and p[v].y from array.
8. End if.

4.2.2 Normalization and Centring

Normalization is one of the most important phase of preprocessing which applied on the input stroke or character because the size of every stroke is not same. Size of stroke depends on writer to writer, some writer writes small and some writes large size of character. So we applied normalization on input stroke to generate the fixed size of every stroke. Figure 4.6 shows the character is normalized in size but not putting in the center of the frame. We implemented the algorithm for normalization process. We put every character in the fixed frame which size is 300x300 pixels. In centring put the text at a constant distance from origin. In Figure 4.6 we have written small character in first frame and in the second frame shown the normalized character. If input stroke is written in small size then applied normalization algorithm to increase the size of stroke and in other hand if input stroke is written in larger then this algorithm reducing the size of stroke. This algorithm provides a stroke of uniform size. After normalization apply algorithm of centring put the character at a constant distance from origin. In Figure 5.7 written small character in first frame and in the second frame show the normalized and centring character.

Algorithm 4.2 Stroke Normalization and Centring:

1. p[]=points captured in array.
2. i= total number of points captured.
3. Set j=0;

4. Set x-scale=300 and y-scale=300 (for frame size is 300*300).
5. Find maximum and minimum coordinate of X and Y plan is xmax, ymax, xmin and ymin.
6. Compute Sx=x-scale/(xmax - xmin).
7. Compute Sy=y-scale/(ymax - ymin).
8. Choose scale=min (Sx, Sy) for scaling.
9. Repeat until j<i and increment j by 1.
10. P[j].X * scale.
11. P[j].Y * scale.

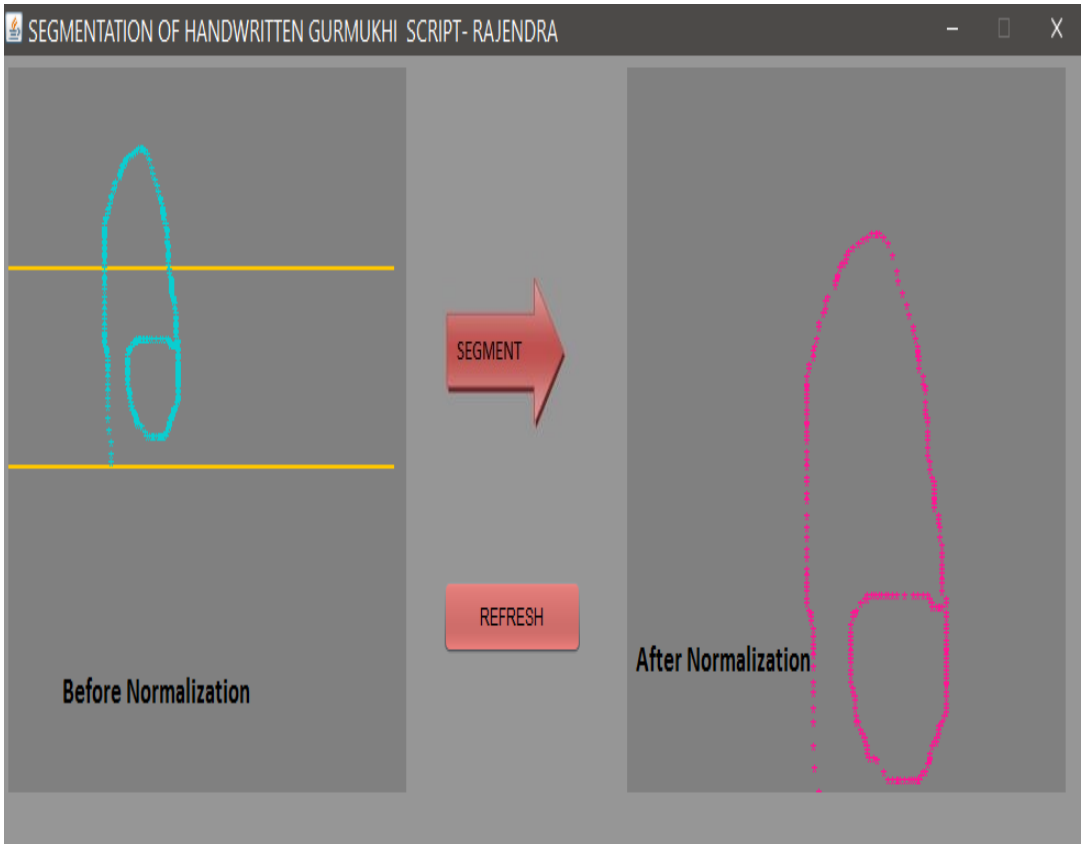


Figure 4.6 Handwritten strokes before and after normalization of Gurmukhi character 'ਜ਼' (*Rara+cehari*).

Centring:

1. p[]=points captured in array.
2. i= total number of points captured.
3. Set r=0;
4. Find minimum of x and coordinate is xmin, ymin.
5. Repeat step until r<i and increment r by 1.
6. P[r].x = p[r] – xmin.
7. P[r].y = p[r].y – ymin.

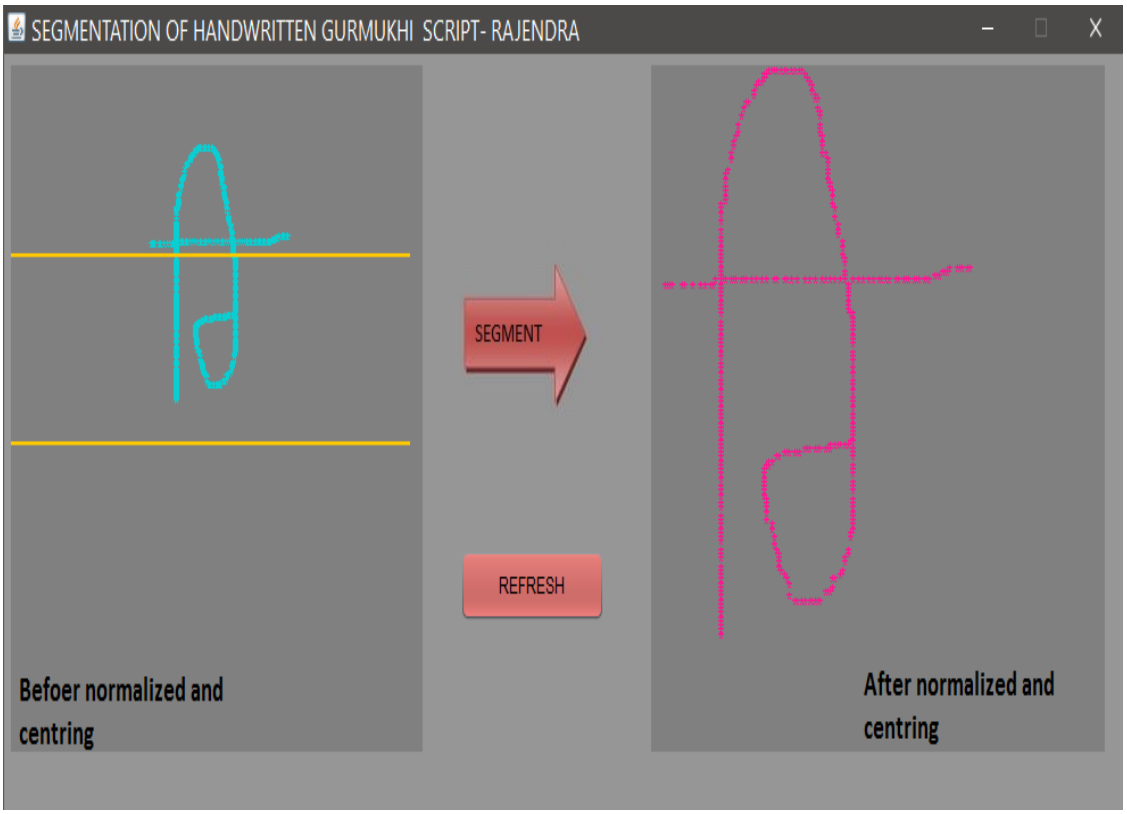


Figure 4.7 Handwritten strokes before and after normalization and centring of Gurmukhi character 'ਹਿ' (*Rara+cehari*).

4.2.3 Interpolation missing point

The missing point interpolation is one of the pre-processing phases, which is applied before recognition as well as segmentation. Point missing depends on writing speed of character if draw character with slowly then this interface capture continuously points Which easily detect but arising problem when draws character with high speed, missing some point due to hardware problem as well as software problem due to interface algorithm cannot capture a high speed writing input coordinates. In Figure 4.8 first frame shows the stroke without interpolation, this stroke shows some point missing due to writing speed, and in second frame shows the stroke after interpolation which shows continuously coordinate points without missing point. We use methods for replacing missing point to estimated point, the interpolation methods garneting estimating point for missing value. Interpolation is a technique of creating new data point with the range of known data set of point.

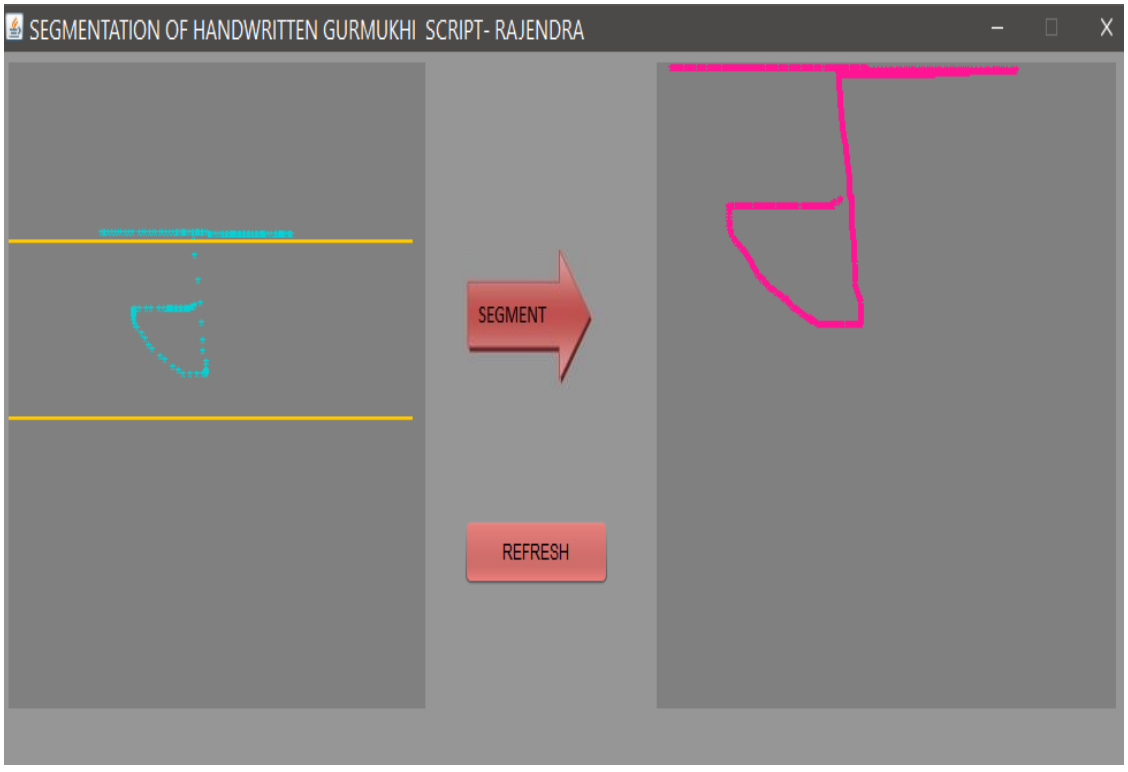


Figure 4.8 Handwritten strokes before and after interpolated of Gurmukhi character 'ਰ' (*Rara*).

Missing points are interpolated by various methods such as B-Spline curve interpolation, Bezier curve interpolation and Linear interpolation methods. The linear interpolation method requires two points first is starting point and second is end point and this algorithm putting estimated point between them. It is calculated estimating points easy and quick. In linear interpolation points are joined by straight line. But it is not accurate. In this thesis work we use cubic Bezier curve interpolation methods.

The Bezier curve is an approximating curve and Bezier curve is used to getting missing points. Bezier curve is not passing throw all control points. Bezier curve takes four successive points these points are considered to calculate the curve and this four consecutive points are known as control points. The next four control points provide the next Bezier curve and so on. The cubic Bezier curve interpolation method is used between these points which having the distance more than one. This algorithm finds these consecutive points which having the distance more than one then calling the Bezier. In Figure 4.9 shown these three different columns. Where first column is displayed the input stroke, the second column displayed the stroke after the normalization and in third column displayed the stroke after interpolation.

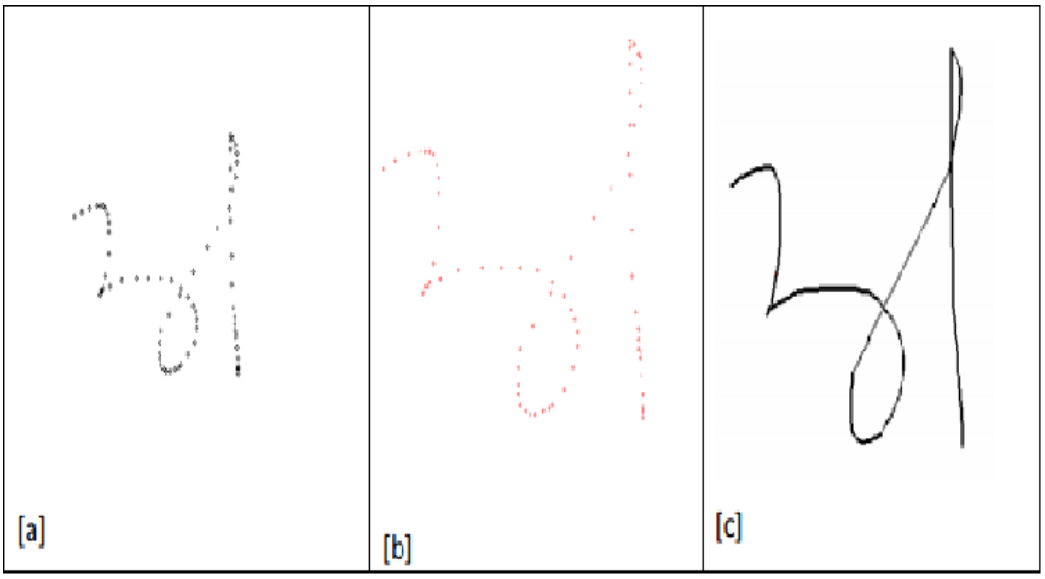


Figure 4.9 A stroke of Gurmukhi script displaying (a) input stroke (b) Normalized stroke (c) Interpolated stroke.

Algorithm 4.3 Missing point interpolation:

1. p[]=points captured in array.
2. i= total number of points captured.
3. c = creating empty list for storing the new generating points.
4. Set l = 0, x-diff = 0 and y-diff = 0.
5. u is a variable exist between 0<= u <=1.
6. Set u=.1.
7. Repeat step 8 to 19 until l< i-3.
8. x-diff= pow(p[l+1].x - p[l].x , 2).
9. y-diff= pow(p[l+1].y - p[l].y , 2).
10. diff = sqrt(x-diff+y-diff).
11. If(diff>1)
12. Repeat step 13 to 16 until u<=1.
13. x= $(1-u)^3$ * p[l+0].x + 3 * $(1-u)^2$ * u * p[l+1].x + 3*(1-u)* u^2* p[l+2].x + u^3*p[l+3].x
14. y= $(1-u)^3$ * p[l+0].y + 3 * $(1-u)^2$ * u * p[l+1].y + 3*(1-u)* u^2* p[l+2].y + u^3*p[l+3].y
15. u=u+.1.
16. Update list c by calculated the new points as the consecutive points.
17. Increment l by 1.
18. End if.
19. Else
20. Increment l by 1.
21. End else.
22. Exist.

4.2.4 Smoothing

When the data acquisition some noise is introduced due to tremor normally present in hand movement hardware and software shown in Figure 4.10. The normalized characters are yet piecewise linear and each points of a stroke is connected to neighbor points with line, then which is not shown smoothly so we apply a filter on the points and generated a

smooth curve between the points. Applied filter on x-axis as well as y-axis and draw curve between points of the stroke.

4.2.5 Resampling

Resampling is mathematical technique used to generate a new version of character, which having the points at equal distance to neighbor points and Resampling of coordinate points is performed to fix number of points in the input handwritten stroke. Resampling also hold the original form of stroke. In Resampling method applied a filter to select only fix number of coordinate points. If we have Resampling points are 64 then obtaining the best result according to literature survey. We have selected the Resampling points from character then the shape of character will not change. After applying the missing points interpolation algorithm we get points which having the distance between two points less one and half, removing of points will having two option: removing of those points will have distance less than one or points will remove at constant distance, i.e., two, three or four and so on.

In Algorithm 4.4 it is shown that the process of Resampling and to acquire a sampled 64 points in output. Each stroke is containing 64 of x coordinates and 64 of y coordinates. In Figure 4.10 shows the Gurmukhi character **'ੜ'** (*Rara*).In first frame shows the input handwritten character and in second frame displays character after pre-processing. Pre-processing is all most done after resampling.

Algorithm 4.4 Resampling:

1. p[]=points captured in array.
2. Create empty R[] list.
3. i= total number of points after missing points interpolation.
4. For getting the 64 points calculate a=int(i*1.0)/64.
5. Set b=a+1.
6. rx=((j-1)-(64*3))/(a-b).
7. ry=64-rx.
8. rx+ry =64 points after Resampling.
9. Set M=0, j=0, ct=0.

10. If(rx>0)
11. Repeat until j<rx and increment j by 1.
12. Assign value from p[M].x and P[M].y to R[ct].x and R[ct].y.
13. M=M+a.
14. ct=ct+1.
15. End if.
16. If(ry>0)
17. Set k=0 and Repeat until k<ry and increment k by 1.
18. Assign value from p[M].x and P[M].y to R[ct].x and R[ct].y.
19. M=M+a.
20. ct=ct+1.
21. End if.

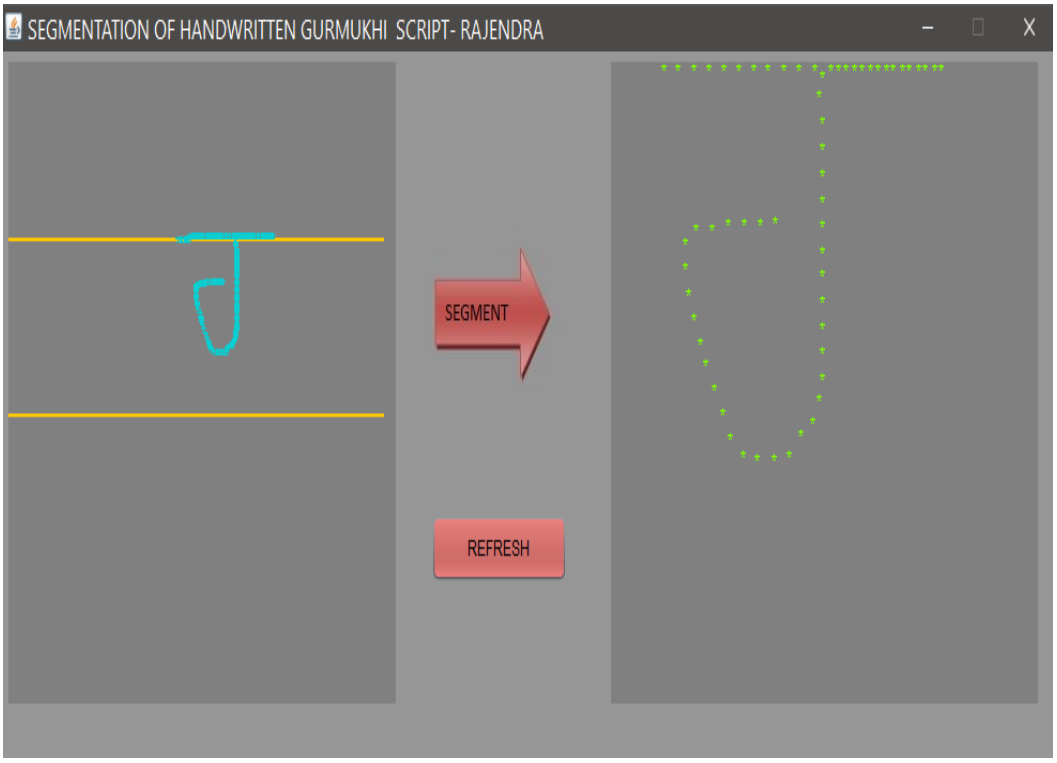


Figure 4.10 Handwritten stroke before and after Resampling of Gurmukhi character 'ੜ' (*Rara*)'.

In Figure 4.11 shows all the steps of preprocessing. Preprocessing is a first step of the online handwritten character segmentation as well as recognition, after in this step process the coordinate points and generating the standard points which easily used in further process.

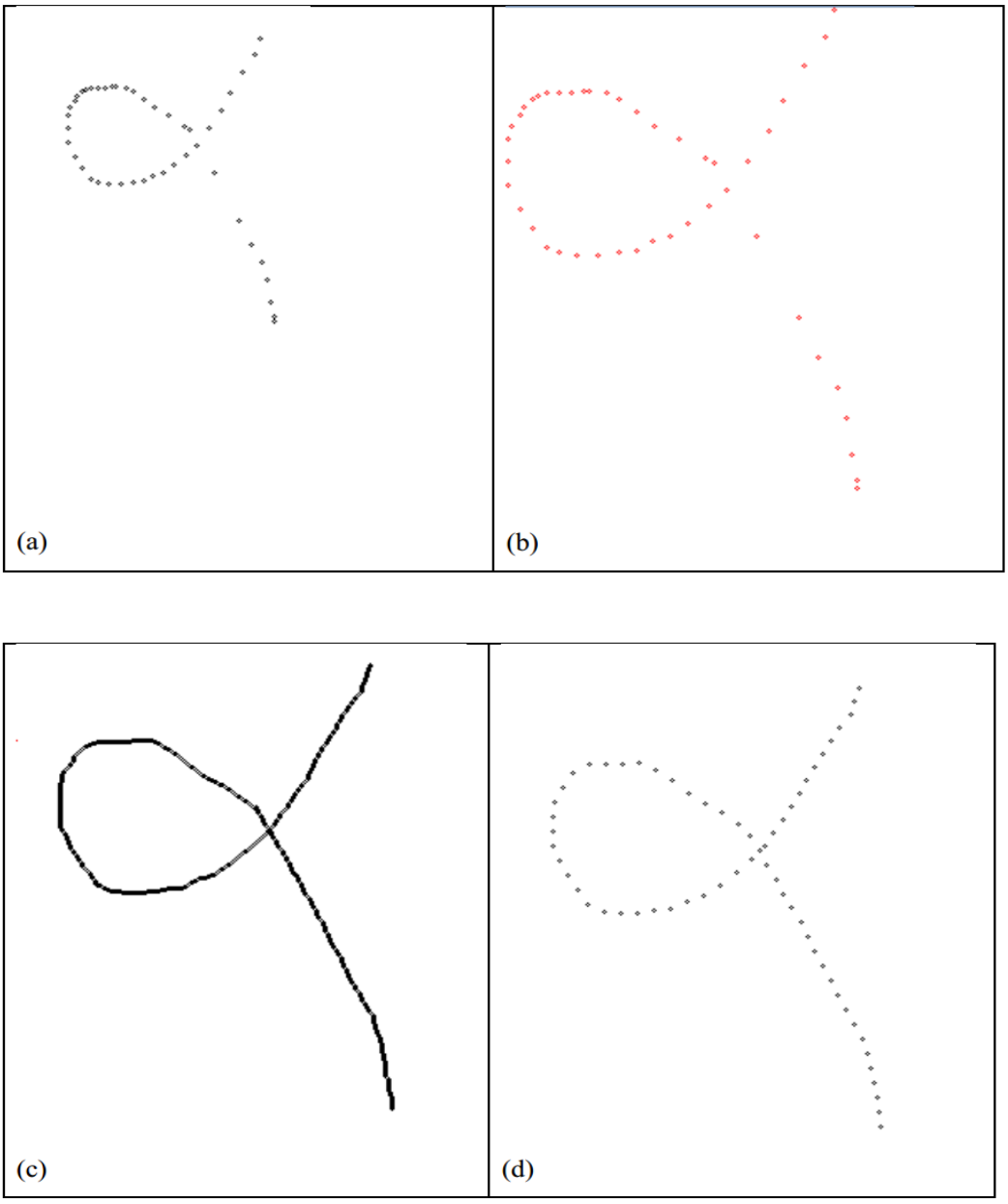


Figure 4.11 Display the entire process (a) input handwritten character (b) Normalized and centring (c) Missing points interpolation (d) Resample of character.

4.3 Feature extraction

Feature extraction from processed online handwritten character input data points are playing the most significant role in online handwritten character recognition as well as segmentation. The step is feature extracting from handwritten Gurmukhi character and to separate it from other character which is already stored in the library. The continuous coordinate points represent the shape or structure of character [18].

The sequence of coordinate points is affected by varied in the writing style. In feature extraction various feature used for handwritten character segmentation as well as recognition. Features are included number of coordinates points and angle between adjacent points. The accuracy of character recognition and segmentation is highly depends on the feature extraction. Feature extraction is the heart of handwritten character recognition as well as segmentation system, it is most important factor is achieving high recognition or segmentation performance rate of handwritten characters. We capture the coordinate points and apply pre-processing on original points and generate processed coordinates points which used for feature extraction [19]. After pre-processing phase we use these points for feature extraction. Each stroke will have a constant number of points and each point is having equal distance from its neighbor points. In this thesis work we proposed 64 coordinate points for each stroke and Distance between its neighbors points are like one or two and so on. In Figure 4.10 containing 64 coordinate points of character **‘ਰ’** (*Rara*) is.

Chapter 5: Segmentation

Segmentation is most important stage of online or offline handwritten character recognition. Recognition of character is massively depends on the proper segmentation. Segmentation can be applied after and before pre-processing [20] .Segmentation is classified into two type's i.e. external segmentation and internal segmentation. When segmentation is applied before the recognition phase is known as external segmentation (pure segmentation) and if when the segmentation is applied concurrent to the recognition phase is known as internal segmentation (recognition based segmentation) [21]. In Figure 5.1 shows the types of segmentation.

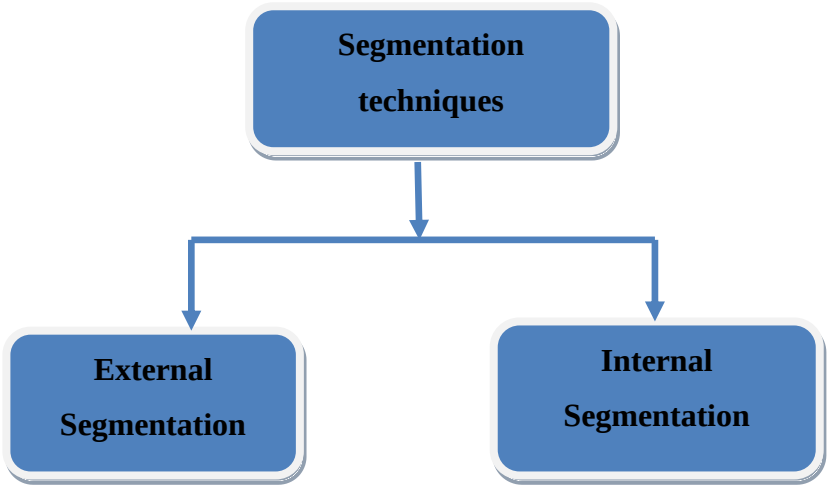


Figure 5.1 Type of segmentation

Handwritten character in Gurmukhi script is prone to problems of two or more characters or character + modifier are written in the single stroke. Many researchers have been done in offline segmentation of handwritten Gurmukhi script but in online segmentation of handwritten Gurmukhi script has little work is done. The purpose of this thesis project to segment newly identifies classes which are not recognized by recognizer. The character segmentation is one of the most crucial phase for handwriting recognition [22].

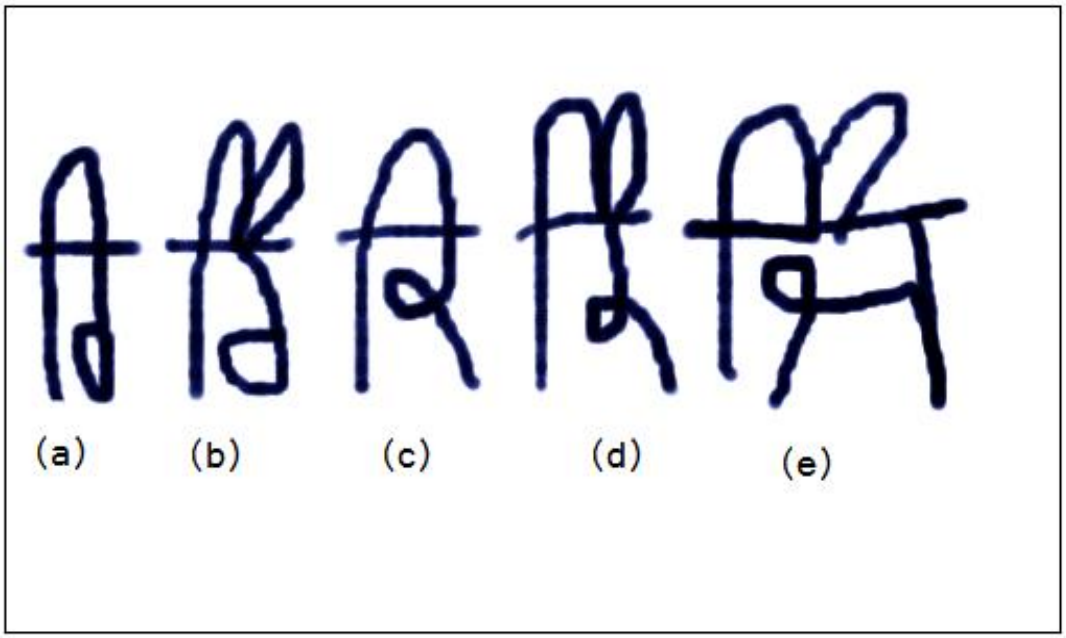


Figure 5.2 Multi stroke character of Gurmukhi script

Many problems are existing in cursive handwriting recognition of Gurmukhi script. So We select some problem of them shows in Figure5.2, it shows two different problem in (a) and (c) have a same problem this stroke is contain one character and one modifier like (a) contain "*rara + cehari*" and (b),(d) and (e) have a same problem contain one character one modifier and *tippi* like (d) contain "*kakka + cehari + tippi*". In this thesis work implement an interface for stroke segmentation of Gurmukhi script. We implement the software using java language and mention some assumption the headline (*shirorekha*) is already given and when writing a stroke, the collected points must be at least 64, and then only the stroke is considered. We implement an external segmentation for segment the stroke, segment stroke before recognition phase. After segmentation coordinate points applied for further process. Proposed method for character segmentation we introduce an approach using calculate the angle between adjacent points in degree and angle existing between +90 to + 60 when pen movement upward and angle existing between -90 to -60 when pen movement downward. We take some variation in angle due to prone of hand shiver at time of writing.

Angle=Math.toDegrees(Math.atan((y-axis3-y-axis1)/(x-axis3-x-axis1))); (1)

Equation 1 is used to calculate the angle between points, in this implementation we take the angle between first and third points (y1, y3) [23] . We try different combination of points but find higher accuracy between first and third points [24]. If points are passing throw shirorekha with negative angle those points known as candidate points or segment point [4]. We segment the stroke at the position of candidate point which shows in Figure 5.3. In a single stroke may be contain one or more candidate point in Figure 5.3(a) is contain two candidate point first separate "*cehari and tippi*" and second separate "*tippi* and *kakka*". In Figure 5.3(b) contain only one candidate point which separate "*cehari* and *rara".*

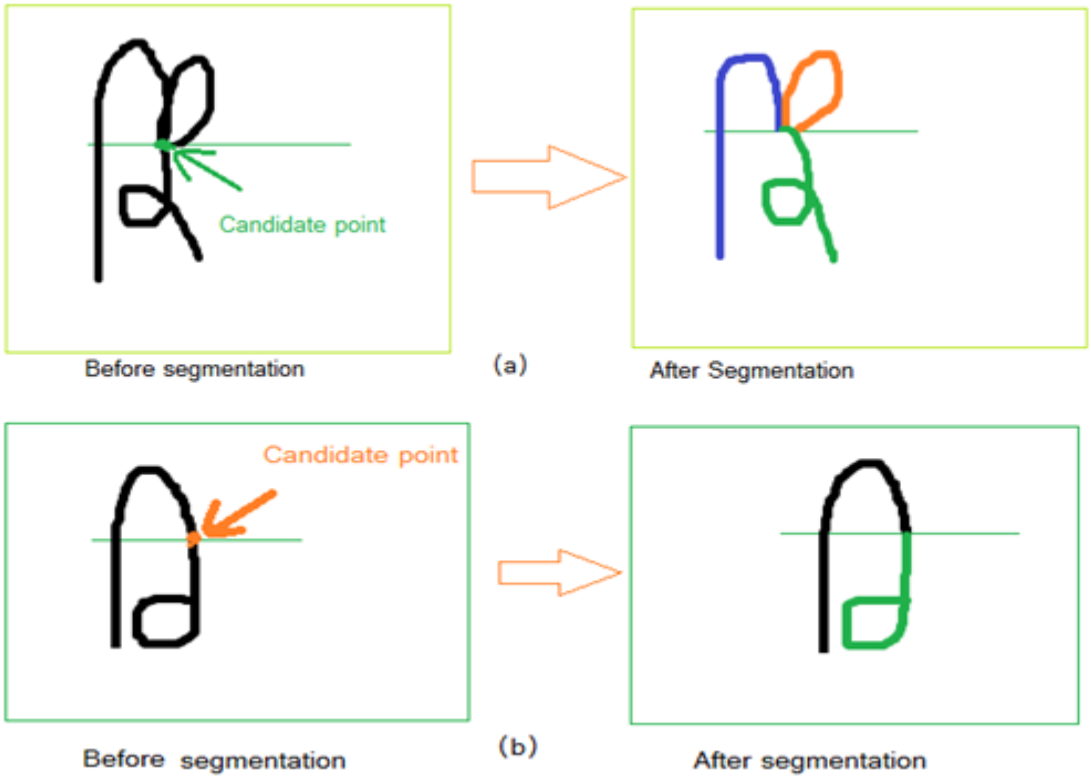


Figure 5.3 shows (a) and (b) both contain first frame is input stroke and second is segmented stroke.

Algorithm 5.1:

Problem Description1(*cehari* + *consonant*) and (*cehari* + *tippi* + *consonant*): Sometimes while writing Gurmukhi online writer writes cursively and while doing so writer may write more than one characters i.e. consonants or matras in single stroke as

shown in Figure 5.3. This is recognized as newly identified class but in actual it is nothing but combination of already existing classes. So we need to find a method to segment these newly identified strokes into sub-strokes so that they can be classified into existing classes. Hence we propose the following algorithm to segment the stroke which is a combination of *cehari + Consonant* and *cehari + Tippi + Consonant***:**

1. p[]=points captured in array.
2. i= total number of points captured.
3. Assumption= top line and bottom-line is known.
4. Set t=0; k=0;
5. If the starting point have the x, y coordinates with minimum values in arrays and have positive slope.
6. Repeat until t<i and increment t by 1.
7. set angle= FindAngle(p[t].x, p[t].y,p[t+3].x,p[t+3].y);
8. if angle is positive and not equal to zero and lies between 60 and 90 degrees.
9. Continue.
10. Else if angle is negative and not equal to zero and lies between 60 and 90 degrees.
11. Continue.
12. Stroke is segmented at the point where it touches the top line with negative angle.
13. If (after touching the top line the angle is positive and is above top line)
14. Segment the stroke where it again touches the top line with negative angle.
15. Else end.

Figure 5.4 shows the character with different combination of *cehari, tippi* and *consonant* which is correctly segmented by our interface. In this proposed method we used the

concept of external segmentation for character segmentation which segment character before recognition.

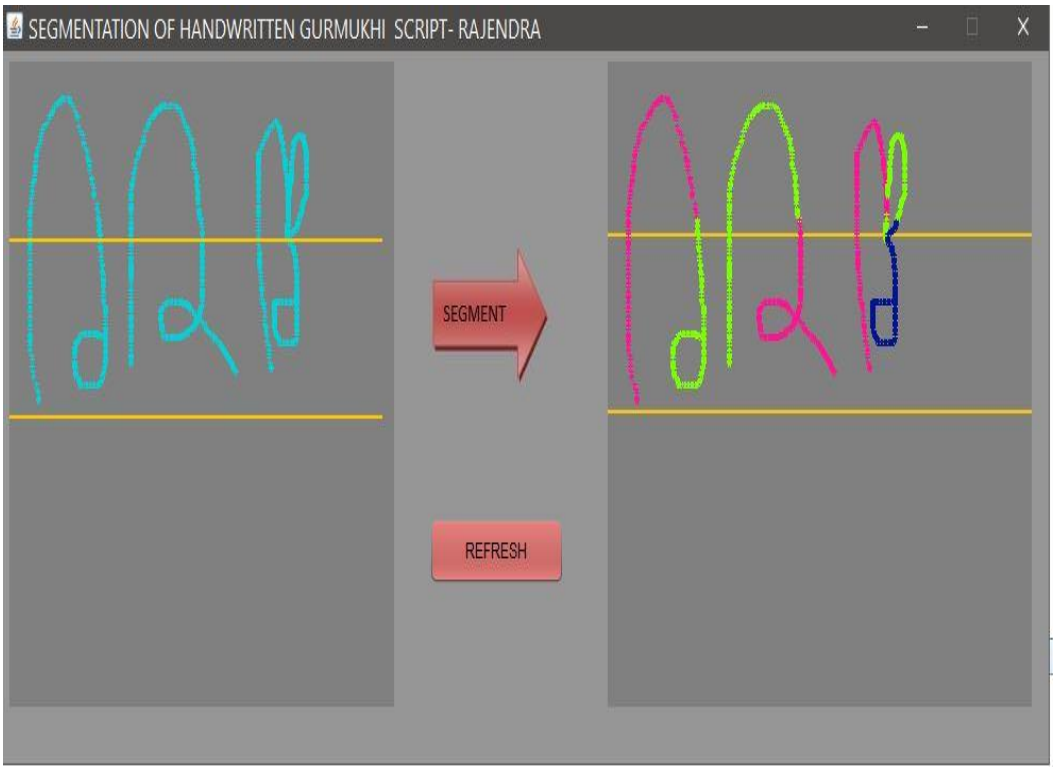


Figure 5.4 correctly segmented characters.

Chapter 6: Experimental Results

This algorithm has shown an increment for recognition of handwritten text of Gurmukhi script. For testing the algorithms for segment the strokes into sub- strokes. We take a set of 400 handwritten Gurmukhi words which was collected from people belong to different background by using mouse on the application developed. This character set contain cursive text. The results in detail are given in Table 6.1. It was noted that 96.5% of the total strokes were segmented properly and 3.5% error was occurred.

Table 6.1 Results of segmentation strokes by different writers

Writer ID	Number of strokes written	Correctly segmented	Incorrectly segmented	% of Accuracy
1.	40	37	3	92.5
2.	40	40	0	100
3.	40	38	2	95
4.	40	37	3	92.5
5.	40	39	1	97.5
6.	40	38	2	95
7.	40	38	2	95
8.	40	40	0	100
9.	40	39	1	97.5
10.	40	40	0	100

Table 6.2 Results for the total strokes segmented.

Total number of strokes	Correctly segmented	Incorrectly segmented	Total% Accuracy	% error Rate
400	386	14	96.5	3.5

Most of the error comes due to difference of angles between adjacent coordinate points in jointly consonant and vowels. Figure 6.1 has some text which is segmented correctly.

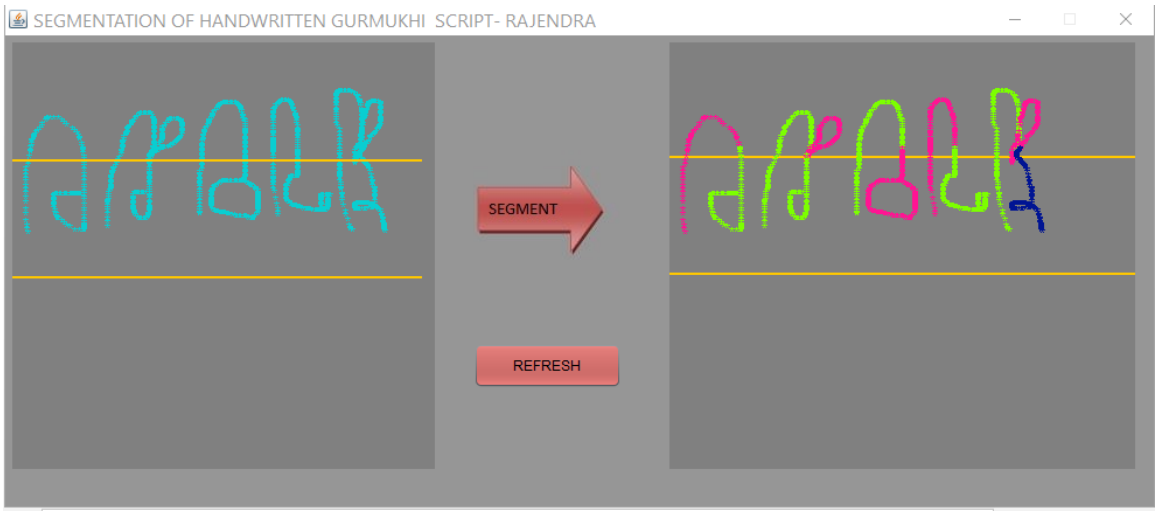


Figure 6.1 Correctly segmented of Gurmukhi word.

Shows in Figure 6.2 has some example of text which is not correctly segmented due to some variation in angles we mention a some range of angle if angle is not exist between then which is not segmented correctly and one more problem arise here that modifier is not connected to shirorekha which is not segmented, because stroke connecting to shirorekha is a segmenting point of the word.

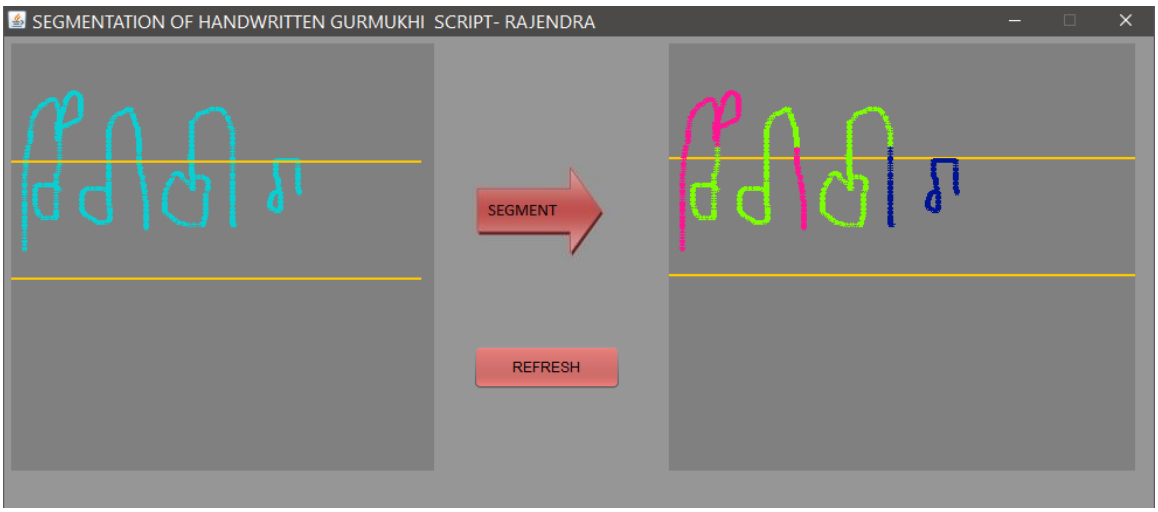


Figure 6.2 Not correctly segmented of Gurmukhi word.

Chapter 7: Conclusion and Future work

7.1 Conclusion:

In this research work, we have proposed a novel approach for segmentation of online handwritten Gurmukhi script. We are using the external segmentation method where the segmentation is done before the recognition of the stroke. 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 data points can be used in character recognition. An algorithms is based on the slope calculation between two consecutive points in the stroke. The method to calculate the positive and negative slope between coordinate points and with the negative angle touch to shirorekha this point is called candidate points. Then the candidate points are found on the basis of negative slope which is connected to shirorekha. In this algorithm we may be obtaining more than one candidate points. 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 algorithm are robust in segmenting the combinations of characters present in a single stroke.

7.2 Future work:

Our proposed approach has given good results. Apart from that, certain limitations like the strokes with *bihaari* [10] [25] *matra* were not segmented. In future work, the strokes which consist of the characters in the lower zone for example, *pairi haahaa* and *pairi rara* can be taken for segmentation. Various new classes can be found and various new combinations of strokes can be segmented.

References

[1] J. Hu., M. K. Brown and W. Turin, "HMM Based On-Line handwriting recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, pp. 1039--1045, 1996.

[2] N. Bhattacharya and U. Pal, "Stroke Segmentation and Recognition from Bangla Online Handwritten Text," in *International Conference on Frontiers in Handwriting Recognition*, 2012.

[3] R. Plamondon and S. N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 63-84, 2000.

[4] G. A. Fink, S. Vajda, U. Bhattacharya, S. K. Parui and B. B. Chaudhuri, "Online Bangla word recognition using sub-stroke level features and hidden Markov models," in *2010 International Conference on Frontiers in Handwriting Recognition (ICFHR)*.

[5] M. Kumar, M. k. Jindal and R. K. Sharma, "k-nearest neighbor based offline handwritten Gurmukhi character recognition," in *Image Information Processing (ICIIP), International Conference on*, 2011.

[6] A. Sharma, R. Kumar and R. K. Sharma, "Rearrangement of Recognized Strokes in Online Handwritten Gurmukhi Words Recognition," in *10th International Conference on Document Analysis and Recognition*, 2009.

[7] R. Rani, R. Dhir and G. s. Lehal, "Script Identification of Pre-Segmented Multi-Font Characters and Digits," in *Document Analysis and Recognition (ICDAR), 12th International Conference on*, 2013.

[8] R. Kumar and A. Singh, "Detection and segmentation of lines and words in Gurmukhi handwritten text," in *Advance Computing Conference (IACC), 2010 IEEE 2nd International*.

[9] M. K. Jindal, R. K. Sharma and G. S. Lehal, "Segmentation of Horizontally Overlapping lines in Printed Gurmukhi Script," in *International Conference on*

Advanced Computing and Communications, 2006.

[10] K. Verma and R. K. Sharma, "Comparison of HMM-and SVM-based stroke classifiers for Gurmukhi script," *Neural Computing and Applications*, Vols. 10.1007/s00521-016-2309-5, pp. 1-13, 2016.

[11] M. Vyas and K. Verma, "A platform independent methodology for on-line handwritten character segmentation," in *Advanced Communication Control and Computing Technologies (ICACCCT), International Conference on*, 2014.

[12] C. J. Mathew, R. C. Shinde and C. Patil, "Segmentation techniques for handwritten script recognition system," in *Circuit, Power and Computing Technologies (ICCPCT), International Conference on*, 2015.

[13] C. Djeddi, I. Siddiqi, L. Souici-Meslati and A. Ennaji, "Text-independent writer recognition using multi-script handwritten texts," *Pattern Recognition Letters*, vol. 34, pp. 1196-1202, 2013.

[14] P. Roy, A. K. Bhunia, A. Das, P. Dey and U. Pal, "HMM-based Indic handwritten word recognition using zone segmentation," *Pattern Recognition*, 2016.

[15] U. Pal and B. B. Chaudhuri, "Indian script character recognition: a survey," *pattern Recognition*, vol. 37, pp. 1887-1899, 2004.

[16] J. J. Brault and R. Plamondon, "Segmenting handwritten signatures at their perceptually important points," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, pp. 953-957, 1993.

[17] G. Kour and R. Saabne, "Real-time segmentation of on-line handwritten arabic script," in *Frontiers in Handwriting Recognition (ICFHR), 14th International Conference on*, 2014.

[18] {. D. Trier, A. K. Jain and T. Taxt, "Feature extraction methods for character recognition-a survey," *Pattern recognition*, vol. 29, pp. 641--662, 1996.

[19] N. VenkateswaraRao, A. Srikrishna, B. RaveendraBabu and G. R. M. Babu, "An efficient feature extraction and classification of handwritten digits using neural networks," *International Journal of Computer Science, Engineering and Applications*, vol. 1, p. 47, 2011.

[20] G. Louloudis, B. Gatos, I. Pratikakis and C. Halatsis, "Text line and word segmentation of handwritten documents," *Pattern Recognition,* vol. 42, pp. 3169--3183, 2009.

[21] G. Seni and E. Cohan, "External word segmentation of off-line handwritten text lines," *Pattern Recognition,* vol. 27, pp. 41--52, 1994.

[22] A. Bishnu and B. B. Chaudhuri, "Segmentation of Bangla handwritten text into characters by recursive contour following," in *Document Analysis and Recognition, 1999. ICDAR'99. Proceedings of the Fifth International Conference on*.

[23] R. Ghos and P. P. Roy, "Study of two zone-based features for online Bengali and Devanagari character recognition," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*.

[24] C. A. Mello, E. Roe and E. B. Lacerda, "Segmentation of Overlapping Cursive Handwritten Digits," in *Proceedings of the eighth ACM symposium on Document engineering*, 2008.

[25] R. Ghosh, D. Bhattacharyya and S. K. Bandyopadhyay, "Segmentation of Online Handwritten Word by Estimating the Busy Zone of the Image," in *The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp),* 2013.

[26] V. Lajish and S. K. Kopparapu, "Online handwritten devanagari stroke recognition using extended directional features," in *Signal Processing and Communication Systems (ICSPCS), 8th International Conference on*, 2014.

[27] D. V. Sharma and G. S. Singh, "An iterative algorithm for segmentation of isolated handwritten words in Gurmukhi script," in *Pattern Recognition, 2006. 18th International Conference on*.

[28] K. Verma and R. K. Sharma, "Performance Analysis of Zone Based Features for Online Handwritten Gurmukhi Script Recognition using Support Vector Machine," in *Progress in Systems Engineering*, 2015.

[29] M. Kumar, R. K. Sharma and M. K. Jindal, "Segmentation of lines and words in handwritten Gurmukhi script documents," in *Proceedings of the First International*

Conference on Intelligent Interactive Technologies and Multimedia, 2010.

[30] M. Younes and Y. Abdelllah, "Segmentation of Arabic handwritten text to lines," *Procedia Computer Science*, vol. 73, pp. 115--121, 2015.

[31] M. K. Jindal, R. K. Sharma and G. S. Lehal, "Segmentation of touching characters in upper zone in printed Gurmukhi script," in *Proceedings of the 2nd Bangalore Annual Compute Conference*, 2009.

[32] W. Li and N. S., "Text recognition for information retrieval in images of printed circuit boards," in *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*.

[33] G. Pagare and K. Verma, "Associative Memory Model for Distorted On-Line Devanagari Character Recognition," in *2015 Fifth International Conference on Advances in Computing and Communications (ICACC)*.

List of Publication

Rajendra Verma, Karun Verma, "Segmentation of Online Handwritten Gurmukhi Script", Journal of Neural Computing and Applications, Springer Verlag (2016), Communicated.

Video Link

https://www.youtube.com/watch?v=maZFr3AEGkA

Plagiarism Certificate

final

ORIGINALITY REPORT

8%	6%	3%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.thapar.edu:8080 Internet Source	2%
2	lrd.yahooapis.com Internet Source	2%
3	gdeepak.com Internet Source	1%
4	www.gdeepak.com Internet Source	1%
5	Bhattacharya, Nilanjana, and Umapada Pal. "Stroke Segmentation and Recognition from Bangla Online Handwritten Text", 2012 International Conference on Frontiers in Handwriting Recognition, 2012. Publication	<1%