

Comparative Analysis Zoning, Distance Profile Feature And K-Nearest Neighbour For Handwritten Devanagari Script Recognition System

Dr. Rohit Sachdeva

Assistant Professor Department of Computer Science Multani Mal Modi College, Patiala
(Punjab) -147001

ABSTRACT

All over the world, OCR of machine-printed as well as handwritten text has become a very prominent area of research. OCR packages are available in the market with a very high degree of accuracy (99 present) in recognizing printed text in European, Kanji, and other scripts, but for Indian scripts, there is hardly any commercial application available. For other scripts, some commercial packages are available with a fairly high degree of accuracy for recognizing handwritten documents as well. Very little work has been done in handwritten Devanagari text when we compare it with printed Devanagari text. This paper presents an overview of the OCR system for Devanagari script which recognizes the handwritten Devanagari text.

INTRODUCTION

In last 6 decades a lot of work has been done on OCR by many authors. They have researched and discussed optical recognition of machine printed and hand written text. For Indic scripts like Bangali, Devanagari, Gujrati, Gurmukhi etc. very few work has been done as compare to Roman script in the field of printed text segmentation. Few papers on segmentation of machine printed Bangla[1], Devanagari [1,2], Gurmukhi[3-6] are available. In last decade segmentation of handwritten text in Indic script is one of the prominent areas of research. Few papers are published on segmentation of handwritten text in Bangla script [7, 8], Devanagari script [9, 10,15,16], Gurmukhi [11,12]. Segmentation of handwritten text in Indic script is challenging job due to variation in handwriting, touching characters, merged characters, overlapped characters etc. By using OCR one can convert handwritten text images or scanned machine printed images into editable form. The development of optical character recognition has following steps[17]:

- a) Image scanning and Digitization
- b) Pre-processing
- c) Segmentation
- d) Feature extraction
- e) Classification

f) Post-processing

a) Image scanning and Digitization: The image should be in the paper based form, first step is scanned the image and then stored in some image file in the form of bitmaps. This is known as digitization of paper form. Coloured form as well as monochrome form processing is supported by the system. While scanning the form, brightness, contrast and Scanning resolution are the key factors. Forms have been scanned by setting resolution at least value at 300 dpi, threshold value minimum at 100 and contrast value should be at 100[18].

b) Pre-processing: In the Pre-processing phase, on the scanned image a set of operations is performed, which is part of pre-processing. Operation which performed are noise removal, skew detection and correction, Normalization etc.

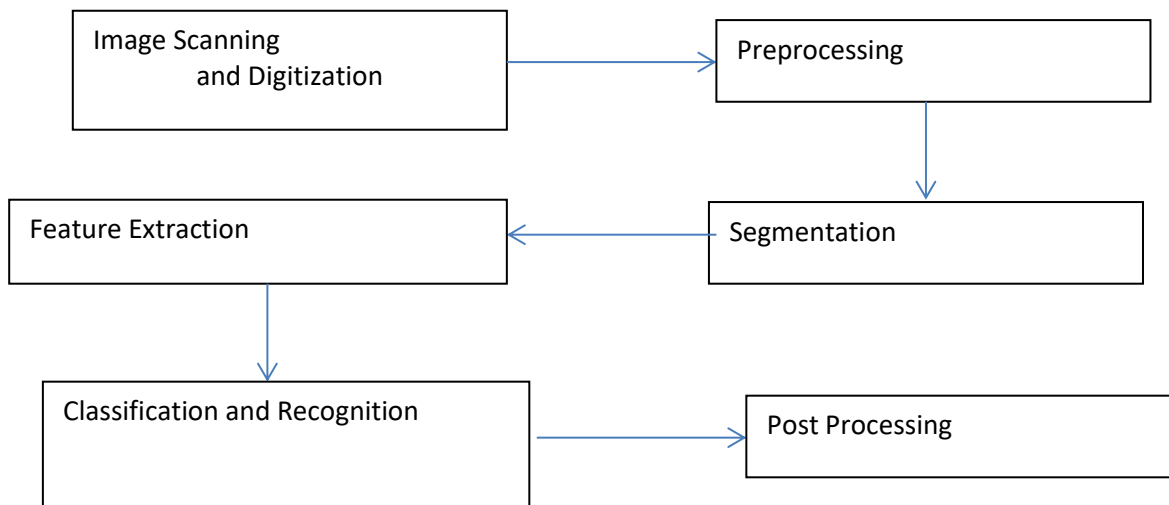


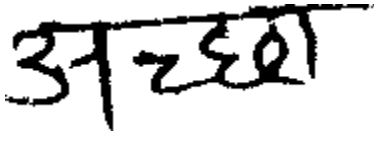

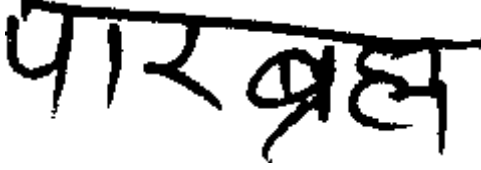


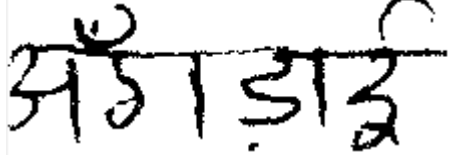


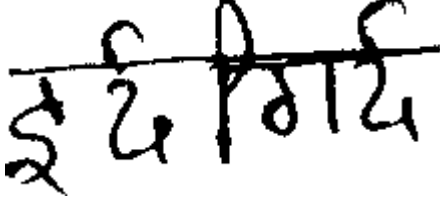
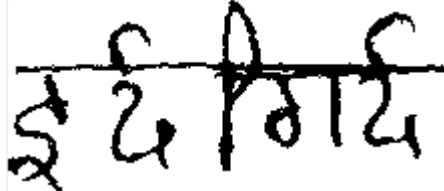
Figure 1: Block Figure of OCR System

(i) **Noise Removal:** Noise may be introduced during the scanning of the image. Noise is an arbitrary deviation of image intensity and visible as small particles in the image. It may be produced at the time of scanning or image transmission. In the recognition system, noise must be cleared because it may mislead. To clear the noise various methods are used. Morphological operations are used to smooth pixel boundary, to connect or complete the unconnected pixels, to eliminate or clear isolated pixels.

(ii) **Skew Detection and Correction:** Skewed image segmentation may give wrong results. So, for proper segmentation, image should not be skewed, it must be straight. As a result before segmentation, detection of skew in image and its correction is essential which straight the image and then it is used for proper segmentation and feature extractions. Skew can be uniform or non-uniform. For the machine printed Gurmukhi document Sharma and Lehal [13] proposed a sturdy method which is for isolated words to detect and correct the skew. Skewness of words is checked

and corrected by using method [13] and the comparison of data before and after skew correction of words are given in Table-I.

Table I: Comparison of word before and after skew correction

Skewed Word	After Skew Correction
	
	
	
	
	

The following table shows that in skewed image headline is not detected properly and after skew correction clear headline is detected which will help in proper segmentation.

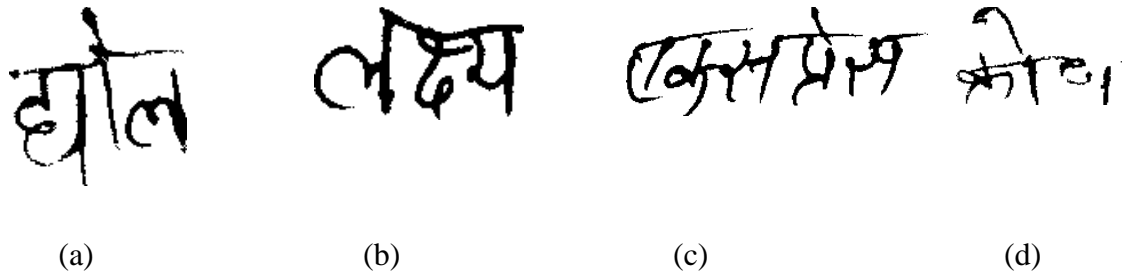


Table-I-A: Comparison of skew word with headline and deskew word with headline











Skewed image with headline detection	After skew correction headline detection
अच्छा 	अच्छा 
पारब्रह्म 	पारब्रह्म 
अँगाड़ाई 	अँगाड़ाई 
प्रेरणाशक्ति 	प्रेरणाशक्ति 
इंद्रगिर्दि 	इंद्रगिर्दि 

Figure 2: Failure cases of word level skew detection

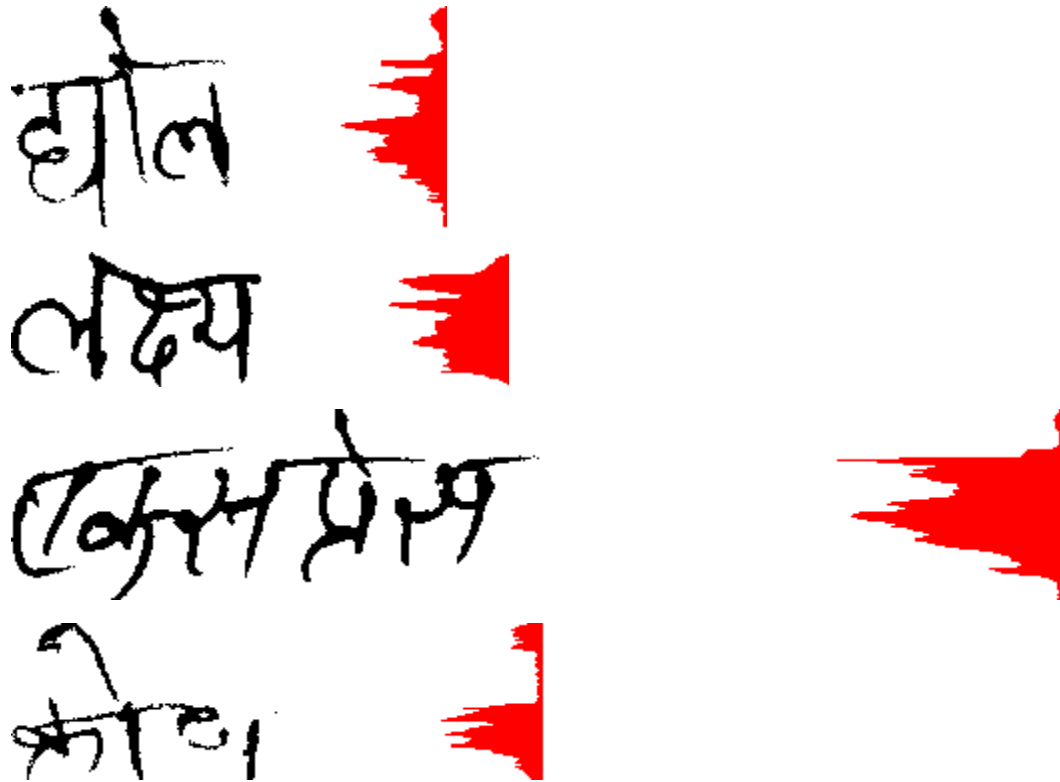


Figure 2a: Failure cases of skew algorithm with headline

As shown in figure 2a if the words are not deskewed properly, then the headline cannot be detected and it becomes challenging task to segment these words.

(iii) Normalization: In handwritten character recognition, shape normalization plays an important role. It is a method which changes different image factors such as shapes, size, rotation and translation to obtain more suitable values. In the present work, characters are normalized at size of 48 x 48. For scaling the characters, characters are segmented and then normalized to particular size such as 48*48 by scaling up or scaling down. These normalized character images form the input of feature extraction methods. Some results of normalization of segmented characters are given in Table II.

Table II: Some results of normalization of segmented characters' images

Original Image with Dimension	Scaled Image with dimension 48*48	Original Image with Dimension	Scaled Image with dimension 48*48
 59*50		 47*64	
 36*69		 47*41	
 36*47		 43*41	
 35*39		 40*46	
 31*52		 38*62	
 55*52		 40*56	
 38*41		 31*61	
 46*43		 33*57	
 65*34		 39*41	

c) **Segmentation:** Segmentation of handwritten text is relatively more challenging than the machine printed text, as hand written text is not properly structured, therefore detection of gaps between the lines and headline is not easy task. Segmentation process performed in three parts. In the first part, the word is segmented into three zones. In the second part words containing overlapped characters are further segmented to the maximum possible level and in the third part over segmentation, that is words containing broken characters, are handled. In Devanagari script, due to uneven strokes in handwritten text, thinning cannot be applied on document as this may lead to incorrect over segmentation of words.

3. **Feature Extraction:**

In order to attain more accuracy in CRS (Character Recognition System), the most important factor is selection of feature extraction technique. The extracted features help to recognize each character set distinctively. Feature extraction is the method of getting information about the object in order to facilitate classification. The feature extraction phase is linked with analysis of text segment and selection of a set of features that can help in finding the unique text segment. It involves various steps for measuring associated shape information contained in a pattern that will make the function pattern classification very easy. In this stage we assign a feature vector which help in its identification. This vector is used to differentiate the character from other characters.

d) **Classification:** In the OCR system, classification is the key decision making stage. In this phase, features are extracted from the data image and given as information to the trained classifier such as Artificial Neural Network, k-NN, SVM, etc.

e) **Post Processing:** Post processing is the last phase of the system and output of the classification phase is used as input in this phase. It is used to further improve recognition rate by using dictionary and other tools.

1.1 Features of Devanagari Script

Devanagari Script has the following features:

- (i) The Devanagari script contains 12 vowels and 34 consonants which are shown in figure 3(a) and 3(b) respectively.

अ आ इ ई उ ऊ ऋ ॠ ए
ऐ ओ औ

Figure 3(a): Vowels are used to produce their own sounds

क	ख	ग	घ	ङ
च	छ	ज	झ	ञ
ट	ठ	ड	ढ	ण
त	थ	द	ध	न
प	फ	ब	भ	म
य	र	ल	ळ	व
श	ष	स	ह	

Figure 3(b): Consonants

- (ii) There is no lowercase and upper case.
- (iii) Writing style is like English that is from left to right.
- (iv) Vowels are used in both ways. Vowels are used to modify the sound of consonants. An appropriate modifier form symbols are attached to the consonant to modify the sound of it. Modifier Symbols corresponding to the vowels are shown in figure 4.

ा ि ी ु ू ृ ्र ै

स सा सि सी सु सू सृ स्र से

ै ो ौ

सै सो सौ

Figure 4: The modifier symbol has also been attached to the consonant स to indicate its placing

1.2 Major Challenges

Following are the challenges of recognition of hand-filled forms in Devanagari script.

1. Large variations in the handwriting of different persons (Table-III).

Table III: Words written by different persons

Word	Samples of Handwritten text
द्वेष	द्वेष द्वेष द्वेष द्वेष द्वेष द्वेष
खरबूजा	खरबूजा खरबूजा खरबूजा खरबूजा खरबूजा
सुफ्फा	सुफ्फा सुफ्फा सुफ्फा सुफ्फा सुफ्फा
छप्पर	छप्पर छप्पर छप्पर छप्पर छप्पर
चिस्टी	चिस्टी चिस्टी चिस्टी चिस्टी चिस्टी
चंडोक	चंडोक चंडोक चंडोक चंडोक चंडोक

2. Non-uniform character styles (Table-IV).

Table IV: A sample of handwritten characters of Devanagari Scripts

अ	अअअअअअअ
इ	इइइइइइइ
ख	खखखखखखख
ग	गगगगगगग
न	ननननननन
य	ययययययय
ह	हहहहहहह

3. Skew detection and correction at form and at word level.
4. Form field data extraction.
5. Segmentation of handwritten words (Figure 5).

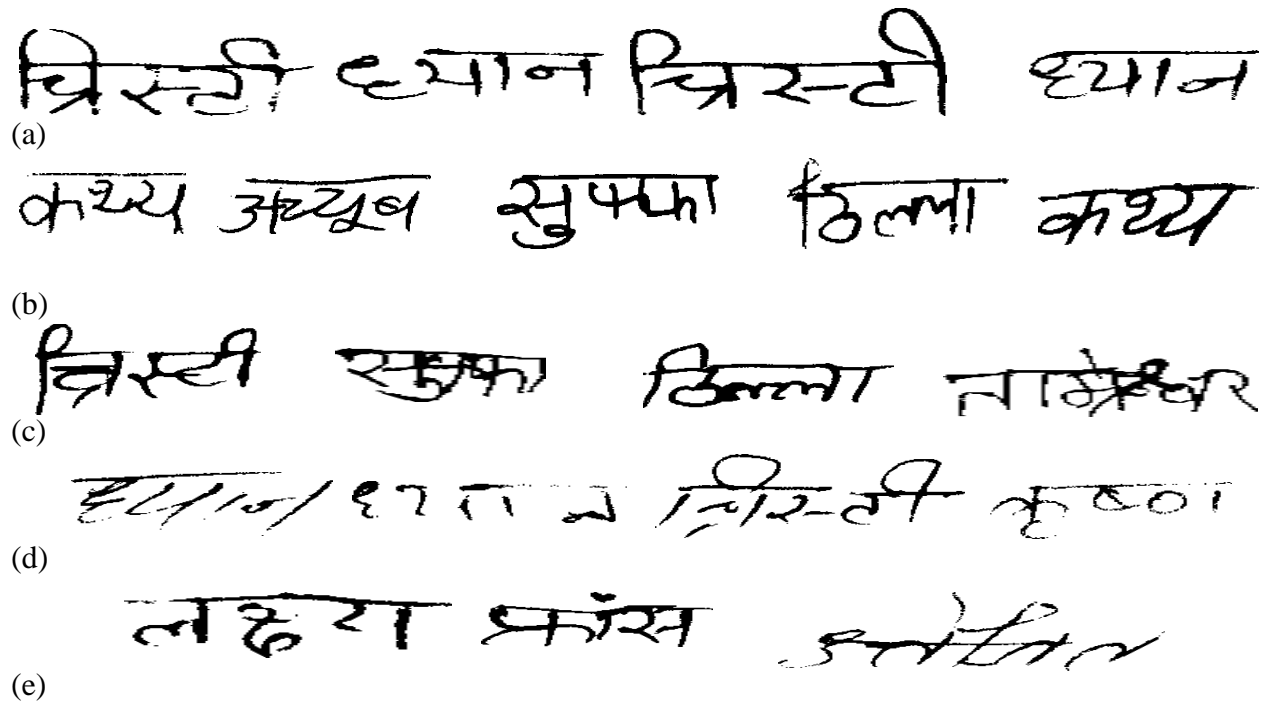


Figure 5: Words with problem of overlapping (a), connected characters (b), merged characters (c), broken characters (d) and other problem (e)

6. Feature extraction and classification of variedly written handwritten character.
7. Refinement of recognition results by applying post processing.

1. FEATURE EXTRACTION METHODS

2.1 Zoning: Zoning based feature extraction is one of the most accepted techniques. For extracting the features from characters the simplest and easy method is zoning. At this stage, original character matrix is initially scaled up or down to regularized or normalized window size that is 48 x 48 and further it divided into zones. In each zone, the density of object pixels is calculated. A feature is computed from each of these zones. Usually, zoning feature is based on the pattern pixels enclosed in that zone. These zones contain some foreground and background intensity pixels. To get the features of whole zone the foreground pixels are added. Similarly, features of all zones are work out. In the corresponding zones, the density of foreground pixels (D) which is divided by the total number of pixels existing in the zone (T). From the set, highest density is computed and density of each zone is divided by this to normalize the features in the range of 0 to 1. Due to writing styles of person is varying, not similar and different stroke width, a method of zoning has been adopted. In this method, 8 x 8 zones are created as shown figure 6, resulting into feature set of size 64 (0 to 63). This method eases the effect of pixel inclusion or exclusion in the zone near boundaries of zone, which is a common problem in handwritten text. Some results of zoning discussed above for 8x8 zones are given in Figure 7.

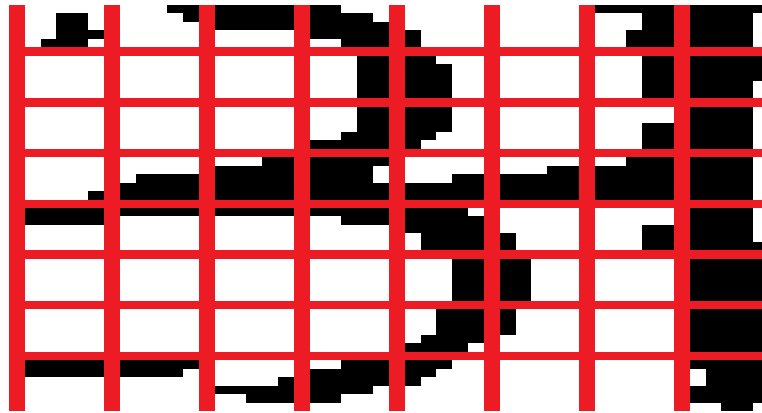


Figure 6: Normalized Character divided into 8 X 8 zones

Figure 7: Normalized Zoning features of scaled images as shown in Figure 6

2.2 DISTANCE PROFILE FEATURE

With the help of profile, the distance (no. of pixels) from bounding box of character image to outer edge of character is determined. The distance so determined can be any direction such as vertical, horizontal or radial. For the purpose of finding top Profiles, a vertical traversing of distance is calculated from top of the bounding in downward direction and similarly bottom profile can be

0.2778	0.0833	0.4722	0.6667	0.1944	0	0.4722	0.8889
0	0	0	0.3333	0.6389	0	0.2778	0.9167
0	0	0	0.5556	0.5278	0	0.2222	0.8333
0.1667	0.5833	0.8611	0.9444	0.5556	0.6667	0.75	0.8333
0.3056	0.1667	0.1667	0.25	0.7778	0.25	0.1667	0.8889
0	0	0	0	0.3333	0.5	0	0.9722
0.1111	0.1667	0	0.0278	0.6389	0.1944	0	0.8333
0.2778	0.3056	0.2778	0.6111	0.25	0	0	0.6944

calculated by changing the direction from bottom of the bounding box in upward direction respectively to outer edges of character.

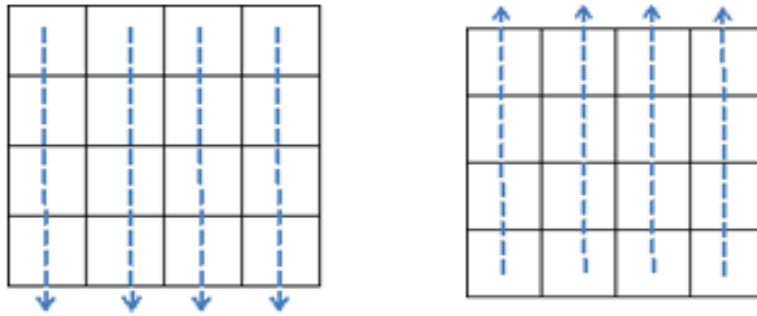


Figure 8: Vertical histogram from Top to Down and Bottom to Up

In the same way, to trace left profiles, horizontal distance is calculated from left of the bounding box in forward direction and to trace right profiles, horizontal distance is calculated from right of bounding box in backward direction.

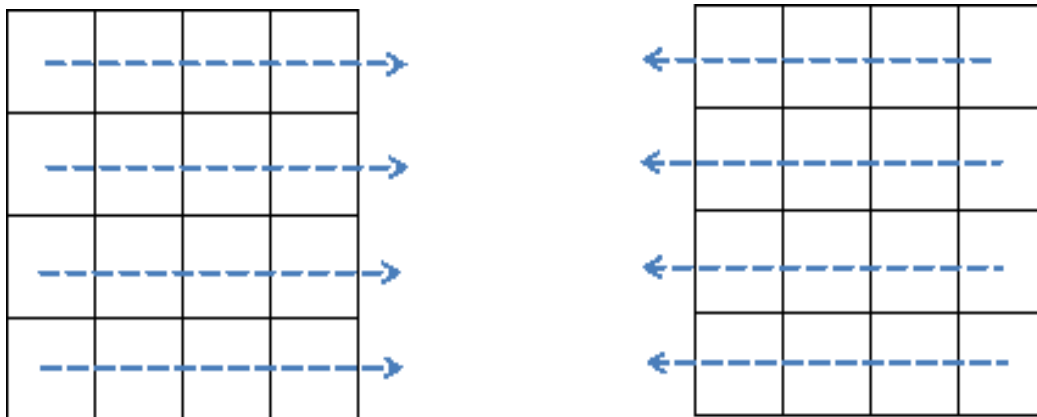


Figure 9: Horizontal histogram from Left to right and Right to Left

Size of scaled image is 48 x 48 used, so total 192 features are generated by all four types of profiles. Some results of distance profile feature discussed above for original image shown in figure 10(a) in and scaled image in figure 10(b) by 48 x 48 and its features are shown in Figure 11 and Table V.



Figure 10: (a) original image 29 x 55 and (b) Scaled image 48 x 48

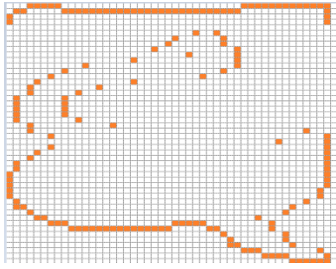
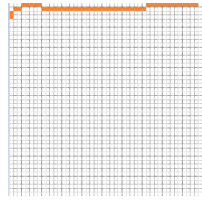
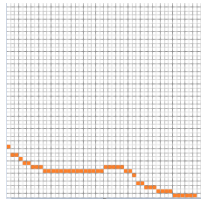
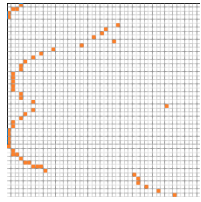
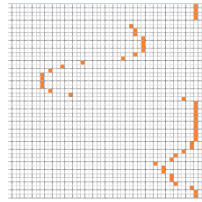
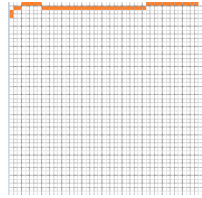
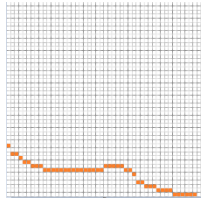
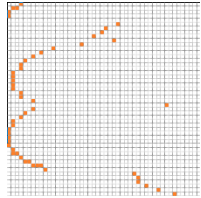
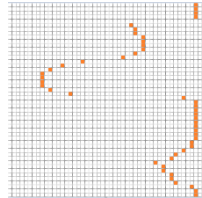


Figure 11 : Distance Profile Features of scaled image 10(b)

Feature no				
0	2	12	3	1
1	1	10	1	1
2	1	10	0	1
3	0	9	0	1
4	0	8	47	46
5	0	8	27	17
6	0	7	24	16
7	0	7	23	16
8	1	7	21	14
9	1	6	26	14
10	1	6	18	14
11	1	6	11	14
12	1	6	8	16
13	1	6	6	19
14	1	6	4	29
15	1	6	3	34
16	1	6	3	37
17	1	6	1	39
18	1	6	1	39

Feature no				
19	1	6	1	39
20	1	6	1	39
21	1	6	1	37
22	1	6	3	32
23	1	6	3	4
24	1	7	6	1
25	1	7	39	1
26	1	7	6	1
27	1	7	4	1
28	1	7	3	1
29	1	6	1	1
30	1	6	1	1
31	1	5	0	1
32	1	3	0	1
33	1	3	0	1
34	0	2	0	2
35	0	2	0	2
36	0	2	1	4
37	0	1	1	6
38	0	1	3	7
39	0	1	4	11
40	0	1	6	9
41	0	0	9	9
42	0	0	31	7
43	0	0	32	7

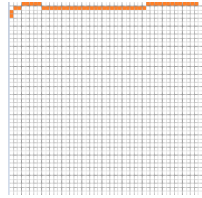
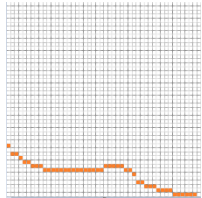
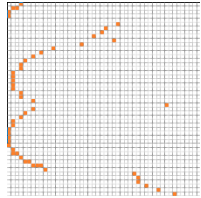
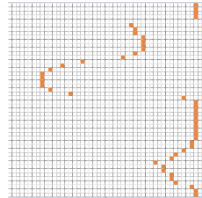
Feature no				
44	0	0	32	6
45	0	0	34	2
46	0	0	37	1
47	46	46	41	1

Table V: Normalized Distance Profile features of scaled image

2. CLASSIFICATION METHOD

In the OCR system, classification is key decision making stage. In this phase, output of feature extraction phase is given as input to the trained classifier such as Artificial Neural Network, k-NN, SVM, etc. Classifiers compare the data features and put away example and figure out the best coordinating class for input. It is the process which is used to classify unknown pattern using training data. It is also used to identify the segmented text by using the extracted features according to preset rules. Then segmented character is identified according to preset rules and assigned to the correct character class. Classification defines the area of feature space in which an unknown pattern falls.

2.1 K-NEAREST NEIGHBOUR

The k Nearest Neighbors (kNN) is very simple but very effective technique in many cases [14]. It is a non-parametric classification method in which, for both the classes, it only requires reference data points. In this method, classification is computed by investigating the training feature vector in the n-dimensional space, here feature size is represent as n or it's working is based on minimum distance from the required (which is query) instance to the training samples to find out the k nearest neighbors. After collecting k nearest neighbors, take simple majority of this k nearest neighbors to be the forecast of the query instance. A test sample is then assigned the same class label as the label of the majority of its k-nearest (reference) neighbors. Weights may be assigned to features for weighted distance calculation. Then between the test points, distance may be calculated by Manhattan distance or Euclidean distance. Manhattan distance is calculated for numeric data and then the local distance function can be defined as the absolute difference of the values:

$$\sum_{i=1}^{j_i} |x_i - y_i|$$

If the global distance is calculated as the sum of these local distances, then it is known as Manhattan distance. Weighted sums and weighted averages are also possible.

Euclidean distance is calculated for non-numeric data and in this all the reference points are considered in order to find k nearest neighbours. The Euclidean distance between an input feature vector Y and a library feature vector F is given by following equation:

$$\sqrt{\sum_{i=1}^N (F_i - Y_i)^2}$$

By using zoning features and kNN Classifier for different values of k, the results at character level for 37593 alphabets are given in Table-VI and based on the results given in Table graphically presented in figure 12.

Value of k	Alphabets			
	Recognition	%age	Error	%age
1-NN	31049	82.59%	6544	17.41%
3-NN	30041	79.91%	7552	20.09%
5-NN	29409	78.23%	8184	21.77%
7-NN	28596	76.07%	8997	23.93%

Table VI: Character level recognition results for zoning features and kNN (with different values of k) for Alphabets

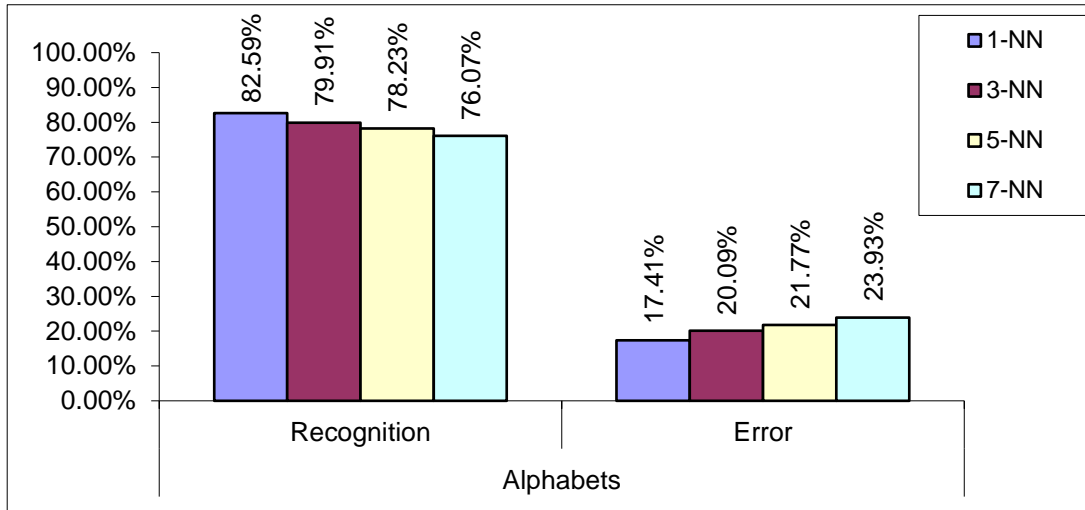


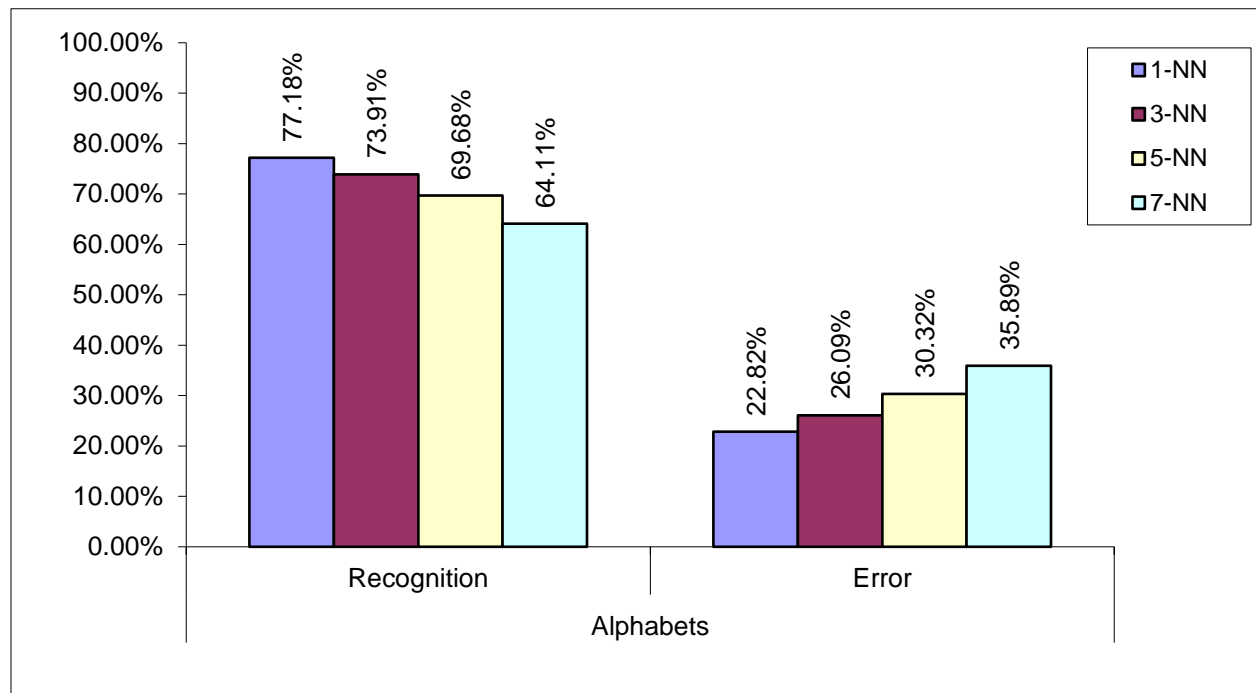
Figure 12: Graphical representation of character level recognition with zoning and kNN classifier (with different values of k) for Alphabets

After the study of results shown in table V, it is clear that the rate of recognition falls as the value of k rises, so they are inversely proportional to each other. The reason behind this is that candidate classes will increase with the increase in the value of k and some other classes are wrongly obtainable as nearest matching classes.

By using Distance Profile Feature with and kNN Classifier for different value of k. the results at character level for 37593 alphabets are given in Table-VII and based on the results given in table, graphically presented in figure 13.

Value of k	Alphabets			
	Recognition	%age	Error	%age
1-NN	29013	77.18%	8580	22.82%
3-NN	27785	73.91%	9808	26.09%
5-NN	26195	69.68%	11398	30.32%
7-NN	24101	64.11%	13492	35.89%

Table VII: Character level recognition results for Distance Profile features and kNN (with different value of k) for Alphabets



3. RESULTS AND CONCLUSION

The achievement rate of the algorithm depends upon input data. If the headline of word is not straight and skewed at single angle, the achievement rate of this algorithm is more than 98% after graphic assessment between the input and output images but if the word, as shown in figure 2, has multi-skewed headline, bended headline, headline not at proper position, broken headline, etc. this algorithm unable to accurate the skewness of these type of words. These are in fact the cases when words are not actually skewed but suffer from other types of irregularities. Based on the results given in table VI and table VII, value of k has been taken as 1, which produces the best results as zoning feature i.e. 82.59% and for Distance Profile Feature it gives 77.18%. There is a need to extend this work from character level to word level and then up to sentence level. There are numerous feature extraction approaches are available which are not directly applied in case of handwritten Devanagari script recognition. So, a lot of work can be done in field of Handwritten Devanagari Character Recognition.

REFERENCES

- [1] Garain, U.; and Chaudhuri, B. B. "Segmentation of Touching Characters in Printed Devnagari and Bangla Scripts using Fuzzy Multifactorial Analysis", in the Proceedings of the 6th International Conference on Document Analysis and Recognition (ICDAR'01), Seattle, United States, 2001, pp. 805-809.
- [2] Bansal, V.; and Sinha, R. M. K. "Segmentation of Touching and Fused Devanagari Characters", Pattern Recognition, 2002, Volume: 35, pp. 875-893.

- [3] Lehal, G. S.; and Singh, C. “A Gurmukhi Script Recognition System”, in the Proceedings of the 15th International Conference on Pattern Recognition (ICPR), 2000, Volume: 2, pp. 557-560.
- [4] Lehal, G. S.; and Singh, C. “Text Segmentation of Machine Printed Gurmukhi Script”, Document Recognition and Retrieval VIII, Proceedings SPIE, USA, 2001, Volume: 4307, pp. 223-231.
- [5] Lehal, G. S.; and Singh, C. “A Technique for Segmentation of Gurmukhi Text”, Computer Analysis of Images and Patterns, in the Proceedings of the Computer Analysis of Image and Patterns (CAIP), 2001, W. Skarbek (Ed.), Lecture Notes in Computer Science, Volume: 2124, Springer-Verlag, pp. 191-200.
- [6] Lehal, G. S., Optical Character Recognition of Machine Printed Gurmukhi Text, Ph. D. Thesis, Punjabi University, Patiala, India, 2001.
- [7] Pal, U.; and Datta, S. “Segmentation of Bangla Unconstrained Handwritten Text”, in the Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR’03), Edinburgh, Scotland, 2003, pp. 1128-1132.
- [8] Badsha, Md. A.; Ali, Md. A.; Deb, K.; and Bhuiyan, Md. N. “Handwritten Bangla Character Recognition Using Neural Network”, International Journal of Advanced Research in Computer Science and Software Engineering, 2012, Volume: 2, Issue: 11, pp. 307-312.
- [9] Shrivastava, S. K.; and Gharde, S. S. “Support Vector Machine for Handwritten Devanagari Numeral Recognition”, International Journal of Computer Applications, 2010, Volume: 7, Issue 11, pp. 9–14.
- [10] Gupta, D.; and Nair, L. M., “Improving OCR by Effective Pre-Processing and Segmentation For Devanagiri Script: A Quantified Study”, Journal of Theoretical and Applied Information Technology, 2013, Volume: 52, Issue 2, pp.142-153.
- [11] Jhaji, P.; and Sharma, D. “Recognition of Isolated Handwritten Characters in Gurmukhi Script, International Journal of Computer Applications (IJCA), 2010, Volume: 4, Issue: 8, pp. 9-17.
- [12] Singh, P.; and Budhiraja, S. “Feature Extraction and Classification Techniques in O.C.R. Systems for Handwritten Gurmukhi Script – A Survey”, International Journal of Engineering Research and Applications (IJERA), 2012, Volume: 1, Issue 4, pp. 1736-1739.
- [13] Sharma, D. V.; and Lehal, G. S. “A Fast Skew Detection and Correction Algorithm for Machine Printed Words in Gurmukhi Script’, in the proceedings of International Workshop on Multilingual OCR, Barcelona, Spain, 2009, Article no 15.

[14] Fulcher, J. "Character Recognition", Handbook of Neural Computation, IOP Publishing Ltd and Oxford University Press, 1997.

[15] Sachdeva, R., & Sharma, D. V. , "A Brief Study of Feature Extraction and Classification Methods Used for Character Recognition of Brahmi Northern Indian Scripts. International Journal of IT, Engineering and Applied Sciences Research (IJIEASR),2015, Volume 4, Issue 2, pp 25-29.

[16] Khanna, N., Dhiman, S., Sachdeva, R., Kumar, S. "Comparative Analysis for Gurmukhi and Devanagari Script at Word Level", International Journal of Engineering Technology and Computer Research, 2017, Volume 5, Issue 3, pp 49-54.

[17] www.ijera.com

[18] Sachdeva, R., & Sharma, D. V. "Data extraction from hand-filled form using form template", International journal on recent and innovation trends in computing and communication, 2015, Volume 3, Issue 8, pp 5311-5317.