Segmentation of Touching Characters in Devanagari

Veena Bansal and R. M. K. Sinha
veena@iitk.ernet.in     rmk@iitk.ernet.in
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur 208016 India

Abstract- This paper presents an algorithm for segmentation of touching Devanagari characters (also referred to as conjuncts) into its constituent symbols and characters. Proposed algorithm extensively uses structural properties of the script. Statistical information about the height and width of character boxes, which are vertically separate from their neighbours, is used to hypothesize character boxes to be touching character boxes. The recognition rate of 85% has been achieved on the segmented touching characters.

1 Introduction

Many algorithms have been proposed for segmentation of touching characters [4, 5, 6, 7, 8] for Roman script.

In this paper, we have considered the problem of conjunct segmentation in the context of Devanagari script¹. Consider the word राष्ट्रीय which is pronounced as ‘rashtriya’. The preliminary segmentation phase based on projections (vertical and horizontal) yields the following units:

र । र । य । and the top modifier।

This process does not separate lower modifiers and leaves the conjuncts unsegmented. Some additional touching characters used in Devanagari are shown in figure 1. Statistical analysis of height and width information of the character boxes indicate that ॥ is a possible composite character and has a possible lower modifier. The output of the classification process is also taken into consideration before attempting further segmentation of the possible composite character. In this case, the image box ॥ ॥ is rejected by the classification process [9]. Therefore, image box ॥ ॥ is further segmented to obtain the following units:

॥ ॥ and the lower symbol।.

In section 2, the algorithm for segmentation of touching characters is presented. The algorithm for lower modifier separation is discussed elsewhere [1]. In section 3, we present results of our investigation followed by conclusion in section 4.

¹Devanagari is the script for Hindi which is official language of India. It is also the script for Sanskrit, Marathi and Nepali languages. It is used by more than 300 million people on the globe.

Touching characters

Constituent characters

<table>
<thead>
<tr>
<th>र</th>
<th>य</th>
<th>न</th>
<th>ब</th>
</tr>
</thead>
<tbody>
<tr>
<td>नम</td>
<td>त</td>
<td>न</td>
<td>ल</td>
</tr>
<tr>
<td>लं</td>
<td>न</td>
<td>ल</td>
<td>न</td>
</tr>
<tr>
<td>लक्ष्य</td>
<td>न</td>
<td>ब</td>
<td>र</td>
</tr>
<tr>
<td>र</td>
<td>य</td>
<td>न</td>
<td>ब</td>
</tr>
<tr>
<td>र</td>
<td>य</td>
<td>न</td>
<td>ब</td>
</tr>
</tbody>
</table>

Figure 1: Some of the Touching characters used in Devanagari script.

2. Segmentation of Touching Characters

We refer to the conjunct image by image(conj_left, conj_right, conj_top, conj_bottom) where conj_left, conj_right, conj_top and conj_bottom are the left, right, top and bottom coordinates of the minimum sized upright bounding box of the image. Before discussing the algorithms, we need to define the follow-
Vertical Projection:
The pixel projection has been defined by [6] as \( \{PXP(k), k = 1, 2, \ldots, W\} \). It consists of total number of black pixels in each vertical column and \( W \) is the width of the image. We refer to this projection as vertical projection.

Collapsed Horizontal Projection:
The collapsed horizontal projection is defined as \( \{HP(k), k = 1, 2, \ldots, H\} \) where \( HP(k) \) records the presence of black pixels in each horizontal row and \( H \) is the height of the image.

Continuity of the Collapsed Horizontal Projection:
We scan collapsed horizontal projection \( HP(k) \) in the sequence \( HP(1), HP(2), \ldots, HP(W) \) and record the position of first row containing a black pixel; let this position be \( i \). We continue the scan and record the position of first row containing no black pixels; let this position be \( j \). We continue the scan and the collapsed horizontal projection \( \{HP(k), k = 1, 2, \ldots, W\} \) has no discontinuity if all the rows below \( j \) contain no black pixels or \( j \) is the boundary row. The difference \( j - i + 1 \) is referred to as height of the projection.

Vertical Bar and its Location:
Devanagari characters can be divided into three groups based on the presence and position of a vertical bar, namely: **end bar characters**: अ ख च ज़ ढ़ ण़ त़ थ़ ध़, **middle bar characters**: ए क़ फ़ and **non-bar characters**: इ उ क़ छ ट ठ ड ढ ऱ.
The position of the vertical bar is the left most column where number of black pixels is 80% or more of the character height. This is determined by scanning the vertical projection of the image from left to right. The first column where target number of pixels are present becomes the position of the vertical bar.

Pen Width:
In Devanagari characters, pen width is same as the thickness of the header line. Thickness of the header line is obtained during the preliminary segmentation process from the horizontal histograms of the words.

The conjunct segmentation algorithm process takes the image of the conjunct and the co-ordinates of the enclosing box. The position of the vertical bar and pen width are also inputs to the algorithm.

We calculate height and width of the conjunct image referred to as \( \text{conj.height} \) and \( \text{conj.width} \) respectively. We also calculate left one third and mid columns of the conjunct image referred to as \( \text{left.onethird} \) and \( \text{conj.mid} \) respectively. For an example, refer to figure 2(a). Next, we establish the class of the first constituent character. Derived half form of the consonants are divided into two classes based on their heights:

**Height of the derived half form is less than 80% of the height of the base form:** Characters of this class are: ट़ ठ़ ड़ ढ़ ठ़ ठ़, ड़ ढ़ ठ़ ड़ ढ़ ठ़. This class is referred to as class \( H_1 \).

**Height of the derived half form remains the same as of the base form:** Some such examples are: ऱ द़ द़ द़ द़. This class of characters is referred to as class \( H_2 \).

The class of the first constituent character is determined by making collapsed horizontal projection of left 1/3rd of the conjunct image which is stored in \( \text{HP1by3} \). If the height of the \( \text{HP1by3} \) is more than 0.8 * height of the conjunct image, the first constituent character of the conjunct belongs to class \( H_2 \). Otherwise, it belongs to class \( H_1 \). Please refer to figure 2(b).

For class \( H_1 \) characters, we include additional columns in steps of one in image under consideration and modify \( \text{HP1by3} \). The inclusion of additional columns stops when the middle column of the con-
junct is reached or the pixels strength in the next column is more than the pixel strength in the column under examination.

In case of class $H_2$ characters, we look at the left one third of the image and locate the right most column which contains 50% or more black pixels of conjunct height. Starting from this column to the middle column of the conjunct image, we examine each column. We stop when the pixel strength in the next column is more than the pixel strength in the column under examination or the middle column is reached. The present column is the right boundary column of the first constituent character. Please refer to figure 2(c).

For extracting the second constituent character of the conjunct, the continuity of the collapsed horizontal projection is checked. The collapsed horizontal projection corresponding to a Devanagari character image has continuity. In case of end bar and middle bar characters, the collapsed horizontal projection corresponding the character image to the left of the bar also has continuity. However, if we vertically chop $\delta$ columns from the image from the left, where $\delta$ is a positive integer greater than or equal to the penwidth, continuity of the corresponding collapsed horizontal projection is lost.

The search for the left boundary of the second constituent character starts from $\text{conj.right}$ if no bar is present and immediately to the left of the bar if it is present. This point is referred to as $\text{temp Conj.right}$. We make an initial guess for the left column of the second constituent character. We refer to this column to as $\text{temp.left2}$ and set it to the $\text{temp Conj.right} - 2 * \text{penwidth}$. We now make a collapsed horizontal projection of the image enclosed by $\text{temp Conj.right}$, $\text{temp.left2}$, $\text{conj.top}$, $\text{conj.bottom}$. This is referred to as $\text{HP}r$. If $\text{HP}r$ has no discontinuity and the height of $\text{HP}r$ is more than $1/3rd$ of $\text{conj.height}$, $\text{temp.left2}$ becomes the left boundary of the second constituent character (referred to as $\text{left2}$). Otherwise, we move $\text{temp.left2}$ to further left in steps of one column and modify $\text{HP}r$. We stop moving $\text{temp.left2}$ further if $\text{HP}r$ has no discontinuity and required number of rows are present. The present value of $\text{temp.left2}$ becomes the left boundary of the second constituent character. However, if $\text{temp.left2}$ has reached $\text{left.onethird}$ and still a break column has not been located, the search is abandoned and no segmentation point is suggested. The magnified conjunct images and their segmentation have been depicted in figure 2. Please refer to figure 2(e) and (f).

If $\text{left2}$ is less than $\text{right1}$, both the segmentation points are ignored and no segmentation point is suggested. If $\text{right1}$ and $\text{left2}$ are same or the difference $(\text{right1} - \text{left2})$ is same or less than double of pen width, the segmentation is accepted. However, the $\text{right1}$ is moved to the left by the pen width. Please refer to figure 2(g).

3 Results

The structural segmentation algorithm has been tested on 18 document pages from two different magazines. The number of conjuncts in the test documents are about 5%. Out of all the conjunct and composite characters, about 90% conjuncts has been marked for further segmentation based on the output of the classification process (see [9, 10, 3] and width and height information of characters. During testing, it was observed that sometimes a composite character was substituted by another Devanagari character. The recognition of the constituent characters obtained after segmentation of touching character is about 88% which matches with the overall performance of the system [3]. The results of touching character segmentation are summarized in tables 1. However, the algorithm is capable of segmenting all
Figure 2: A Conjunct and its Segmentation
<table>
<thead>
<tr>
<th>Font</th>
<th>total number of chars.</th>
<th>overall recognition</th>
<th>number of touching chars.</th>
<th>touching char. recognition</th>
<th>touching chars. substitution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Font I</td>
<td>20856</td>
<td>18162</td>
<td>1172</td>
<td>981</td>
<td>191</td>
</tr>
<tr>
<td></td>
<td></td>
<td>87.08%</td>
<td>5.61%</td>
<td>83.70%</td>
<td>16.29%</td>
</tr>
<tr>
<td>Font II</td>
<td>16356</td>
<td>14359</td>
<td>802</td>
<td>678</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td></td>
<td>87.79%</td>
<td>4.90%</td>
<td>84.53%</td>
<td>15.46%</td>
</tr>
</tbody>
</table>

Table 1: Performance of the system for Font I and II.

the conjuncts when suggested to do so.

A sample text page and the text after recognition is shown in figure 3. The segmented image after preliminary segmentation and lower modifier segmentation is also shown in the same figure. The image after conjunct segmentation and the OCR output after post-processing [9] are also presented in the same figure.

4 Conclusion We have introduced a new method for the segmentation of the conjuncts for Devanagari script. This strategy is based on the structural properties of the script. The right and left part of the images are extracted independently.

Acknowledgments A part of this work was supported by TDIL sponsored project.

References


(a) The Input Text

भारत में गरीबी ग्रामीण क्षेत्र में ही विद्यमान है ऐसी धारणा चित्रन के स्तर पर मानी जाती रही है। इसीलिए गरीबी उपमूलन के अधिकांश कार्यक्रम ग्रामोन्मुख रहे हैं। गरीबी की विकार्तलता तथा विशालता निष्कंद है ग्रामीण क्षेत्र में विराट रूप से परिलक्षित होती है, किंतु शहरी क्षेत्र इससे अछूता हो, ऐसा नहीं है। शहरी क्षेत्रों में

(b) The image after preliminary segmentation and lower modifier separation

(c) The image after conjunct segmentation

(d) The output of the classification process; the word has been underlined if the true word is the second or third choice

भारत में गरीबी ग्रामीण क्षेत्र में ही विद्यमान है ऐसी धारणा चित्रन के स्तर पर मानी जाती रही है। इसीलिए गरीबी उपमूलन के अधिकांश कार्यक्रम ग्रामोन्मुख रहे हैं। गरीबी की विकार्तलता तथा विशालता निष्कंद है ग्रामीण क्षेत्र में विराट रूप से परिलक्षित होती है, किंतु शहरी क्षेत्र इससे अछूता हो, ऐसा नहीं है। शहरी क्षेत्रों में

Figure 3: