A Fuzzy Approach to Hand written Devnagari Script up to five Characters Recognition

Pankaj Sharma
Research Scholar
Department of CSE
Bhagwant University
Ajmer, Rajasthan, India

S. K. Gupta, PhD
Department. of CSE
Bundelkhand Institute of Engineering & Technology
Jhansi (U. P.), India

ABSTRACT

This paper elaborates Hindi curve character recognition based on fuzzy logic. Characters are classified by if then rules based on Fuzzy logic into seven classes according to vertical bar. The experiment is performed for 10 different datasets of handwritten three-character, four-character and five-character words from 100 peoples. This group of people has different aged members. The experiment was successfully performed and the average recognition rate is found to be 94.8%.

General Terms

Fuzzy logics, Digital Image Processing, Algorithm, Pattern Recognition, Security et. al.

Keywords

Devnagari characters, Segmentation, classification, Feature Extraction,

1. INTRODUCTION

A lot of activity is taking place today on the development of OCRs for the scripts of Indian languages [1- 6]. Electronic representation and access of documents and other material in Indian languages have been lagging behind for many years. The recent wide adoption of the computer and the channels provided by Information Technology in India has resulted in a big spurt in the use of electronic representation, storage, and access of information in Indian languages. The demand for content in the local languages is ballooning as Information Technology reaches beyond the English-speaking sections of the populace. This trend is likely to continue in a populous country like India. A similar trend is expected in other developing nations also. An Optical Character Reader is a critical tool in creating electronic content in Indian languages as a rich tradition of literature and a vibrant publishing industry have existed in all of them for a long time. Indian languages and the scripts used to write them pose great challenges to an OCR designer.

In this paper, we will present a fuzzy rule based classification and recognition system for handwritten curve Hindi characters. Hindi is one of the official languages of India. It is world's third most commonly used language after Chinese and English. Brahmi is the mother script for most of the Indian languages. These scripts are used for two major linguistic communities: in the north, Indo-European languages, and in the south, Dravidian languages [7].

Devnagari contains 11 vowels and 33 consonants. Vowels can be written as independent characters, or by using a various diacritical marks which are written below, above, after or before the consonant they belong to. The vowels written in this manner are called as *modifiers* and the characters formed in this manner are represented as *conjuncts*. Sometimes two or more consonants take new shapes as they combined together and form *compound characters*. Such type of characters, modifiers and compound characters are present in Devnagari as well as in other scripts. Hindi as well as Sanskrit, Nepali and Marathi, follows the Devnagari script [8]. A sample data set of Devnagari character is given in Figure 1.

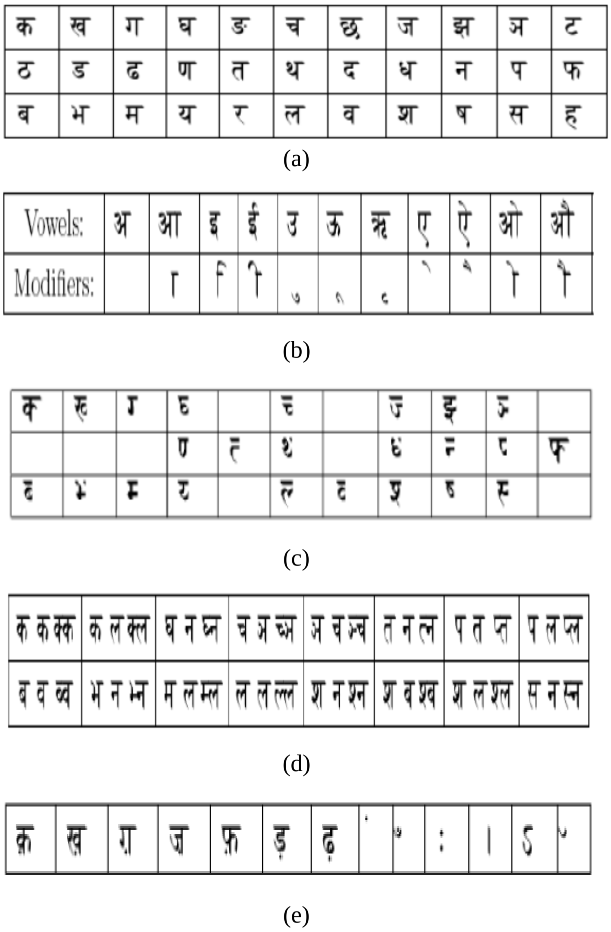
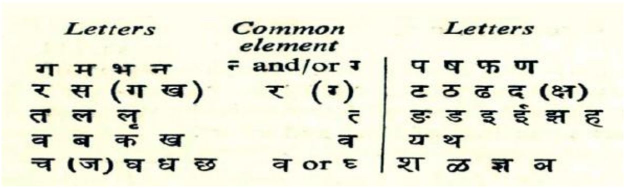

(a)
Vowels:
Modifiers:
(b)
(c)
(d)
(e)

Figure 1: (a) Basic Constants, (b) Swars (vowels) & corresponding matras (modifiers) and (c) Half constants with vertical bar (d) some combinations of half constants and constants (e) Set of Special symbols

Each of the characters has a horizontal line at its upper part, which is called as *Shirorekha* or headline. Such characteristic is not available in any English character, so this characteristic can be used to distinguish Devnagari from English. In continuous handwriting, in the direction from left to right, one character's shirorekha joins with the shirorekha of the next or previous character if both the characters belong to the same word. Multiple characters and modified shapes in a word look as a single connected component that is joined through the common shirorekha. Devnagari also has consonants, vowels, vowel modifiers compound characters, and numerals. Moreover, many characters have similar shapes. All these variations make Hand written Hindi Character Recognition a challenging problem [9].

2. ANATOMY OF DEVNAGARI CHARACTERS

The first graphical classification of Devanagari script was done by S. V. Bhagwat [10]. He grouped letters on the basis of graphical similarities as shown in figure 2.



Letters	*Common element*	*Letters*
ग म भ न	न and/or ग	प ष फ ण
र स (ग ख)	र (ग)	ट ठ ढ द (क्ष)
त ल लृ	त	ङ ड इ ई झ ह
व ब क ख	व	य थ
च (ज) घ ध छ	व or ६	श ळ ज्ञ ञ

Figure 2: Bhagwat's grouping of letters on the basis of graphical similarities [9]

He has also defined guidelines for the letters and terminology for some of the graphical elements present in the letters which are shown in figure 3.

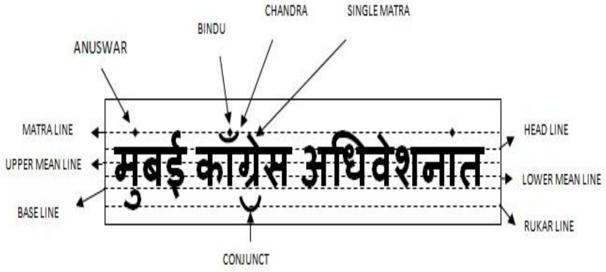
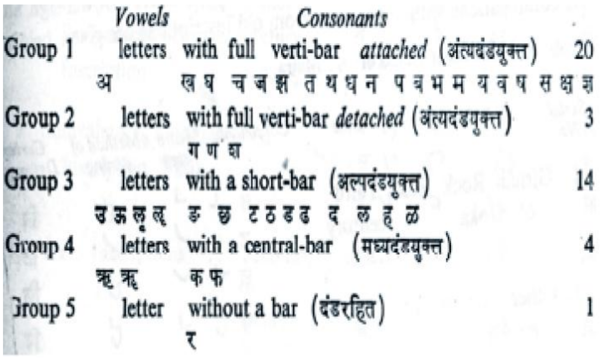


Figure 3: Bhagwat's guidelines [9]

The top most line is the *Matra* Line that denotes the top of the upper *matras*. The *Matra* Line is followed by the Head Line. The top of the *Shiro Rekha* is called as Head Line which is followed by the Upper Mean Line and the Lower Mean Line. Upper Mean Line indicates the point where the actual letter starts. Lower Mean Line indicates the point where the characteristic feature of the letter comes to an end. Lower Mean Line is followed by the Base Line. Base Line denotes the end of the character and the point where the lower *matra* starts. The lowermost line is the *Rukar* Line. Bapurao Naik also attempted the graphical grouping of letters. Naik organized letters graphically in five groups on the basis of the position of the vertical bar. Naik's grouping of letters is shown in figure 4.



	Vowels	*Consonants*	
Group 1	letters	with full verti-bar *attached* (अंत्यदंडयुक्त)	20
	अ	ख घ च ज झ त थ ध न प ब भ म य व ष स क्ष ज्ञ	
Group 2	letters	with full verti-bar *detached* (अंत्यदंडयुक्त)	3
		ग ण श	
Group 3	letters	with a short-bar (अल्पदंडयुक्त)	14
	उ ऊ लृ लॄ	ङ छ ट ठ ड ढ द ल ह ळ	
Group 4	letters	with a central-bar (मध्यदंडयुक्त)	4
	ॠ ऋ	क फ	
Group 5	letter	without a bar (दंडरहित)	1
		र	

Figure 4: Naik's grouping of letters on the basis of the position of the vertical bar or the kana [9]

3. PROPOSED SYSTEM

The proposed system executes the following six stages:

3.1 Preprocessing

During this stage, following operations are performed on the gathered data to make it appropriate for further processing—

3.1.1 Scanning

Handwritten word data samples, collected from various resources, are scanned to transform data into a gray scale image.Text embedded on a paper are usually scanned optically and saved in the form of pixels in a file. These pixels may have one of the two values: 0 (OFF) or 1 (ON) for binary images, 0– 255 gray levels for gray scale images, and 3 channels of 0–255 colour values for colour images.

3.1.2 Thresholding

A colour or grayscale image is transformed to a binary image.

3.1.3 Noise reduction

The noise, generated by the writing instrument or by the optical scanner, may create bumps, disconnected line segments, gaps in lines, and filled loops etc. The distortion as dilation, erosion, rounding of corners, local variations are required to eliminate before the character recognition [11- 12].

3.1.4 Skew Detection and Correction

Handwritten document may be skewed originally or skewness may arise in the process of document scanning. This accidental effect should be removed because it highly reduces the subsequent processes accuracy, such as classification and segmentation. Skewed lines are aligned using skew angle and making suitable correction in the raw image [13- 14].

3.1.5 Size Normalization

To convert all characters in to same data size, each segmented character is truncated to be stored in suitable matrix of size 32x32 or 64x64 [15].

3.1.6 Thinning

To identify relevant features the boundary detection of image is done [16].

3.2 Segmentation

Histogram equalization is applied on Thinned word image to break it into individual characters. The steps of Histogram equalization are mentioned below-

(i) To detect the boundaries of theword a horizontal histogram is taken.

(ii) To get the region of each character vertical histogram is taken.

(iii) In some cases it is possible that number of regions exceeds the number of characters inthe word. It may be the case when a character presented in word while its vertical bar is saparated from the character. Here, the region of the vertical bar with the highest peak value is selected as a part of the character to its left.

This process is used to segment an image of a sentence into sub images of individual symbols by splitting words and lines [17- 18]. By removing Shirorekha, words can be segmented to individual character for recognition (see figure 5 (c)). Various vowel modifiers can be splitted to extract structural feature [19- 20].

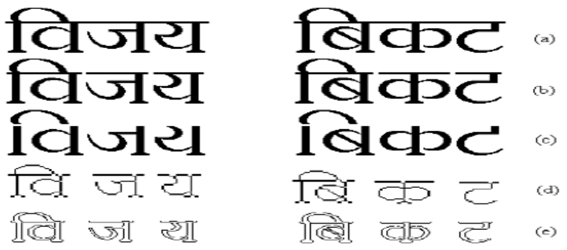


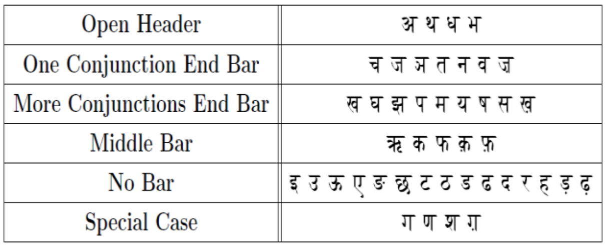
Figure 5: Preprocessed Images (a) Original, (b) segmented (c) Shirorekha removed (d) Thinned (e) image edging

3.3 Classification

Each character of Hindi language is constituted by three major components: header line or SHIROREKHA, vertical bar, and curves. In the method we are proposing next, vertical bar component is used to classify characters.

Classification of Characters on the basis of vertical bar

A vertical bar either appears at the right end (End Bar) or in the middle (Middle Bar) of a Hindi character not at the left end of a Hindi character. On the basis of the presence and position of vertical bar and the conjunction, all the core Hindi characters can be categorized into the following six groups:



Open Header	अ थ ध भ
One Conjunction End Bar	च ज ञ त न व ज़
More Conjunctions End Bar	ख घ झ प म य ष स ख़
Middle Bar	ऋ क फ क़ फ़
No Bar	इ उ ऊ ए ङ छ ट ठ ड ढ द र ह ड़ ढ़
Special Case	ग ण श ग़

Figure 6: Classification of Hindi characters on basis of vertical bar

The four characters ग ण श ग़ in the Special Case class have an end bar, but if we remove the header line and compute the vertical projection, each of these characters will segment into two parts. A point with 3 or more pixels in its neighbourhood is called a junction point. Method to extract the above features is elaborated in VERTICALBAR_INFO and JUNCTIONPOINT_COUNT algorithm. A movable 3X3 mask (Fig.7) is applied on the image, which covers 8-neighborhood of the pixel P0.

P8	**P1**	**P2**
P7	**P0**	**P3**
P6	**P5**	**P4**

Figure 7: Mask 3X3

TABLE 1 shows the features various features which are the basis to form different classes of characters. A character can classify into one class only.

Table 1: Features used for classification

Feature	**Symbol**	**Values**
Presence of Vertical Bar	VB	P (Present) NP (Not Present)
Position of Vertical bar	POS	M (Middle) RE (Right End)
Length of vertical bar	LEN	S (20%-30% of Character width W) L (70%- 80% of character width W)
Connectedness of vertical bar to character	CON	C (Connected) NC (Not Connected)
Number of junction points	JP	1,2,3,4 or 5

Following are the notations used in these algorithms:
Pc -- current pixel
CL -- current location
COUNT_P -- counter variable to count the number of pixels. Initial value is set to 0.
COUNT_JP -- counter variable to count the number of junction points. Initial value is set to 0.
ROW -- current row number
COL -- current column number

Algorithm VERTICALBAR_INFO

To determine the information about the vertical bar do the following:

1. Starting from the last column of the first row i.e. ROW==0 & COL==8, convolute the mask on the binary image of character and check:
(i) Call pixel as P0 if it is a foreground pixel.
IF number of neighboring pixels of P0 _ 3 and one pixel is P5 then do the following --
(a) Pc = P0.
(b) Set N = COL.
(c) COUNT_P = COUNT_P+1
(ii) ELSE COL = COL - 1 and repeat step (i) till COL >= 4
2. Identify the presence of vertical bar as:
IF COUNT_P ==1
THEN VB → P
ELSE VB → NP.

```
3. Identify the position of vertical bar as:
IF N >=8
THEN POV → RE
ELSE POV → M
4. To identify the length of vertical bar and its connectedness to character:
(i) IF POV==M
THEN do till P5 is encountered
(a) Set P5=P0
(b) COUNT_P = COUNT_P + 1
(ii) IF COUNT_P >3
THEN LEN → L
ELSE LEN → S
(iii) IF POV ==RE
THEN Set CP=P0 and do till P5 is encountered
IF P6 OR P7 OR P8 exists
THEN CON → C
ELSE CON → NC
```

Algorithm NUM_JUNCTIONPOINTS

To determine the number of junction points do the following:

```
1. Starting from ROW = 0 and COL = 0, moving the mask on the image from left to right.
2. Find the first foreground pixel and call it P0
   IF number of neighbouring pixels of P0 >= 3
         THEN COUNT_JP = COUNT_JP + 1
ELSE P0=P3
3. Repeat step 2 for each pixel in the image.
4. Set JP=COUNT_JP
```

Following fuzzy rules are formed using these two algorithms to classify the characters into one of the seven classes.

(i) IF VB == NP THEN character belongs to class A (उ ए र)

(ii) IF VB == P AND POV == M AND LEN == L THEN character belongs to class B (क फ ऋ)

(iii) IF VB == P AND POV == M AND LEN == S AND JP < 2 THEN character belongs to class C(ङ ट ड)

(iv) IF VB == P AND POV == M AND LEN == S AND JP >= 2 THEN character belongs to class D (इ छ ठ ढ द ह)

(v) IF VB == P AND POV== RE AND CON == NC THEN character belongs to class E (ग ण श)

(vi) IF VB == P AND POV == RE AND CON == C AND JP <4 THEN character belongs to class F(च ज ञ त थ न प य ल व ष)

(vii) IF VB == P AND POV == RE AND CON == C AND JP >= 4 THEN character belongs to class G(अ ख घ झ ध ब भ म स)

3.4 Feature Extraction

Steps for extracting features are given in following algorithm-

Algorithm FEATURE_REC

Following algorithm explains the steps for extracting features-
Algorithm FEATURE_REC

```
1. Apply the following method to remove header line-
(i) Convolute the 3X3 mask on the normalized image and scan the first row from right to left.
(ii) Find the first foreground pixel and call it P0.
IF P7 is a foreground pixel OR P0 is an end point OR P0 is a disconnected component
SET P0 = 0
ELSE convolute mask in the left direction.
```

Image is scanned from right to left To avoid the deletion of character pixels such as अ थ ध भ श , image is scanned from right to left because these characters, except भ , can be written in two styles— (a) header line covers the whole character and, (b) when header line covers only half or a portion of the character. In the first case, this step may delete the pixels, those are common to character and header line, as in characters mentioned above as well as characters such as उ र स and may produce some disconnected components with small number of pixels.

```
2. To delete disconnected components follow the steps written below-
(i) Set Row = 1 and Scan the image from left to right.
(ii) Assign the name P0 to the first found foreground pixel found.
(iii) IF P3 ==1
IF any pixel in 8 neighbourhood of P3 does not exists
THEN SET P0=0 AND P3=0
ELSE IF P5==1
IF any pixel in 8 neighbourhood of P5 does not exists
THEN SET P0=0 AND P5=0
```

On applying the 3X3 movable mask on the normalized image of classified character and scan the image from top to bottom row wise, we can have following information for junction points and end points--

(i) N1: total number of junction points
(ii) N2: total number of end points
(iii) JPi: ith junction point, where i=1 to N1
(iv) EPi: ith end point where i=1 to N2
(v) Curve (JPi): curve on ith junction point (Table 6.2)
(vi) Curve (EPi): curve on ith end point Line(JPi) : line on ith junction point (Table 6.2)
(viii) Line(EPi): line on ith end point
(ix) Loop(JPi): loop on ith junction point
(x) D1(i): direction of next endpoint from ith end point
(xi) D2(i): direction of next junction point from ith junction point

Values and symbols of different types of curves, lines & loops are given in the Table 2.

Table 2: Values and symbols for curves, lines and loop

Features	Values	Symbol
Curve	Left Curve	LC
	Lower Left Curve	LLC
	Right Curve	RC
	Upper Right Curve	URC
	Lower right Curve	LRC
	U Curve	U
Line	Vertical Line	VL

	Horizontal Line	HL
	Back slash	BS
Loop	Present	P
	Not Present	NP

Algorithm CURVE_LINE_LOOP_INFO

To find out the nature of the curve:

Convolute the mask on the binary image of classified character from bottom to top row wise.

Call the first foreground pixel as current pixel (Pc).

```
1. If Pc is a junction point or end point, then check the 8-neighbourhood of Pc.
(a) IF P1 is true THEN
(i) Repeat till P1 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(b) IF P3 is true THEN
(i) Repeat till P3 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(c) IF P8 is true THEN
(i) Repeat till P8 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(d) IF P1 OR P2 is true THEN
(i) Repeat till P1 OR P2 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(e) IF P1 OR P8 is true THEN
(i) Repeat till P1 OR P8 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(f) IF P2 OR P3 OR P4 is true THEN
(i) Repeat till P2 OR P3 OR P4 traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(g) IF P4 OR P5 is true THEN
(i) Repeat till P4 OR P5 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
(h) IF P6 OR P7 OR P8 is true THEN
(i) Repeat till P6 OR P7 OR P8 is traversed
(ii) COUNT = COUNT + 1.
ELSE stop.
2. Check the following to know the type of curve and line:
(i) IF step 1(h) is true
IF step 1(a) is true
IF step 1(f) is true
IF COUNT >=3
THEN Curve → LC

(ii) ELSEIF step 1(e) is true
IF step 1(f) is true
IF COUNT >=2
THEN Curve → ULC
(iii) ELSEIF step1(h) is true
IF step 1(e) is true
IF COUNT >=2
THEN Curve → LLC
(iv) ELSE IF step 1(f) is true
IF step 1(a) is true
IF step 1 (h) is true
IF COUNT >=3
THEN Curve → RC.
(v) ELSE IF step 1(d) is true
IF step 1(h) is true
IF COUNT >=2
THEN Curve →URC.
(vi) ELSE IF step 1 (f) is true
IF step 1(e) is true
IF COUNT >= 2
THEN Curve →LRC.
(vii) ELSEIF step 1(g) is true
IF step 1(h) OR step1 (f) is true
IF step 1(d) is true
IF COUNT >=3
THEN Curve →U
(viii) IF step 1(a) is true
IF COUNT _>=2
THEN Line →VL
(ix) IF step 1(b) is true
IF COUNT >=2
THEN Line →HL
(x) IF step 1(c) is true
IF COUNT >= 2
THEN Line →BS
THEN Loop →P.
```

3.5 Recognition

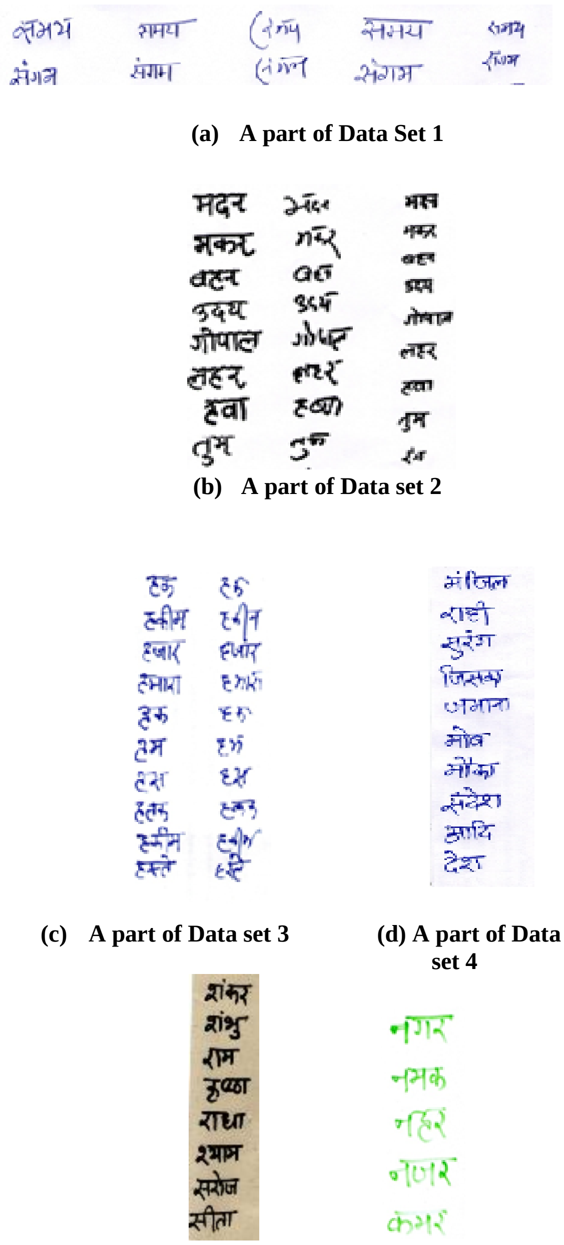
Fuzzy rules are applied for the purpose of recognition. Following Rules based on classification applied for characters:

```
1. IF Class is A
IF Curve (EP1) == RC
THEN character is उ
ELSE IF Curve (JP1) ==LRC
IF N2==4 OR D1 (3) == P3
THEN character is ए
ELSE character is र
2. IF Class is B
IF Curve (EP2) == LC
THEN character is ऋ
ELSE IF Curve (EP2) == URC
IF Curve (JP1) == LC OR Loop(JP1) ==P
THEN character is क
ELSE character is फ
3. IF Class is C
IF Curve (EP1) == LC
THEN character is ट
ELSE IF Curve (EP1) == RC
IF N2==3
THEN character is ङ
ELSE character is ड
4. IF Class is D
IF Curve (EP1) == LC
THEN character is ह
ELSE IF Curve (JP1) == LC
IF N2 < 2
THEN character is ढ
ELSE IF N2==2
THEN character is द
ELSE character is छ
ELSE IF Loop (JP1) ==P
IF Curve (JP1) == RC OR URC
THEN character is इ
ELSE character is ठ
5. IF Class is E
IF Loop (JP1) ==P
IF N1==2
THEN character is श
ELSE character is ग
ELSE IF Curve (EP1) == U
THEN character is ण

6. IF Class is F
IF N2 > 3
IF Curve (EP1) == ULC
THEN character is ल
ELSE IF Curve (EP1) == RC OR Curve (EP2) == RC
THEN character is ञ
ELSE character is च
ELSE IF N2==3
IF Curve (JP1) == LLC
THEN character is प
ELSE IF Curve (JP1) ==U
THEN character is य
ELSE IF Curve (EP1) == ULC
THEN character is त
ELSE character is ज
ELSE
IF Curve (JP1) ==U
THEN character is न
ELSE IF Curve (JP1) ==LLC
THEN character is ष
ELSE IF Curve (JP1) ==LC OR Loop (JP1) ==P
THEN character is व
ELSE character is थ
7. IF Class is G
IF N2>4
IF Curve (EP1) == RC
THEN character is अ
ELSE IF Line (EP1) == BS
IF D2 (1) ==P3 OR D2(2)==P3
THEN character is स
ELSE character is ख
ELSE IF N2 ==4
IF Loop on JP1 ==P
THEN character is झ
ELSE character is घ
ELSE
IF Curve (JP1) ==LLC OR U
THEN character is ध
ELSE IF Curve (JP1) == LC
THEN character is ब
ELSE IF Loop on JP1 ==P
IF Loop on JP3 ==P OR LINE (EP2) == HL
THEN character is भ
ELSE character is म
```

4. RESULT AND DISCUSSION

10 Datasets have been created for each type of word string: three, four and five characters by collecting handwritten Hindi words samples from 100 people of different age groups on A4 data sheets using different colored pens and styles. The persons were requested to write asked string of characters as words. A part of datasets for three characters word is shown in the figure 8.


समय
संगम
मदर
गोपाल
लहर
हवा
तुम
शंकर
शंभु
राम
कृष्णा
राधा
श्याम
सरोज
सीता
नगर
नमक
नहर
कमर

(a) A part of Data Set 1

(b) A part of Data set 2

(c) A part of Data set 3

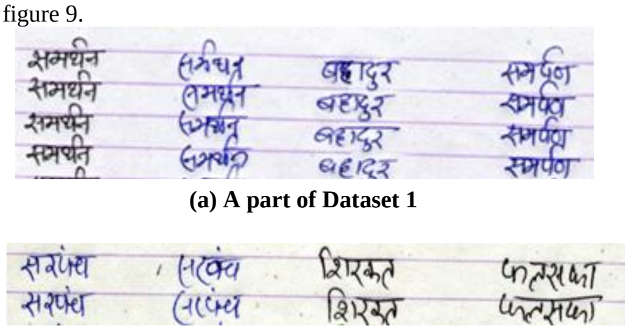
(d) A part of Data set 4

(e) A part of Data set 5

(f) A part of Data set 6

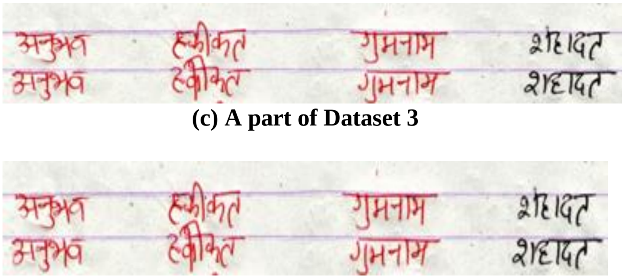
Figure 8 (a) to 8 (f): Sample Datasets for three characters word

A part of datasets for four characters word is shown in the figure 9.


समर्थन
बहादुर
समर्पण
शिरकत

(a) A part of Dataset 1

(b) A part of Dataset 2

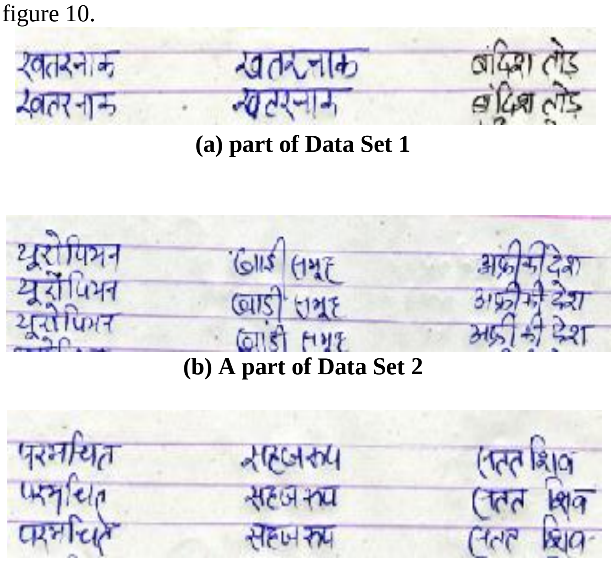

अनुभव
हकीकत
गुमनाम
शहादत

(c) A part of Dataset 3

(d) A part of Dataset 4

Figure 9 (a) to 9 (d): Sample Datasets for four characters word

A part of datasets for five characters word is shown in the figure 10.


खतरनाक
यूरोपियन
अफ्रीकी देश
सहजरूप

(a) part of Data Set 1

(b) A part of Data Set 2

(c) A part of Data Set 2

Figure 10 (a) to 10 (c): Sample Datasets for five characters word

Comparative results of the three types of datasets on the basis of recognition rate are shown in table 3.

Table 3: Recognition rates for various characters on the basis of size

Word Size (in characters)	Average Recognition Rate (in %)
3-character word	95.186
4-character word	89.65
5-character word	90.03

The graphical representation of comparative study of 3-character word with 4-character and 5-character word is shown in figure 11.

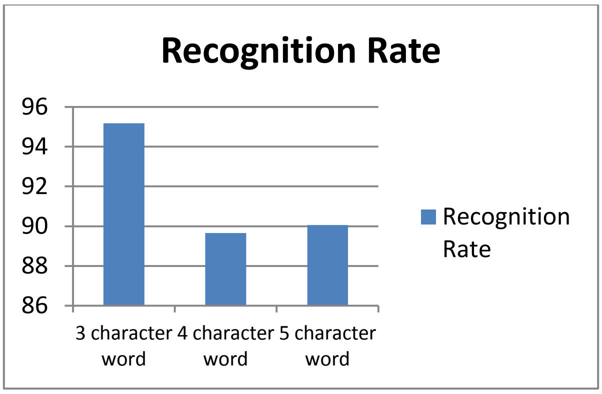
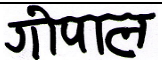
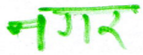
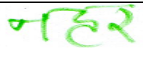
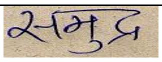
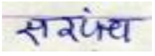
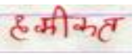
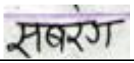
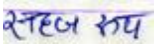


Figure 11: Recognition rate for words having 3-character, 4-character, and 5-character size

The average recognition rate do not depend on the character set while it is definitely depends on the writing style of the character. When a character repeats in a word, the recognition rate also increases. As we can see from the results, the recognition rate for five character set is quiet competitive and satisfactory as the system was observing the similar characters in a word also which increased the recognition rate. While the recognition rate for four characters is observed not so good, it may be because of the writing style of the data set available.

The difficulties we observed when a similar character is represented to the network with a little with difference, means when a confusory character is represented to the network, the network absolutely perturbed and rate of recognition falls a bit. Sometimes machine could not understand the handwriting, style or font and the networks stuck in falls minima, results system failed absolutely to recognize the word. Such words from the datasets are listed below:

Table 4: Unrecognized or partial recognized or difficulty in recognition in words

Sr. No.	Words not recognized completely
1	
2	
3	
4	
5	
6	
7	
8	

5. CONCLUSION

The network perform faster and steady for the characters represented in repeated format i.e. when the same character is repeated again and again to the network within different words. Overall the experiment performed here shown very good result over the fuzzy classification rule sets.

The limitations of the experiment were its treatment with only for three types of character strings from hand written Hindi script. It can further be improved by enhancing fuzzy if-then rules for classification of more than five characters word and further sentences to scripts.

6. REFERENCES

[1] Jesse Hansen, "A Matlab Project in Optical Character Recognition (OCR)", http://www.ele.uri.edu/~hansenj/projects/ele585/OCR/OCR.pdf

[2] J. Mantas, "*An Overview of Character Recognition Methodologies",* Pattern Recognition, Vol. 19, No 6, p. 425-430, 1986.

[3] Eikvil Line, "OCR-Optical character recognition system", Report No. 876, *Document Image Analysis Publications*, Norwegian computing center, 1993, http://www.nr.no/documents/samba/research_areas/BAMG/Publications/OCR.ps.gz.

[4] U. Pal, B.B. Chaudhuri "Indian script character recognition: a survey" *Pattern Recognition* 37, pp. 1887 – 1899, 2004.

[5] George Nagy, Thomas A. Nartker, Stephen V. Rice, "Optical Character Recognition: An illustrated guide to the frontier", *Proceedings of Document Recognition and Retrieval VII*, SPIE Vol. 3967, Sanjose, pp. 58-69, January 2000.

[6] Abdel Belaid, "OCR Print - An overview" chapter 2 in "Survey of the state of the art in human language technology", pp: 71 – 74, 1997.

[7] Richard Ishida, "An Introduction to Indic Scripts", http://www.w3.org/2002/Talks/09-ri-*indic/indic-paper.pdf.*

[8] U. Pal, B. B. Chaudhuri, "Indian Script Character recognition: A survey", *Pattern Recognition,* vol. 37, pp. 1887-1899, 2004.

[9] Ram Sarkar *et al*, "A Script Independent Technique for Extraction of Characters from Handwritten Word Images", *International Journal of Computer Applications* Vol. 1, No. 23, 2010.

[10] Bapurao S. Naik — Typography of Devanagari

[11] J. Serra, "Morphological Filtering: An Overview", *Signal Processing*, vol. 38, no.1, pp.3-11, 1994.

[12] Nafiz Arica, Fatos T. Yarman-Vural, "An Overview of Character Recognition Focused On Off-line Handwriting", *C99-06-C-203, IEEE,* 2000.

[13] Mohamed Cheriet, Nawwaf Kharma, Cheng-Lin Liu, Ching Y. Suen, "Character Recognition Systems: A Guide for students and Practioners", John Wiley & Sons, Inc., Hoboken, New Jersey, 2007.

[14] Rajiv Kapoor, Deepak Bagai, T. S. Kamal, "Skew angle detection of a cursive handwritten Devnagari script character image", *Journal of Indian Inst. Science*, pp. 161–175, May-Aug. 2002

[15] U. Pal, M. Mitra and B. B. Chaudhuri, "Multi-Skew Detection of Indian Script Documents*", CVPRU IEEE*, pp 292-296 , 2001.

[16] V. H. Mankar *et al*. "Contour Detection and Recovery through Bio-Medical Watermarking for Telediagnosis," *International Journal of Tomography & Statistics*, Vol. 14 (Special Volume), Number S10, summer 2010.

[17] Richard G. Casey and Eric Lecoline, "A survey of methods and strategies in Character segmentation".

[18] Naresh Kumar Garg, Lakhwinder Kaur, M. K. Jindal, "A New Method for Line Segmentation of Handwritten Hindi Text", *7th International Conference on Information Technology*, 2010.

[19] C. V. Jawahar, M. N. S. S. K. Pavan Kumar, S. S. Ravi Kiran, "A Bilingual OCR for Hindi-Telugu Documents and its Applications", *Seventh Intl. Conf. on Document Analysis and Recognition (ICDAR 2003*) , pp 1-7, 2003.

[20] M. Hanmandlu and Pooja Agrawal, "A Structural Approach for Segmentation of Handwritten Hindi Text", *International Conference on Cognition and Recognition.*pp 589-597.