

An Algorithm for Segmenting Modifiers from Bangla Text

Nasreen Akter¹, Saima Hossain¹, Md. Tajul Islam¹, and Hasan Sarwar²

¹Military Institute of Science and Technology (MIST), Dhaka, Bangladesh

²United International University (UIU), Dhaka, Bangladesh







Email addresses: kona_mir@yahoo.com, erin_saima@yahoo.com, mdhasan70@yahoo.com














Abstract — Script segmentation is an essential and preprocessing task for any OCR system. Bangla is one of the most popular scripts in the world. Since segmentation effects the recognition process, accurate and proper segmentation is necessary to implement Bangla OCR. Many works have been done for both handwritten and printed Bangla text. This paper presents the segmentation process of different bangla modifiers from printed bangla words.

Index Terms — Segmentation, Optical Character Recognition, Bangla Modifier.

I. INTRODUCTION

The structure of bangla language is quite different than any other European language. It consists of 50 basic characters including 11 vowel and 39 consonant characters. Again, there are vowel and consonant modifiers, touching and compound characters which are formed by touching the adjacent characters and by combining 2 or more characters in the word accordingly. So when they are used to form words, we find that thousands of various combinations (both simple and complex) are formed. Segmentation of these complex parts are extremely difficult task. As a result, printed Bangla OCR still remains researchable.

Generally Bangla texts are divided into three zones namely upper zone, middle zone and lower zone [1], [2], [4], [6], [7]. Upper zone contains , , . Middle zone contains the basic character symbols and the lower zone contains the lower modifiers such as , , , etc. Detection of text line, matra line and base line [1], [2], [4], [6], [7] lead towards the character segmentation process. If one of the above three is ignored, it will generate many errors. So it is necessary to detect them accurately.

It is observed that Bangla words have different kinds of complexities like , ,  etc. Some works have been done to solve a few of above the problems such as , , , , ,  [1], [2], [4], [6]-[8]. A, font size independent, more efficient technique is proposed in this paper, which can perfectly segment , , , .

II. LITERATURE REVIEW

Segmentation of a Bangla document into lines, words and words into individual characters is an important and preprocessing task for Bangla OCR. The line segmentation of a document is solved by finding white pixels between two consecutive headlines [1], [2], [4]-[7]. Matra zone detection has been done with the approach that matra line will be that row which contains maximum no of black pixels than any other portion of the text [4], [6], [7] and considering multiple rows that defines matra zone [1], [3]. Baseline is detected by selecting that row which contains highest lowermost points of the components [1,3].

Word segmentation is done by finding some couple of 0 consecutive columns (white pixels) as a word delimiter [5], [6], or n (taken experimentally) consecutive 0 columns [3], or first two 0 consecutive columns [4] or the midpoint of a run of at least k consecutive 0 columns [1], [7].

















The most difficult and important part is to segment the characters individually from the words. At first, the adjacent characters are separated by finding the 0 columns between them [1]-[5], [7]. But this process is not sufficient at all due to the problems of overlapping (multiple characters which do not touch each other but enter into the zone of each other like  ( )), touching and compound situations. Even the situation becomes more complex due to many peculiar patterns of the bangla characters and their placement in the word. Sometimes a character () gets divided into sub portions in the character segmentation process and the problem is overcome by joining them to each other [1]. As there are many modifier, , ,  are segmented as matra upper portion character like , ,  [1], [4], [7] and , ,  are segmented after the detection of base line [1], [2], [4], [7] or considering the height, which exceeds the average size of the segmented characters [6],  is picked up as it makes a regular angle with the matra line [2] and  is picked up as it is not touching the matra line [2]. A piecewise linear scanning [1], [3], DFS (Depth First Search) [7], a combination of flood fill and boundary fill algorithm [4] were proposed when the characters overlap with other characters. In paper [2], they detect  when they find that the present

TABLE I
SUMMARY OF LINE DETECTION TECHNIQUES

Type	Papers							
	1	2	3	4	5	6	7	Our process
Textline detection	Y	Y	Y	Y	Y	Y	Y	Y
Matraline detection	Y	Y	Y	Y				Y
Base line detection	Y		Y				Y	Y
Upper portion of matraline detection			Y					Y
Lower portion of matraline detection	Y	Y	Y	Y				Y

character size is greater than the regular size and apply the segmentation process.

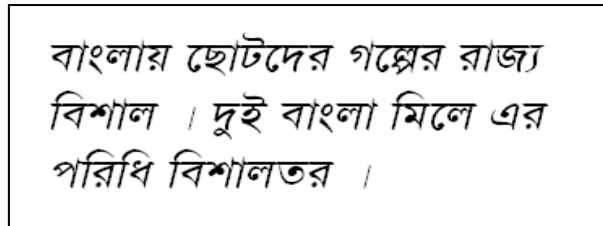
III. SEGMENTATION PROCESS

This is the most vital and important portion for designing an efficient Bangla OCR because character recognition process depends on this phase to make the recognition process successful. In this paper, the following steps are considered to do the segmentation with the existing techniques along with our approach. The steps are:

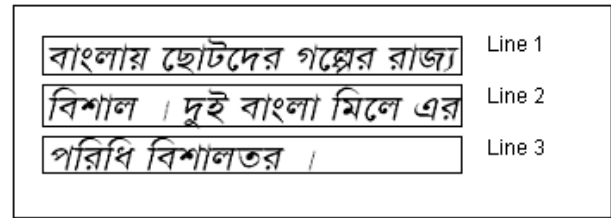
- Line Segmentation Process
- Baseline Detection Process
- Word Segmentation Process
- Character Segmentation Process
 - Words with only basic character
 - Words with modifier ি ি ি
 - Segmentation of মি, টি, ঠি
 - Segmentation of জী, চী, ঠী
 - Segmentation of কী, ঠী, ঠী
 - Words With lower Modifier ৲, ৳, ৴

A. Line Segmentation Process

The lines of a text block are segmented by finding white pixels between two consecutive matra zone [1], [2], [4]-[7]. Fig: 1 shows the result.



(a)



(b)

Fig. 1. Text line detection process (a) full image, (b) result of text line detection

B. Baseline Detection Process

Base line is the row where the middle zone ends and lower modifier zone starts i.e. a separator between middle and lower zone. This is the row where an abrupt change occurs between the previous and next row [7]. Detection of base line is very important.

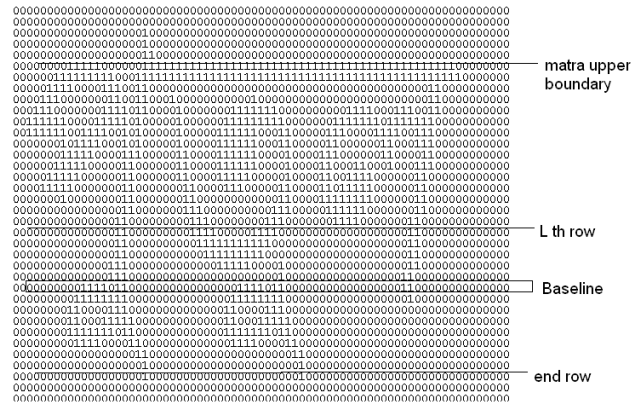


Fig. 2. Process of baseline detection

We proceed with the idea, shown in fig: 2, that there are small no of black pixels at the end portion of the middle zone and from there lower modified zone starts with increasing black pixels. So at first we detect L th row which denotes the lower position of the middle zone. Then we scan from L to end row to find where the black pixels start to increase. That row will be denoted here as the base line.

The difference between the end row and the base line is the lower modified zone. With this technique, sometimes the lower portion of the middle zone may enter into this zone. We can ignore them since they do not touch end row. Lower modifiers always touch end row. Table I compiles the various line detection capability of different works.

C. Word Segmentation Process

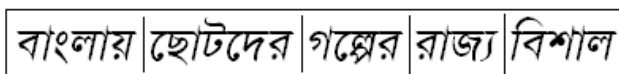


Fig. 3. Process of word segmentation

There are always some white spaces between two words in a text line shown in fig: 3. Vertical scan is done ignoring the matra zone.

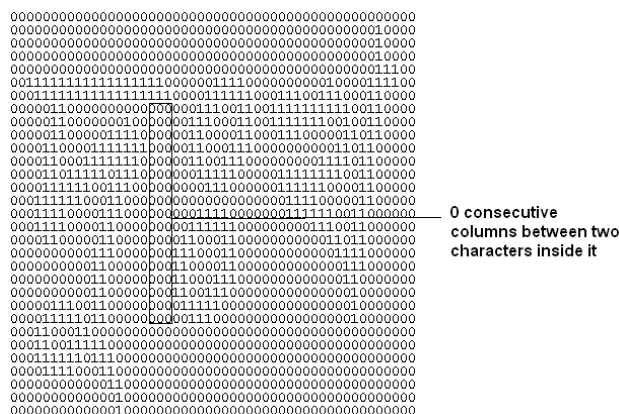


Fig. 4. Word with three 0 consecutive columns between two characters inside it

Sometimes it is also seen that some 0 consecutive columns are formed between two characters in a word shown in fig. 4. Most of the time it occurs when the matra less character is used in a word. If there is a white vertical column then it is treated as a gap between two characters in a word [2]. But this gap is not fixed and can increase with the increase of font size. So these issues require care during word segmentation. We try to solve it with the following technique.

During word segmentation, we start scanning vertically from start row to end row and keep the record of the width

of 0 consecutive columns for each line. Then we take the average of the total widths. If any width is less than half of average, then it is considered as a gap between two characters and easily ignored. Table II shows the capability of different papers considering the above problems.

TABLE II
SUMMARY OF WORD SEGMENTATION

Type	Papers						
	1	2	3	4	5	6	7
Word segmentation	Y		Y	Y	Y	Y	Y

D. Character Segmentation Process

It is the most difficult and challenging part for building printed Bangla OCR. Many works have been done in order to segment the bangla characters from the words. Different problems are discussed below and some algorithms are made to solve them.

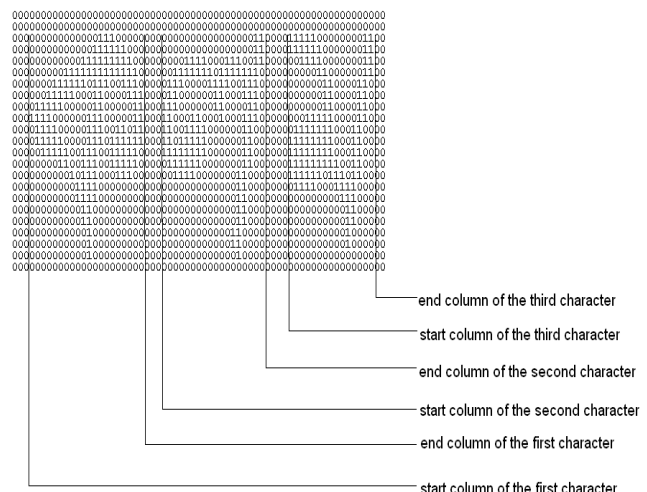


Fig. 5. Character segmentation

Words with only basic character

The basic idea is that, under the lower boundary of the matra zone, the first black pixel column is fetched and is treated as a starting boundary and then a column of white pixel is searched. When it is found, it is treated as a separator from the next character [1]-[5], [7]. It is shown in fig: 5. The works that use this technique for character separation as well as ours are mentioned in Table III.

TABLE III
SUMMARY OF WORDS WITH SINGLE CHARACTERS

Simple	Papers						Our process
	1	2	3	4	5	7	
কলম	Y	Y	Y	Y	Y	Y	Y
ঐষধ	Y	Y	Y	Y	Y	Y	Y

Some times this technique fails to segment the characters properly [1]-[7]. Extra care is needed to solve those problems. They are discussed below.

Words with modifier ি, ি, ি

Most of the previous works considered the segmentation of these modifiers as matra upper portion character. In our system মি, জী, টি, ঠি, টী, ঠী, কাঁ, টাঁ, ঠাঁ are segmented properly and efficiently keeping their original shape. The technique is given below.

1) In order to decide whether the current portion contains basic character or character with ি, ি, ি, the system finds first discontinuity point i, e. first white column. Then it searches in the top of the matra zone to get whether the same column contains black pixel or not. If it gets that column of black pixel, then it goes right to end and searches a column of white pixel in the matra upper zone. If gets that column, it searches for the same column contains white pixel under the matra zone. If it gets that column then it will be a basic character. Otherwise, along with not getting the column of white pixels in the matra upper zone, it will be the character with above modifiers.

2) Then the thickness of the characters x and y is determined shown in fig: 6.

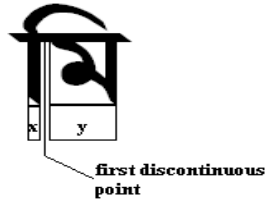


Fig: 6. Determination of the thickness of each character in the syllable

3) If $x < y$ then there exists ি. For ি, we will get the following situations. They are like মি, টি/ঠি etc. From the immediate upper row of the matra upper boundary we go right to fetch black pixel. After finding that we again go right to find white pixel. After getting that we continue to go to the same direction. During that time if there exist all 0 in that row, the system can understand that the character, which is under the matra zone, has no upper

part. If it is not, then the system can understand that the character has upper part.

a) For the first case, to separate ি it is picked up by removing the adjacent character and that adjacent character is picked up by removing ি. The result is shown below:

$$\text{মি} = \text{মি} + \text{মি}$$

(b) For later case, to separate ি, it is needed to detect whether it is টি or ঠি.

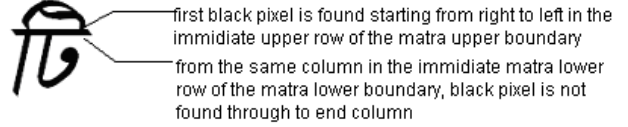


Fig: 7. Segmentation process of টি

At first, first black pixel is searched in the immediate upper row of the matra upper boundary from the end column to left. If it is found, then from the same column of the immediate lower row of the matra lower boundary, next searching is occurred, which lasts at the end column, for finding black pixels. If it is found then there is ঠ otherwise there will be টি[2]. Fig. 7 shows the following process.

i) Segmentation of টি: Form the start row we scan right to get the first black pixel and after getting that we start to pick up those. When we get the first white pixel i, e. 0, we pick up the rest of the image as 0. In this way we pick up the matra upper portion. Matra zone remains identical. Under the matra zone, the first character is picked up by removing the adjacent character. Then the remaining character is picked up from the upper part of matra zone by making the black pixels to white, which was unchanged during the previous one and the rest of the pixels remain unchanged. Under the matra zone, the second character is picked up by removing the previous character. Here, removing means making the 1(black pixel) to 0 (white pixel). But this is not actually done. The result is shown below.

$$\text{টি} = \text{মি} + \text{টি}$$

ii) Segmentation of ঠি: At first the thickness of upper portion of ঠি and a gap are calculated. The system counts the black pixels where the gap becomes zero. Then ঠি is separated by starting to find the start position where summation of the black pixels is greater than the thickness and removing the adjacent portions from the upper zone and as well as from the middle zone. ঠি is separated by picking up the thickness portion and the middle zone portion,

removing rest of the pixels. The result is shown below:

$$\text{ঐ} = \text{ি} + \text{ঔ}$$

4) If $x > y$ then there exists ঐ or ঔ. For ঐ or ঔ we will get the following situations. They are like ঐ, ঔ, ঐ, ঔ etc. From the startrow if we find that the sum of the black pixels is greater than 3 then there will be ঐ and if not Then ঔ remains there. For the above technique, for ঐ, ঐ is detected as ঐ. We solve this problem later. In the case of জী, টী- From the startrow to half of the upper zone we calculate the maximum gap (no of sequence of 0s between two 1s) between the black pixels.

a) *Segmentation of জী* - If only one gap is found then the system can understand that, the basic character does not have upper part. Then to separate ঐ, it is picked up by removing the adjacent character and that adjacent character is picked up by removing ঐ. The result is shown below.

$$\text{জী} = \text{জ} + \text{ঐ}$$

b) *Segmentation of টী* - From the start row to the matra upper boundary the area is divided into two halves. In the upper half, the system starts from left to right to find the cutting point (the white column after getting the first black pixel). After getting that the rest of the pixels will be removed. This process will continue for this half. In the lower half the system stats from right to left to find the cutting point. The left pixels from the cutting point will remain same by removing others. In this way the upper portion of ট is detected and picked up with the matra zone and middle portion by removing the adjacent one. ঐ is segmented by keeping unchanged the right portion of the cutting point removing the left portion. This will continue to 1st half. The second half will remain identical. In this way the upper potion of ঐ is picked up with the matra zone and by removing adjacent characters of the middle portion. The result is shown below

$$\text{টী} = \text{ট} + \text{ঐ}$$

In the case of গী, কী, ণী, ঙী -- গী is isolated from by finding the gaps. There are 2 gaps in the immediate upper row of the matra upper boundary for গী where the others hold one.

c) *Segmentation of গী* - the thickness of the upper portion of ঐ is calculated. From start row, the system scans horizontally to search for black pixels. When the first one is found then the thickness is added with that black pixel's column position called thicknessColumn. Now the system goes left to right and keep the pixels same until it reaches to thicknessColumn making the rest of the pixels 0. In this way the upper potion of ঐ is picked up with the

matra zone and by removing adjacent characters of the middle portion. For ঐ, the system starts from left to right and if the gap is less than or equal to 1, the pixels will remain unchanged. Else the pixels will be zero to thicknessColumn. Then the rest of the pixels will remain unchanged. In this way the upper potion of ঐ is picked up with the matra zone and by removing adjacent characters of the middle portion. The result is shown below.

$$\text{গী} = \text{ঐ} + \text{ঐ}$$

d) *Segmentation of কী, ঙী* - This segmentation process is similar to the segmentation process of মি. The only difference is the process starts from the right side. The result is shown below.

$$\text{কী} = \text{ক} + \text{ঐ}, \text{ঙী} = \text{ঔ} + \text{ঐ}$$

e) *টী* - the system starts from right to left and continue until the system get white pixel after getting the first black one i.e. cutting point making the rest of the left pixels 0. During this time if the cutting point less than the previous one the current point will be replaced by the previous one.

TABLE IV

SUMMARY OF ঐ, ঐ, ঐ SEGMENTATION

Complex	Papers					Our process
	1	2	4	5	7	
আমি		Y				Y
ঢিয়া		Y				Y
জীব		Y				Y
টাকা		Y				Y
জেঠী		Y				Y
কোটা						Y
টা						Y
গী						Y

In this way the upper portion of ট is detected and picked up with the matra zone and by removing adjacent characters of the middle portion. ঐ is picked up by going right to left to the cutting point removing the rest of the left pixels. In this way the upper potion of ঐ is picked up with the matra zone and by removing adjacent characters of the middle portion. The result is shown below.

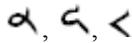
$$\text{টী} = \text{ট} + \text{ঐ}$$

Table IV shows the result for different papers as well as ours.

Words With Lower Modifier ঐ, ঐ, <

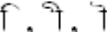

As they occur under the middle zone, they can be segmented properly after the successful detection of base

line [1], [2], [4], [7]. If the base line is not equal to end row of the text then the system can understand that there are lower modifiers. Then the system segments them by finding white pixels after finding first black pixels. Table V shows the use of this technique in different papers as well as ours.

TABLE V
SUMMARY OF  SEGMENTATION

Complex	Papers						
	1	2	3	4	5	7	Our process
সুখ	Y	Y	Y	Y		Y	Y
কুল	Y	Y	Y	Y		Y	Y
কুড়া							Y
গুন	Y	Y	Y	Y		Y	Y
রুই/রুই							Y
তুন	Y	Y	Y	Y		Y	Y

IV. CONCLUSION

This paper presents an approach to detect Baseline and a new algorithm to segment  in an intact shape i.e. by not separating the upper portions of them like . This algorithm is font size independent and matches best with Sutonny, karnaphuli, Rinkiy fonts etc. We didn't consider any noise in the textual image. We are working to develop segmentation algorithm to separate the overlapping, touching and compound characters.

REFERENCES

- [1] B.B.Chaudhury and U.Pal, "A Complete Printed Bangla OCR System", *Pattern Recognition*, vol. 31, pp. 531-549, 1997
- [2] Md. Abdus Sattar, Khaled Mahmud, Humayun Arafat and A F M Noor Uz Zaman, "Segmenting Bangla Text For Optical Recognition", *ICCIT 2007*, pp. 27-29 December, Dhaka, Bangladesh.
- [3] J U Mahmud , M F Rahman and C M Rahman, "A Complete OCR System for Continuous Bengali Characters" *Proc. IEEE TENCON*, 2003, pp. 1372-1376.
- [4] Md. Al Mehedi Hasan, Md. Abdul Alim, Md. Wahedul Islam and M Ganger Ali, "Bangla Text Extraction and Recognition From Textual Image", *Proceedings of 2nd national Conference on Computer Processing of Bangla*, pp. 171-176, February 2005, Dhaka, Bangladesh.
- [5] Md. Murad Hossain, Md. Shamsul Alam "BD Fine Reader: A Bangla Character Recognition system for Converting Printed Bangla Text Documents into Editable Electronic Document", pp. 192-199, *ICCPB 2006*, Dhaka, Bangladesh.
- [6] Veena Bansal and R M K Sinha, "Segmentation of Touching and Fused Devnagari Characters", www.iitk.ac.in/ime/veena/PAPERS/s.pdf
- [7] S M Millky Mahmud, Nazib Shahrier A S M Delowar Hossain, Md Tareque Mohmud Chowdhury and Abdus Sattar, "An Efficient Segmentation Scheme For The Recognition Of Printed Bangla Characters", *ICCIT 2004*, pp. 26-28, December 2004, Dhaka, Bangladesh.
- [8] U Garain and B B Chaudhury, "Segmentation Of Touching Characters in Printed Devnagari and Bangla Scripts Using Fuzzy Multifactorial Analysis", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 32, pp. 449-459, Nov.2002.
- [9] Apurba Lal Saha "Performance Evaluation of Neural Network Based Character Recognition System on Different Bengali Fonts", Masters Thesis Paper, IST, 2007, Dhaka, Bangladesh.