Segmentation of Touching Conjunct Consonants in Telugu using Minimum Area Bounding Boxes

J. Bharathi, P. Chandrasekar Reddy

Abstract— This paper addresses the problem of segmenting touching characters which are written or printed in the bottom zone. In the segmentation of machine printed Telugu document image, conjunct consonants are more prone to touching due to shape of the characters. It is important to segment them properly to improve the accuracy of the Telugu OCR as otherwise the reconstruction and mapping to editable electronic document is incomplete and often needs lot of tedious manual intervention. It is based on the script level characteristic that the secondary form of consonants are written in smaller size and its bounding box is smaller compared to the primary character. The structural feature of sharp peaks in both left and right side profiles at the touching location of the combined character is used for determining the correct segmentation location. The algorithm is tested on a dataset created from large set of documents. The success rate of 96.39% is achieved.

Index Terms— Minimum area bounding box, segmentation, side profile peaks, touching conjunct consonants.

I. INTRODUCTION

Telugu language is syllabic in nature. There are eighteen vowels, thirty-six consonants and three dual symbols, each represents a complete syllable. Telugu script has a vital inclination towards circular forms. All the letters and their modifiers can be derived by a combination of parts of circles. The script has basic symbols, modifier symbols (vowel modifiers, conjunct consonants) and script level grammar rules.

Conjunct consonants are consonant-consonant combinations. The consonants have secondary form known as 'Vattulu'. A consonant is combined with a secondary form of consonant to form a conjunct consonant. In Telugu script secondary form of consonants are written next or below the core character. Based on the zone in which they are written, these can be categorized into two types. The 'Type-1' are written in bottom and middle zones; and the 'Type-2' are written only in bottom zone and in smaller size. The 'Type-1' may touch with the primary character at the junction of bottom zone or at middle zone. The 'Type-2' may touch with the primary character at the junction of bottom and middle zone. The consonant (strictly speaking a half-consonant) is modified by the vowel modifier [Fig.1].

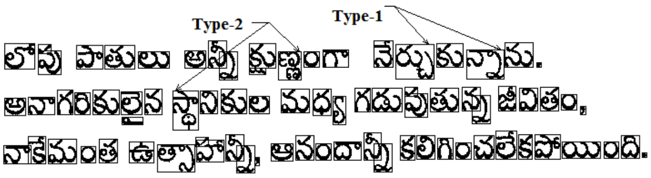

Type-2
Type-1

Fig.1 Touching conjunct consonants – Type-1 and Type-2

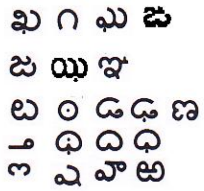


Fig.2 Secondary form of consonants (Type-2) that are written in bottom zone

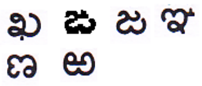


Fig.3 Secondary form of consonants which resemble the primary form

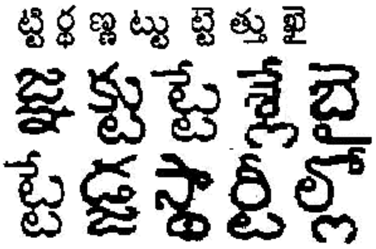


Fig.4 Some of the bottom zone touching conjunct consonants.

The secondary form of consonants of Type-2 that are written in bottom zone as shown in Fig.2 are prone to touching at the junction of middle and bottom zones. Few secondary forms (six) resemble the primary consonants [Fig.3][1].

Each character width varies considerably with the use of vowel modifiers and the character itself. Also most of the characters occupy the two zones viz., middle, top-middle zones. Parts of very few characters extend into bottom zone (eg. pu, sha, bha etc.). Due to the touching, the aspect ratio (defined as ratio of width to height) still gets reduced and this can be used to narrow down the search domain for identifying the Type-2 conjunct consonants.

It is observed that the horizontal profile of the combined touching character shows a valley at the location of the touching. As there are many other valleys present in the profile, it is difficult to identify the correct location. A better property is required for segmentation.

Manuscript Received July, 2013.

J. Bharathi, Department of Electronics and Communication Engineering, Deccan College of Engineering and Technology, Hyderabad, India.

Dr. P. Chandrasekhar Reddy, Department of Electronics and Communication Engineering, JNTU College of Engineering, Hyderabad, India.

II. LITERATURE SURVEY

The touching character segmentation is considered by many researchers earlier. Richard G. Casey and Eric Licolinet [2] described three strategies for segmentation. They are classical approach, in which segments are identified based on "character-like" properties, recognition based segmentation, in which the system searches the image for components that match classes for its alphabets and holistic method, in which system seeks to recognize words as a whole.

Liang *et al.* [3] proposed a dynamic recursive segmentation algorithm for words in Roman script. A discrimination function based on pixels and projection profiles is developed to find the break locations. Contextual information and spell check are used to correct errors caused by incorrect segmentation and recognition. Combining heuristic and holistic methods Min-Chul Jung and others [4] have proposed a recognition based segmentation algorithm for machine printed character strings of arbitrary length. Far left and far right profiles will not effected due to touching. Based on this, right profile of prototypes is matched. The touching word is segmented with the width of one of matching candidates and other three profiles are matched to identify the touching characters. The process is repeated until all characters are identified in the word. Kahan *et al.* [5] have defined an objective function as the ratio of second difference of the vertical projection profile function at a pixel to next pixel. The maximum of this objective function was used to find the possible break points.

Utpal Garain and Bidyut Choudhari [6] proposed a Technique for identification and segmentation of touching characters in printed Devanagari and Bangla scripts using fuzzy multi factorial analysis. Aspect ratio and measure of dissimilarity are used for identification of touching characters. A predictive algorithm is developed for effectively selecting probable cut columns to segment the touching characters. Jindal M. K., Sharma, R. K. and Lehal, G. S. [7] proposed to segment the touching characters in the top zone of printed Gurumukhi script using top profile projections based on the concavity and convexity of the characters. Devessar *et al.* [8] proposed a two pass algorithm for segmentation of machine printed touching characters in Gurmukhi script. Initially segmentation point is approximated and then the cutting point is optimized. This algorithm can be used to segment two or three touching characters. It can be extended to scripts having headlines.

Utpal Garain and Bidyut Choudhari [9] proposed an algorithm for segmentation of touching characters in mathematical expressions on multi factorial analysis. It evaluates four different factors defined in four directions of vertical, horizontal, +45^0 and -45^0. These are combined to obtain a single value 'f' for finding appropriate cut column with highest 'f' in each direction. Dong-Yu Zhang et al. [10] presented an improved method for segmentation of touching symbols in printed mathematical expressions by initially extracting the contour of the symbol image using contour tracing algorithm,, Next the concave corner points are detected and these points are considered as segmentation points.

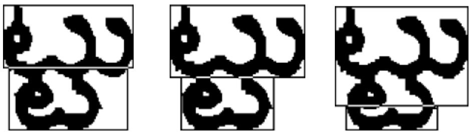
Less amount of literature is available for segmentation of touching characters in Telugu. L.P. Reddy *et al.* [11] proposed an algorithm for segmentation of touching characters based on topological properties for Telugu script and by splitting the vertical projection profile.

III. METHODOLOGY

A. Bounding box

Consider bounding boxes around the characters in Fig.5. The touching characters have bounding boxes enclosing both the characters. If the combined character is segmented properly, as the secondary form of consonant in bottom zone (Vattu) is relatively small compared to the first character, correspondingly its bounding box is also smaller than the bounding box enclosing the primary character.

It is observed that the width of the characters in Telugu script is more at the center of the middle zone because of the circular nature. So the combined character is segmented horizontally at mid depth. In the above figures [Fig.5a] the character is segmented at mid height and the bounding boxes are fitted for the top and bottom characters separately. Then gradually the line of segmentation is lowered. When the segmentation line is at the junction of primary consonant and the smaller secondary consonant, the bounding box of the lower part gets smaller as the character is small.



(a) (b) (c)

Fig.5 Bounding boxes for the top and bottom parts of the proposed segmentation line

Three parameters viz., the total area of bounding boxes A, the total of perimeters of the bounding boxes P and density of the pixels D defined as the number of pixels per unit area are studied for different locations of the segmentation line.

$A = A_1 + A_2$

where A$_1$ and$_2$ are the individual area of each bounding box

$P = P_1 + P_2$

where P$_1$ and P$_2$ are the perimeters of each bounding box

$$total\_blackpixels = \sum_{1}^{NXM} I_{inv}$$

$$pixel\_density = total\_blackpixles /(A_1 + A_2)$$

where I_{inv} is the inverted binary image

The total area A reaches the lowest value when the segmentation line is at the junction of middle and bottom zones. After still lowering the segmentation line, the area A1 increases and the area A2 decreases. However the increase in the area A1 is more compared to the decrease in the area A2. So the total area A in the Fig5b is the lowest. The graph in Fig.6 shows total area A versus the height from the top of the character in terms of pixels.

The perimeter also lowers and reaches a minimum value and remains constant thereafter [Fig.7]. This is because after it reaches the lowest value, increase of one pixel height of the top box increases the perimeter of top box by two and decreases the perimeter of bottom box by two pixels as the widths of the respective boxes remains same.

The density of the pixels D reaches maximum value when the boxes are at their lowest sizes as the area A is inversely proportional to density [Fig.8].

We can see that at the segmentation proposed at line corresponding to the lowest value of A or lowest value of P or the highest pixel density D effectively separates the touching character. Any of these parameters can be used to segment the character as all the parameters indicate a change in their value at the segmentation location. However for characters where the difference in the relative size is not much, the location of the proposed segmentation line is not accurate [Fig.9] because binarization may lead to fusing of the two characters with additional black pixels in between the characters.

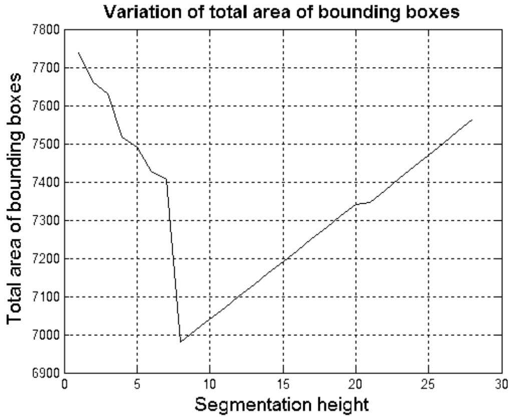


Fig.6 Variation of the total area of the bounding boxes

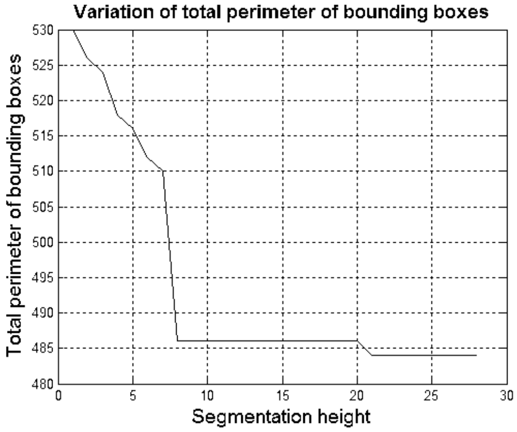


Fig.7 Variation of the total perimeter

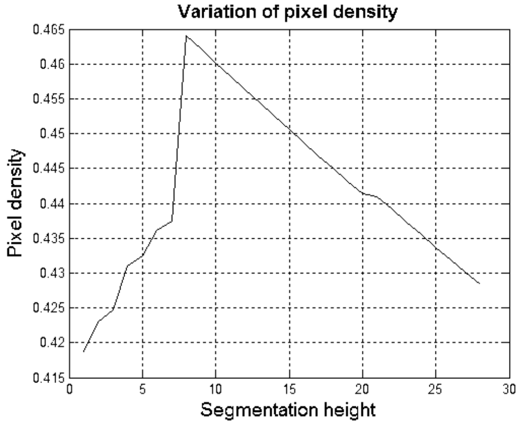


Fig.8 Variation of the density of pixels

B. Side profile peaks

We need another characteristic to accurately locate the segmentation line. It is to be noted that side profiles have few more peaks at other places. This feature in the side profiles may lead to false segment locations. This should be combined with the minimum area of bounding boxes concept described above, to identify the correct segmentation location. The sharp peak in the side profiles i.e., the white pixel count on either side of the character correctly segments the touching characters [Fig10]. Combining both the above phenomena clearly locates the segmentation line.

C. Identification

It is interesting to observe that for touching characters other than the Type-2 touching conjunct consonants, the above two conditions fail. This is used to effectively identify them. For the Type-1 touching conjunct consonants which extend into the middle zone the point of touching can be either at bottom or middle zone or both. For these characters the sum of the areas of the two bounding boxes will have lowest value (a steep fall followed by a steady rise), however the side profiles i.e., the white pixel count on either side will not have sharp peaks at the junction of the lowest areas. This feature can segregate touching conjunct consonants into two groups viz., Type-1 and Type-2. The segmentation of touching conjunct consonants of Type-1 was addressed in [12].

D. Procedure

All these rejected characters by the recognition module of the OCR are to be considered as the candidates for segmentation. A rejected or unidentified character has more distance than the given threshold value from the prototype database character [13].

Initially the segmentation line is considered at mid height of the character. A bounding box is fitted to the resulting top and bottom segments of the combined character. The areas of the top and bottom bounding boxes are calculated. In an iterative loop the combined character is segmented at increased height of top box, the sum of the areas and perimeters of the individual top and bottom bounding boxes are calculated. The index at the location of the minimum area is the probable location of segmentation. The search for the correct location is limited from mid height to a specified threshold value (0.8 times the height of combined character is considered here) beyond which it is unlikely to find the segmentation location or the combined area may have minimum value but with shallow fall.

The segmentation location calculated as above is further tested for the additional characteristic that the left side profile and right side profile has a peak [Fig.10].

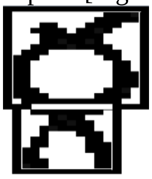


Fig.9 Bounding boxes with less area difference

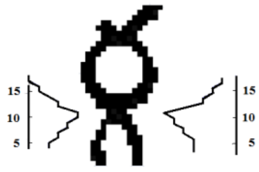


Fig.10 Peaks in the side profiles

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

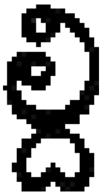


Fig.11 Touching character before segmentation

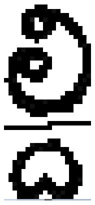


Fig.12 Touching character after segmentation

The probable segmentation location this aspect is fine tuned by calculating of the side profiles of left and right sides. A few scan lines at the top and bottom of the proposed segmentation line are considered and their peak positions on either side of the character are found.

If they fall on the same scan line a uniform horizontal segmentation line is proposed otherwise half of the touching width is segmented into the top character and the other into the bottom character [Fig.11 and Fig.12], where touching width is the horizontal width of the character at touching location.

E. Algorithm

```
1.  Read the binarized image
        $I \xleftarrow{read} Image\ file$
2.  Compute total pixel count in the image
        $pix = count(I)$
3.  Initialize segmentation location to half of line height
        $sl = 0.5 * h$
4.  Calculate the bounding box for the top part of the image
        $[w_1 h_1] \leftarrow bounding\_box(I_1)$
5.  Calculate the area of the top bounding box
        $a_{1=}\ 2 * (w_1 + h_1)$
6.  Calculate the bounding box for the bottom part of the image
        $[w_2 h_2] \leftarrow bounding\_box(I_2)$
7.  Calculate the area of the bottom bounding box
        $a_2 = 2 * (w_2 + h_2)$
8.  Compute total areas, perimeters and density of pixels of two bounding boxes
        $a = (a_1 + a_2)$
        $den = pix/a$
9.  Repeat the steps 4 to 7 incrementing sl by one pixel up to sl = 0.8*h
10. Find $sl_{opt}$ at which total area is minimum or density is maximum
        $sl_{opt} = argmin(a)$
        or
        $sl_{opt} = argmax(den)$
11. Calculate the count of white pixels of top and bottom n scan lines of $sl_{opt}$ on left side
        $count\_l_i = (cl_i|i = 1:n)$
12. Find the index cl_i of maximum count of white pixels
        $cl\_i = argmin(count)$
13. Calculate the count of white pixels of top and bottom n scan lines of $sl_{opt}$ on right side
        $count\_r_j = (cl_j|j = 1:n)$
14. Find the index cr_i of maximum count of white pixels
        $cr\_i = argmin(count\_r_j)$
15. If cl_i = cr_i
        segment at cl_i
    Else segment left half of touching width at cl_i and right
        half of touching width at cr_i
    where
    $touching\_width = width\ of\ image$
        $- count_{cl\ i} - count_{cr\ i}$
```

IV. RESULTS

Documents printed in Anupama, Hemalatha , Priyanka and Goutami fonts having sizes 10, 12, 14 points are collected.

TABLE I. MAXIMUM AND MINIMUM VALUES OF PARAMETERS

	Area		Perimeter		Density	
	Max	Min	Max	Min	Max	Min
	5244	4784	412	392	0.456	0.416
	6862	6104	480	444	0.438	0.390
	6380	4954	452	406	0.420	0.326
	11187	9467	650	564	0.461	0.390
	7232	6488	482	460	0.447	0.401
	7344	6733	488	462	0.392	0.360
	5916	4849	446	414	0.485	0.397
	7176	6301	496	446	0.406	0.356
	7524	6866	492	468	0.402	0.366
	9492	8442	562	502	0.383	0.340

We have also collected documents of children's books and the scanned and binarized documents from Digital Library of India (DLI). Each document other than the documents from DLI are scanned at 300 dpi, binarized, segmented for lines words and characters using horizontal and vertical profiles respectively and further the characters are subjected to connected component analysis to segment into glyphs which are separated by spaces and which cannot be segmented by vertical profiles. The maximum and minimum values of the total area, total perimeter and the density of the pixels at shown in Table I for different Type-2 touching characters.

TABLE II. Results

Total documents	221
Total characters	211,232
Total touching characters	4,164
Conjunct consonants(Type-1)	1,907
Conjunct consonants in bottom zone (Type-2)	526
% of conjunct consonants (Type-1)	45.80%
% of conjunct consonants in bottom zone (Type-2)	12.63%
Correctly segmented	507
% of success	96.39%

Our algorithm is used for identifying the bottom touching characters and also to segment them properly. The overall success rate of 96.39% is achieved on the data set consisting of all fonts, font sizes including the documents from DLI [Table II].

V. CONCLUSION

The data set consisted of all touching characters. It is shown that the script level properties combined with the structural properties can be used to successfully identify and segment the touching characters of Type-2 which are touching at the junction of bottom and middle zone. A success rate of 96.39% is achieved with the proposed algorithm.

REFERENCES

1. Edward Hill, C. “A Primer of Telugu Characters,” Manohar Publications, New Delhi. Can be viewed online at Digital South Asia Library (DSAL), University of Chicago, 1991.
2. Richard Casey, G., Eric Licolinet, “A Survey of Methods and Strategies in Character Segmentation”, *IEEE Trans. In Pattern Analysis and Machine Intelligence*, Vol. 18, No. 7, July 1996, pp. 690–706.
3. Su Liang, Shridhar, M, Ahmadi, M. “Segmentation of Touching Characters in Printed Document Recognition”, Pattern Recognition, Vol. 27, No. 6, 1994, pp. 825–840.
4. Min-Chil Jung, Yong-Chul Shin, Srihari, S. N., “Machine Printed Character Segmentation Method using Side Profiles”, IBM Journal of Research and Development, Vol. 26, No. 6, 1999, pp. 647–656.
5. S. Kahan, T. Pavlidis, and H. S. Baird, “On the recognition of printed characters of any font and size”, *IEEE Transactions on PAMI*, Vol. 9, No. 2, March 1987, pp. 274-288.
6. Utpal Garain and Bidyut B. Chaudhuri, “Segmentation of touching characters in printed Devnagari and Bangla scripts using fuzzy multifactorial analysis”, *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, Vol. 32, No. 4, November 2002, pp 449-459.
7. Jindal, M. K., Sharma, R. K., Lehal, G. S.,. “Segmentation of Touching Characters in Upper Zone in Printed Gurmukhi Script”, Compute ’09, Proc. Of 2nd Bangalore Annual Compute Conference, Article 9, Jan 9-10, 2009, Bangalore.
8. Neena Madan Davessar, Sunil Madan, Hardeep Singh, "A Hybrid Approach to Character Segmentation of Gurmukhi Script Characters," aipr, pp.169, 32nd Applied Imagery Pattern Recognition Workshop AIPR 2003, 2003, pp 169-173.
9. Utpal Garain and B. B. Chaudhuri, “Segmentation of Touching Symbols for OCR of Printed Mathematical Expressions: An Approach based on Multifactorial Analysis”, Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition (ICDAR‟05), IEEE, 2005, pp. 177-181.
10. Dong-Yu Zhang, Xue-Dong Tian, Xin-fu Li, “An Improved method for segmentation of touching symbols in printed mathematical expressions”, *International Conference on Advanced Computer Control, ICACC*, Vol 2, March, 2010, pp 251-253.
11. Pratap Reddy, L., Ranga Babu, T., Venkata Rao, N., Raveendra Babu, B., 2010. “Touching Syllable Segmentation using Split Profile Algorithm”, IJCSI, Vol. 7, Issue 3, No. 9, Nov 2010, pp. 17–26.
12. Bharathi. J, Chandrasekhar Reddy. P, “Segmentation of Telugu touching conjunct consonant using overlapping bounding boxes” in *International Journal on Computer Science and Engineering (IJCSE)*, Vol. 5, No. 06, Jun 2013, pp 538-546.
13. Pavan Kumar. P., Chakravarthi Bhagavathi, Atul Negi, Arun Agarwal, Deekshatulu. B. L. “Towards improving the accuracy of Telugu OCR system”, *International Conference on Document Analysis and Recongnition, ICDAR*, 2011, pp 910-914.

AUTHORS PROFILE

J. Bharathi received her B.Tech degree in Electronics and Communication Engineering from Acharya Nagarjuna Unineiversity, Guntur, India. She received her M.Tech degree in Digital Systems and Computer Electronics from Jawaharlal Nehru Technological University, Hyderabad, India. She joined as faculty member in Electronics and Communication Engineering Department, Deccan College of Engineering and Technology, Hyderabad, India and is currently working as Associate Professor. Her research interests include Image Processing, Speech and Signal Processing.

Dr. P. Chandrasekhar Reddy received his B.Tech. degree in Electronics and Communication Engineering from Jawaharlal Nehru Technological University, Hyderabad, India and M.E. from Bharatiyar University, Coimbatore. He received his M.Tech and Ph.D from Jawaharlal Nehru Technological University, Hyderabad, India. He joined as faculty in JNTU, Anantapur. Currently he is working as Professor Co-ordination in JNTU, Hyderabad, India. He is an author of numerous technical papers in the Fields of High Speed Networking and Wireless Networks. His research interests include Mobile and Wireless Communications and Networks, Personal Communications Service and High Speed Communications and Protocols.