CHARACTER SEGMENTATION FOR HANDWRITTEN BANGLA WORDS USING ARTIFICIAL NEURAL NETWORK

T. K. Bhowmik [#] , A. Roy [*] and U. Roy [*]

IBM Global Services Pvt Ltd, Embassy Golf Link, Bangalore - 560 071, INDIA.
tbhowmik@in.ibm.com
*Dept. of Computer and System Sciences, Visva-Bharati, Santiniketan 731235, INDIA.
uroyin@yahoo.co.in

Abstract—
Character segmentation is a necessary preprocessing step for character recognition in many handwritten word recognition systems. It is important because incorrectly segmented characters are less likely to be recognized correctly. The scripts those are cursive in nature is difficult to segment. Bangla as well as almost all other Indian language have this feature in common. That is why fully cursive nature of Bangla handwriting as well as the natural skewness in words poses some high challenges for automatic character segmentation. In this article segmentation scheme of Bangla handwritten words have been proposed with some unconventional and easy-to-implement technique. An area based algorithm has been proposed for the skew detection of the Bangla specimen skewed handwritten words available in the self-prepared database. The features have been extracted for segmentation mainly by the analysis of directional chaincode as well as its positional information. Finally segmenting points have been recognized through Multilayer Perceptron (MLP) Neural Networks.

Keywords— Skew Detection, Character Segmentation, Multilayer Perceptron.

1. INTRODUCTION

Character segmentation is a necessary preprocessing step for character recognition in many handwritten word recognition systems. The most difficult case in character segmentation is the cursive script. Fully cursive nature of **Bangla** handwriting poses some high challenges for automatic character segmentation**.** In literature basic segmentation algorithms can be classified into three main categories: region-based, contour-based, and recognition-based methods. Although many methods on handwritten character segmentation have been published in the literature for different scripts [1,2], at the best of our knowledge only two reports are available on **Bangla** handwritten scripts [3, 4].

Apart from segmentation, skew estimation and correction is another important step in any document analysis and recognition system. A wide variety of skew detection algorithms have been proposed in literature. Most of the algorithms are based on Hough transform [5] and projection profiles [6]. Algorithms based on feature point distribution [7], run length analysis [8] has also been proposed in literature.

In this study segmentation has been done based on some characteristics observed in Bangla handwritten words. In the pre-segmentation stage we have proposed an area-based algorithm to disskew the word image.

2. CHARACTERISTICS OF BANGLA HANDWRITING

A detailed study of handwritten Bangla words has shown that most of them have a long horizontal run called a '*Matra' (Headline)* and somewhat identifiable '*Baseline*'. The Matra represents the boundary of upper and middle zone, and the baseline indicates boundary of middle and lower zone. Hence if we can detect the Matra and the baseline then we can immediately segregate the entire image into three zones namely *upper, middle* and *lower*. The correct detection of Matra facilitates the segmentation process, because in Bangla words segmentation is done mostly along the Matra line. However the problems with handwritten words are that the writer may not include such a long Matra (Fig 1). In that case it is difficult to detect the Matra line. So we omit the detection of actual Matra, instead we detect the baseline and the headline.

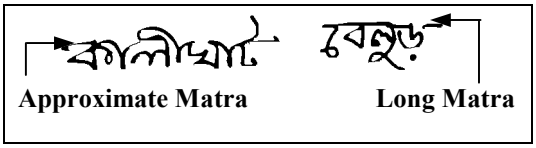


Fig. 1: Examples of Bangla Words

Generally in Bangla handwritten words any two consecutive characters are connected at the upper portion of the word. In most of the cases, at the intersection points of two successive characters and their connecting line, one of the following situations (Fig. 2) may arise along lower contour.

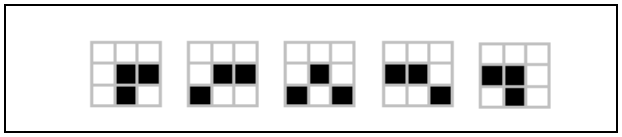


Fig. 2: Patterns observed in Bangla Words

Further in Bangla word generally a segmented region is bounded by two long vertical strokes. Thus vertical histogram analysis shows that local minima of vertical histogram lie in the segmented region, as shown in Fig. 3.

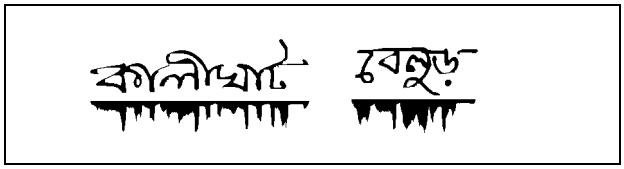


Fig. 3: Words with Vertical Histogram

Finally when considering the vertical black pixel run, it is observed that the frequency of black pixel run in segmented region is lower than that of nonsegmented region.

3. SKEW DETECTION AND CORRECTION

Prior to segmentation it is necessary to preprocess all word images. Initially the images are in gray-level format. The gray image is median filtered and then Otsu's thresholding algorithm [9] is used to binarize the images.
Document skew is a distortion that often occurs during document scanning or copying. This mainly concerns the orientation of text lines, and with no skew the lines are horizontal or vertical, depending on the language. This effect visually appears as a slope of the text lines with respect to the X-axis. Document skew is an unavoidable effect because of the complex structure of handwritten words and the copying/scanning process, especially when digitizing automatically huge document bulks. The skewed images ought to be disskewed for proper segmentation. However it has been found that most normal handwritten words do not exceed range of skew angles ($\pm 10^0$), so our system concentrates on words within a document, which are within positive 10^0 to negative 10^0.

Hough transform has been used by Srihari and Govindaraju [5] for skew detection. The basic method consists of mapping points in Cartesian space (x, y) to sinusoidal curves in (ρ, θ) space via the transformation

$$\rho = x\cos\theta + y\sin\theta$$

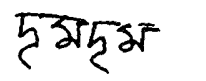
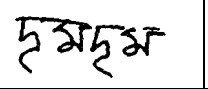
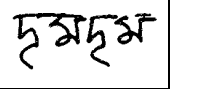
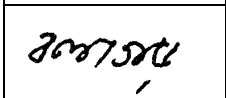
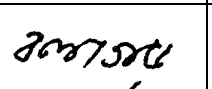
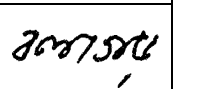
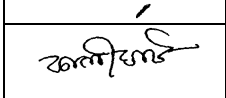
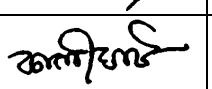
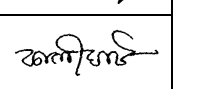
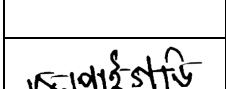
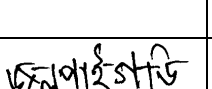
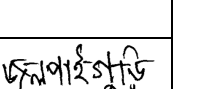
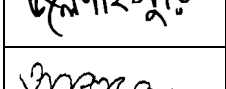
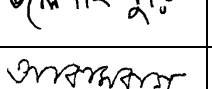
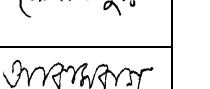
Each time a sinusoidal curve intersects another at a particular value of ρ and θ, the likelihood increases that a line corresponding to that (ρ, θ) coordinate value is present in the original image. An accumulator array is used to count the number of intersection at various ρ and θ values. The skew is then determined by the θ values corresponding to the highest number of counts in the accumulator array. However in case of handwritten words the skew detected by conventional Hough transform technique may be not so accurate. Here we have proposed an area-based algorithm, which is imposed to usual Hough, transform technique. As a result efficiency of the Hough transforms increases to a considerable amount. The algorithm is described bellow.

ALGORITHM SKEW:

1. Initialize $\hat{\theta}$ to $\left(\frac{\pi}{2}\right)$. Calculate $area_{old} = area\ of\ bounding\ box$
2. Apply Hough transform to the binary image for estimating the values of the accumulator array. Impose skew correction to the image, based on the angle θ detected by Hough transform.
3. Calculate the area ($area_{new}$) of the bounding box of the image modified in **Step 2**.
4. IF $area_{new} < area_{old}$ THEN
   $area_{old} = area_{new}$ and $\hat{\theta} = \theta$
5. Let $x_{i,j}$ be the $\langle i, j \rangle^{th}$ element of the accumulator array and θ be the angle corresponding to $x_{i,j_{\max}}$.
   IF $\left(x_{i,j_{\max}} - x_{i,j}\right)^2 < \varepsilon$ THEN
   Calculate area of the image corrected by an angle θ corresponding to $x_{i,j}$.
   IF $area_{new} < area_{old}$ THEN
   $area_{old} = area_{new}$ and $\hat{\theta} = \theta$
   Repeat **Step 5** $\forall (i, j \neq j_{\max})$
6. The angle $\hat{\theta}$ corresponding to the minimum area is chosen as the optimum skew angle.

Table 1 shows the results obtained from our algorithm for several sample skewed images placed in first coloum of the table. The comparison shows that the algorithm produces better result that could be obtained from usual Hough transform.

Table 1. Comparison between Hough Transform and present algorithm

Original Image	Result of Hough Transform	Result of Area-Based algorithm
		
		
		
		
		

4. PROPOSED SEGMENTATION TECHNIQUE

The disskewed images have been made available for segmentation process.

In our implementation the approximate baseline and headline has been detected by analyzing the horizontal projection as well as the histogram of horizontal black pixel run. Finally the baseline has been refined using the least square method. The detected middle zone, the headline and the baseline, also the horizontal projection for our examples are shown In the Fig. 4.

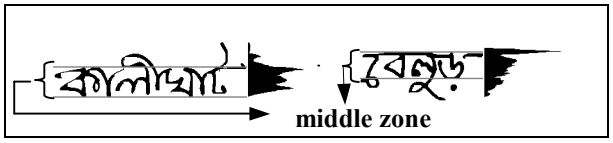


Fig. 4: Middle Zone and Horizontal Projection

To segment characters from a word we first detect isolated and connected characters within a word. Then the connected components are segmented into individual characters. In both training and testing phases, a feature detection algorithm was used to locate probable segmentation points in a word. The overall algorithm is given below—

STEP 1: The gray level image of a Bangla word is median filtered and then converted into a binary image using Otsu threshold technique.

STEP 2: Apply the skew detection and correction algorithm and then detects the headline and baseline as well as the bounding box.

STEP 3: Connected components of a word to be segmented are detected.

STEP 4: Lower contour of each connected component is traced anticlockwise. During this tracing process the relevant features are extracted.

STEP 5: the feature vectors are normalized. Also the MLP is trained with the normalized feature set.

4.1. FEATURE EXTRACTION

For feature extraction the lower contour of the word image is traced anticlockwise. Suppose $x_t = x_t(i, j \mid ij^{th} pixel)$ and d_t be the contour points and corresponding 8-directional codes respectively.

The index t has been increased sequentially with the progress of tracing. The left and right touching points of the word image with its bounding box have been taken as the starting and ending points of tracing. Considering all our previous observations, we construct a feature vector of length ((2L + 1) + 3) as:

$$f_L = (D_L, P_i, H_V, H_{Run})$$

where,

$$D_L = \{d_{i-l}, d_{i-l+1}, \ldots, d_{i-1}\} \cup \{d_i\} \cup \{d_{i+1}, d_{i+2}, \ldots, d_{i+l}\}$$

Sets $\{d_{i-l}, d_{i-l+1}, \ldots, d_{i-1}\}$ and $\{d_{i+1}, d_{i+2}, \ldots, d_{i-l}\}$ are the corresponding 8-directional values of the previous and next contour points of x_i respectively. We have generated a feature vector for those points whose lower bound and upper bound cardinalities are greater than or equal to L, excluding that point.

Since the characters are joined at the upper portion, the positional information P_i has also been included.

$$P_i = \frac{height\ of\ x_i\ with\ respect\ to\ middle\ zone}{height\ of\ the\ middle\ zone}$$

$$H_V = \frac{value\ of\ vertical\ histogram\ of\ x_i}{height\ of\ the\ middle\ zone}$$

$$H_{Run} = value\ of\ black\ pixel\ run\ of\ x_i$$

To make the above features suitable as input to an MLP classifier we further normalize them. The normalized feature vector is

$f_L{}^N = (D_L/8, P_i, H_V, H_{Run}/\zeta)$ where ζ is a predefined constant.

4.2. TRAINING PHASE OF MLP

Use of Artificial Neural Networks (ANN) in handwritten character recognition becomes popular because ANN tools performs efficiently when input data are affected by noise. In our approach we feed the feature vector of length ((2L+1)+3) into suitable MLP classifier for segmentation purpose.

Segmentation is basically a two-cluster problem. All the contour points are either a segmenting point or a nonsegmenting point. According to a program we at first manually categories the feature of each contour point for each connected component into two classes.

The class containing features of segmenting points is further splited into two classes. One of them consists of information for those points following some pattern at the lower contour of the word. Another class contains features of segmenting points for which no pattern is followed.

On the other hand the features of nonsegmenting points are also divided into two classes. One class contains those features, for which some pattern indicating segmentation is present, yet the corresponding point is nonsegmenting. The other class contains the extreme contour points of the main portion of a character.

Our main objective behind this classification is that, the classes for segmenting points can be used further to refine the headline of the word, and thus to improve segmentation. Also the baseline can be corrected with the help of the class for extreme points of each character.

4.2. TESTING PHASE OF MLP

Following MLP training, the words used for testing are also segmented using a feature-based algorithm. However for testing purpose, there is no manual processing. The features are extracted automatically and are then fed into a well-trained MLP.

4.3. POST PROCESSING

The segmentation algorithm tends to over-segment the characters, i.e., certain characters are split into sub-characters. This is due to the fact that there are certain simple shapes in the body of some characters that resemble other characters.

In post processing the class of segmenting points generated by the MLP is analyzed to refine our results. The point, bounded by points belong to the class of segmenting points with no pattern followed (Class Two), is taken as the refined segmenting point. Thus by collecting such points the headline is refined according to the baseline detection algorithm. The baseline is also refined according to the knowledge we gained by analyzing the class of extreme upper points. So finally we can construct the proper bounding box for a word.

5. RESULTS AND DISCUSSION

We have simulated our scheme in a moderately large database of Bangla handwritten words. The database consists of the popular district and town names of West Bengal. We have collected the specimen of handwritings from the writers of various levels of literacy and profession. Obviously the handwritten words have some expected skew angle, which is now to be corrected.

The novelty of our area based algorithm resides on the fact that the area of the bounding box of a disskewed image is less than that that of a skewed image. It has been observed that sometimes the Hough transform produces more skewed image instead of correcting the skew. To circumvent this problem we calculate the areas for each value in the neighborhood of the highest value in accumulator array and extract the skew angle for which the area of the bounding box is the smallest one. Now disskewed images are available for segmentation.

We mainly vary the feature vector size, by choosing different L, and tested the scheme through a MLP. In our MLP structure we have one hidden layer and four output layers representing the four previously defined classes. The number of input layers, however, is the feature vector size ((2L+1)+3) and thus depends upon L.

The experimental results are presented in Table 2.

Table 2. MLP results for segmentation

L	Feature vector size = number of Input Nodes	Classification rate (in percent)
5	14	87.4
8	20	88.1
10	24	87.3
15	34	87.5

The experimental results shows that when the feature vector size is 20, that is our L is 8 we have experienced the best result with 88% accuracy, feature vectors of size 14, 24 and 34 also give reasonably good result.

A significantly improved segmentation result can also be expected if this segmentation technique is combined with the recognition process in a holistic system. Furthermore

this methodology may be applied successfully for segmentation and recognition of many other Indian languages like Hindi, Marathi, Gugrathi etc. for postal and office automation this scheme is useful in nowadays systems.

6. ACKNOWLEDGEMENTS

Authors acknowledge Prof. S.K.Parui, CVPR unit ISI kolkata, for his suggestions and reading the manuscript carefully.

7. REFERENCES

1. Casey, R. and E. Lecolinet, A Survey of Method and Strategies in Character Segmentation, *IEEE Transaction on PAMI, 18(7), pp. 690-706, 1996.*
2. Lu, Y. and M. Shridar, Character Segmentation in Handwritten Words – An Overview, *Pattern Recognition, 29(1), pp. 77-96, 1996.*
3. U. Pal and Sagarika Datta, Segmentation of Bangla Unconstrained Handwritten Text, *Proc. of the 7th International Conference on Document Analysis and Recognition (ICDAR 2003).*
4. A. Bisnu and B. B. Chaudhuri, Segmentation of Bangla handwritten text into characters by recursive contour following, *Proc. 5th ICDAR, pp. 402-405, 1999.*
5. S.Srihari, V.Govindaraju, Analysis of textual images using the Hough transform, *Machine Vision and Applications, pp 141-153, 1989.*
6. H.S.Baird, The skew angle of printed documents, *Proc. SPSE 40th symp. Hybrid imaging systems, pp 21-24, 1987.*
7. H.S.Baird, The skew angle of printed documents, *Proc. of the Society of Photographic Scientists and Engineers, Rochester, New York, 1987,14-21.*
8. Z.Shi, V.Govindaraju, Skew Detection for Complex Document Images Using Fuzzy Run length, *Proc. Of the Seventh Int. Conf. on Document Analysis and Recognition, ICDAR'03.*
9. N.Otsu: A thresholding selection method from graylevel histogram, *IEEE Transactions on Systems, Man, and Cybernetics, 9 (1979) 62–66.*