Segmentation of Touching Hand written Telugu Characters by using Drop Fall Algorithm

Srinivasa Rao A V
Department of ECE,
AKRGCET, Nallajerla,
AP, India.

D R Sandeep,
Department of ECE,
SVEC, Pedatadepalli,
Tadepalligudem, AP,
India

V B Sandeep
Department of ECE,
VLITS, Vadlamudi,
AP, India.

S Dhanam Jaya
Department of CSE,
AKRGCET, Nallajerla,
AP, India.

Abstract— Recognition of Indian language scripts is a challenging problem. Work for the development of complete OCR systems for Indian language scripts is still in infancy. Complete OCR systems have recently been developed for Devanagri and Bangla scripts. Research in the field of recognition of Telugu script faces major problems mainly related to the touching and overlapping of characters. Segmentation of touching Telugu characters is a difficult task for recognizing individual characters. In this paper, the proposed algorithm is for the segmentation of touching Hand written Telugu characters. The proposed method using Drop-fall algorithm is based on the moving of a marble on either side of the touching characters for selection of the point from where the cutting of the fused components should take place. This method improvers the segmentation accuracy higher than the existing one.

Index Terms— Segmentation, Telugu Alphabets, Drop Fall algorithm,

INTRODUCTION

India is a multilingual country with a large number of written scripts. OCR development is yet to take a commercial shape for many of these scripts. Character segmentation is the first step of OCR system that seeks to decompose a document image into a sequence of sub images of individual character symbols. Segmentation methods are described in[1,2]. The "classical" approach consists of methods that partition the input image into sub- images, which are then classified. The operation of attempting to decompose the image into classifiable units is called "dissection." The second class of methods avoids dissection, and segments the image either explicitly, by classification of pre-specified windows, or implicitly by classification of subsets of spatial features collected from the image as a whole. The third strategy is a hybrid of the first two, employing dissection together with recombination rules to define potential segments, but using classification to select from the range of admissible segmentation possibilities offered by these sub-images. Finally, holistic approaches that avoid segmentation by recognizing entire character strings as units. Segmentation of handprint or kerned machine print demands for a two-dimensional analysis. Even non-touching characters may not be separable along a single straight line. A common approach is based on determining connected black regions (”connected components,” or” blobs”). Further processing may be necessary to combine or split these components into character images. There are two types of follow-up processing. One is based on the “bounding box,” i.e., the location and dimensions of each connected component. The other is based on detailed analysis of the images of the connected components.

A segmentation technique for touching of type printed Thai characters is proposed[3] by Sarin Watcharabutsarakham, which uses structural characteristics to detect suitable segmentation points in both horizontal and vertical directions. A new machine printed Arabic character segmentation algorithm is proposed [4] by Liying Zheng, based on vertical histogram with the rules are based on the structural characteristics between background regions as well as character components. The characteristics of isolate d Arabic characters are used to check whether the sub-word includes only one character. The vertical histogram associated with other rules used to find real segmentation points. A new technique is proposed[5] by Utpal Grain et al for identification and segmentation of touching characters based on fuzzy multi- fractural analysis. A predictive algorithm is developed for effective selection of possible cut columns during segmentation process. Recognition of text heavily depends on proper segmentation of text into lines, words and then individual characters or sub-characters for feature extraction and classification of these characters. An error in segmentation may lead to wrong recognition of text and the system may be rendered useless. A detailed survey on Indic script recognition is presented in [6]. In some of North Indian script alphabets like Bangla, Gurmukhi, etc., it is noted that many characters have a horizontal line at the upper part. It is called, head-line or sirorekha. A new approach to segmentation of machine printed Gurmukhi text is proposed [7] in literature using a two-pass mechanism. In pass-one it approximates the segmentation point, while in pass-two the cutting point is optimized. This approach is tested to be successful in segmenting a pair as well as triplets of touching characters. Chaudhuri et al [8] put forward a method for segmentation of handwritten Bangla text into characters. Based on certain characteristics of Bangla writing methods, different zones across the height of the word are detected. These zones provide certain structural information about the constituent characters of the respective word. In Bangla handwritten texts often there is overlap between rectangular hulls of successive characters. So a method of recursive contour falling in one of the zones across the height of the word is proposed to find out the extents within which the main portion of the character lies. An attempt is made to segment handwritten Devanagari words [9] by R.J.Ramteke. The segmentation algorithm is motivated by the structure of characters. To take care of variability involved in the writing style of different individuals, a novel set of features are proposed for clustering the end-bar and no-bar characters. The categorization is done by the centre calculated by fuzzy C-means theorem followed by the features drawn by invariant moment. The Gaussian distribution function has been adopted for classification. The structural properties of South Indian scripts differ completely from North Indian scripts. Unlike Devangari,formation of words from individual characters does not possess a uniform link. The characters exist as individual components, which is a common phenomena in all South Indian scripts. A font and size independent OCR system for printed Kannada documents is reported [10] recently by Ashwin and Sastry. The system first extracts words from the document image and then segments into sub-character level pieces. The segmentation algorithm is motivated by the structural features of the script.

Nagi et al [11] presented an approach to Telugu OCR which limits the number of templates to be recognized to just 370, avoiding issues of classifier design for thousands of shapes or complex glyph segmentation. A compositional approach using connected components and fringe distance template matching is tested to give a raw OCR accuracy of about 92%. In the present work we propose a drop-fall method for segmenting printed words in Telugu script. Segmentation of printed words in Telugu script is a challenging task because of the large number of shape variations of Telugu character set.

Proposed drop fall algorithms attempt to build a segmentation path by mimicking an object falling or rolling in between the two characters which make up a connected component. This class of algorithm was first proposed by Congedo et al. in [12]. There are four primary types of drop-fall algorithms which differ on the direction and the starting point of the drop fall. These are top-left (or left-descending), top-right (or right-descending), bottom-left (or left ascending), and bottom-right (or right ascending).

I. TELUGU SCRIPT

India is a multilingual country. Out of 18 officially recognized languages in India, 9 languages have separate scripts and the other languages are written either in Perso-Arabic script or Devanagari script..Telugu is the official language of the state of Andhra Pradesh in southeastern India where it is spoken by close to 120 million people. Telegu is a richly developed language and the biggest linguistic unit in India. The script consists of vowels, consonants, consonant-vowel core formation and a large number of conjunct formations. For all these formations there exist nasal sounds represented with the help of 'anuswara' sign's an addition. Vowels are 16 independent letters represented with individual glyph. Consonants are 35 individual letters with distinct glyph set. The dependent vowel signs also called 'matras', play an important role in the formation of the glyph. Logically the character glyph formations for these combinations are 455.The shape of consonant-vowel formations and conjuncts is dependent on the context and is affected by the order of consonants and vowels. Here Indic scripts provide different types of glyph orders for different languages, though the canonical structure is common.

Methodology

Drop fall algorithms are based on the principle that a fairly optimal 'cut' between two connected characters can be made if one were to role a hypothetical marble off the top of the first character and make the cut where the marble falls. Despite its apparent simplicity, the algorithm has proven itself to be quite useful.

The drop-fall algorithm is based on the principle of letting a hypo-theatrical marble fall in between two connected characters and making the cut where the marble lands. Based on this simple description of the method, the main issue which needs to be addressed in its implementation is where to drop the marble from. This is important since if the algorithm starts in the wrong place, the 'marble' could easily roll down the left side of the first digit or the right side of the second digit and, thus, would be completely ineffective. There are several methods available to decide where to start the drop-falling process from. Obviously, it is best to start as close as possible to the point at which the two characters are connected. Dimauro et al [1] outline a method which does this quite robustly. In this method, the pixels are scanned row-by-row until a black boundary pixel with another black boundary pixel to the right of it is detected, where the two pixels are separated by only white space. This pixel is then used as the point from which to start the drop fall. By scanning row-by-row, left to right, this is the first pixel which would meet the criterion of being a border pixel separated from another pixel to the right only by white space.

A more naive choice for the initial pixel would be the first pixel found by scanning row-by-row which has white space to the right of it. This method fails, however, in when the first such pixel encountered is part of the second of the two connected characters. In this case, the algorithm will 'fall off' the right side of the characters After the initial pixel is found, the next step is to begin the actual drop fall. The drop-falling algorithm is designed to mimic falling, so it will always move downwards, diagonally down-wards, to the right, or two the left. The directions that the algorithm will 'move' is according to the current pixel position and its surroundings.

Variations of Drop-falling:

The standard version of the drop-fall algorithm described above falls down and to the right from the top left of the connected component. There are three other variations of the algorithm accept for the fact that they don't necessarily initiate from the top left or fall 'down'.

Top-Right Drop Fall: This algorithm is identical to the standard drop-fall except that it initiates from the top-right of the connected component rather than the top left. Also, instead of falling down to the right, it falls down and to the left. The standard drop fall algorithm can be used if the input image is 'flipped' vertical. The resulting segmentation path P can be obtained through the transformation of equation (1) where $Pinv_x$ is a vector of the x coordinates (or column indices) of the segmentation path resulting from the standard drop fall algorithm being performed on the vertically inverted image; $Pinv_y$ is the vector of the y coordinates (or row indices) of the segmentation path resulting from the standard drop fall algorithm being performed on the vertically inverted images; and w is the width of the image. The segmentation process carried out on touching Telugu words like THADU by using this method is shown in Fig 1. Along with English pronunciation

$P_x^i = w – Pinv_x^i$

for i=1, 2, ..., n where n is the length of Pinv -------(1)

$P_y^i = Pinv_y^i$

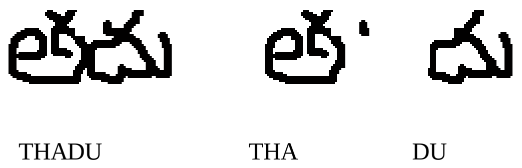

Fig.1 Segmentation of Touching Telugu alphabets by using Top-Right drop fall

Bottom-Left Drop Fall: This algorithm is identical in principle to the original drop-fall algorithm accept that it initiates from a pixel at the bottom left of the image and 'falls' up and to the right. This algorithm can be implemented by performing a standard drop-fall on an input image which is flipped along its horizontal axis . The bottom-left drop fall path can then be obtained using equation (2) where h is the height of the image. The segmentation process carried out on touching Telugu words like AADHAA and TARAA by using this method is shown in Fig 2. Along with English pronunciation

$P_x^i = Pinv_x^i$

for i=1, 2, ..., n where n is the length of Pinv --------(2)

$P_y^i = h - Pinv_y^i$

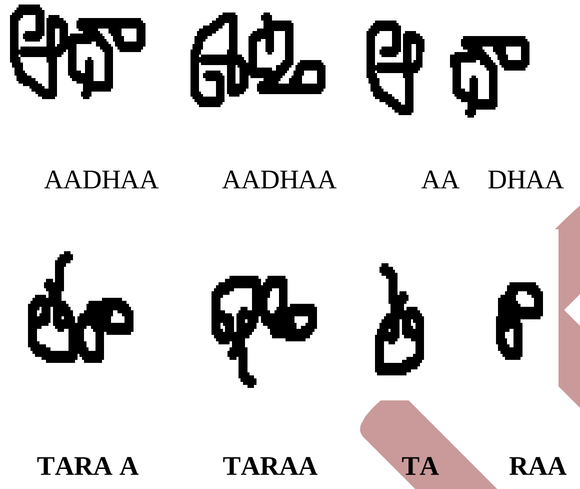

Fig.2. Segmentation of Touching Telugu Alphabets by using Bottom-Left drop fall from Left to Right and Top to Bottom

Bottom-Right Drop Fall: This splitting heuristic is identical to the previous three except that in this case, the falling commences from the bottom right of the image and goes in an up and leftward direction. It can be viewed as the exact opposite of the standard top-left drop fall. As before, a bottom-right drop fall can be implemented by performing a standard top-left drop fall after appropriately transforming the input image. This time, the image must be flipped in both the horizontal and vertical directions. The bottom-right path P can be obtained from the transformation of Pinv given in equation (3). The segmentation process carried out on a touching Telugu words like LAALU and YEKKU by using this method is shown in Fig 3. Along with English pronunciation

$P_x^i = w - Pinv_x^i$

for i=1, 2... n where n is the length of Pinv -- --------(3)

$P_y^i = h - Pinv_y^i$

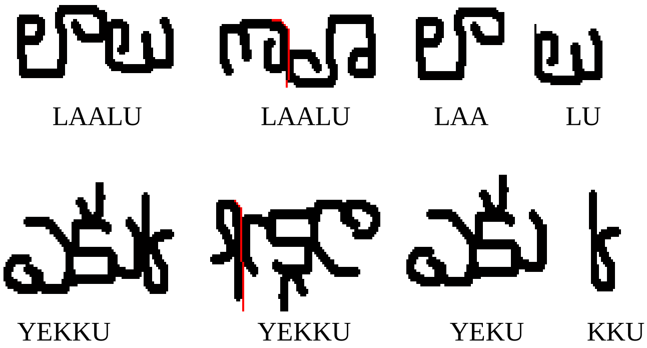

Fig.3. Segmentation of Touching Telugu Alphabets by using Bottom-Right drop fall from Left to Right and Top to Bottom

Results

The four cases of proposed algorithm tested on various touching words of Telugu samples created through the paint brush based on the analysis of 3000 samples collected from daily news papers. For any of the drop-fall heuristics, it is easy to find cases where they work well and where they don't work well. It is however, much more difficult to find examples when all of them do not work well. As mentioned earlier, the drop falls which start from the bottom of the image (bottom-left and bottom-right) tends to succeed in most of the cases where the top-based drop falls fail some of examples are given in Fig.5. Not only Top Left, remaining three cases also fails for some words due to the cursive nature of the script. Based on the fact that the top-based and bottom-based drop falls 'complement' each other

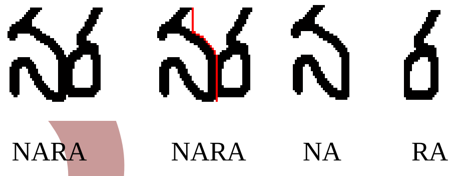

Fig 4. Examples for which the Top-Left drop fall method succeeds

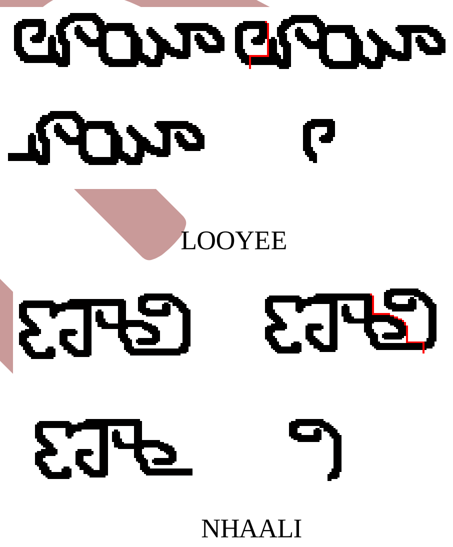

Fig 5. Examples for which the Top-Left drop fall method fails

CONCLUSIONS

The algorithms described here are proposed for improving the performance of segmentation systems which has the broader impact of improving the quality of automatic handwriting recognition. The proposed algorithm gives better results in segmentation of touching hand written (developed by using software) Telugu characters. Especially this drop fall is more useful for cutting the connected components on various places of touching characters. The same algorithm will be tested on machine printed and hand written touching Telugu characters as future extension.

References

[1] Casy, R.G. and Lecolinet. E., "A Survey of Methods and Strategies in Character Segmentation" IEEE Trasactions on Patterns Analysis and Machine Intellegence, 1996, vol.18, no.8, pp.690-706.

[2] Liang, S., Sridhar. M. and Ahmadi., "Segmentation of Touching Characters in Printed Document Recognition", Pattern Recognition, 1994, vol.27, no.6, pp.825-840.

[3] SarinWatcharabutsarakham.,"SegmentationforTouching Typewrittens", TENCON 2004. 2004 IEEE Region 10 Conference Volume A, Issue, 21-24, Vol. 1, Nov. 2004 Page(s): 199 - 202 .

[4] Liying Zheng, Abbas H. Hassin a, Xianglong Tang, " A new algorithm for machine printed Arabic character segmentation" Pattern Recognition Letters vol.25, (2004), pp.1723–1729.

[5] G.condego, G.Dimauro, S.Impedovo and G.Pirlo" Segmentation of Nuemaric Strings" Proc. Of Third Int. Conf. on Document Analysis and Recognition, Montreal, Aug 14-16, 1995, pp. 1038-1041.

[6] Utpal. Garain and Bidyut B. Chudhury., "Segmentation of Touching Characters in Printed Devanagari and Bangla Scripts Using Fuzzy Multifactorial Analysis", IEEE Transactions on Systems, Man and

[7] U.Pal and B.B.Chaudhari, "Indian Script Character recognition: A Survey", Pattern Recognition, 37(2004), pp.1887-1899.

[8] Neena Madan Davessar, Sunil Madan, Hardeep Singh," A Hybrid Approach to Character Segmentation of Gurmukhi Script Characters" Proceedings of the 32nd Applied Imagery Pattern Recognition Workshop (AIPR'03), IEEE, 2003.

[9] A. Bishnu and B. B. Chaudhuri," Segmentation of Bangla Handwritten Text into Characters by Recursive Contour Following" . Proceedings of the Fifth International Conference on Document Analysis and Recognition, 1999. ICDAR apos;99 Volume , Issue , 20-22 Sep 1999 Page(s):402 – 405

[10] R. J. Ramteke, S. C. Mehrotra,"Segmentation and Clusteing of Handwritten Devanagari Text" IEEE International Conference on Signal and Image Processing, ICSIP-06,7-9 Dec 2006, Hubli

[11] T.V.Ashwin, P.S.Sastry, "A font and size independent OCR system for printed Kannada documents using support vector machines",2002, Sadhana, 27, pp.35-58.

[12] Atul Negi, Chakravarthy Bhagvati, B.Krishna, "An OCR for Telugu", in proceedings of the sixth International Conference on Document processing,2001, pp.1110-1114.

First Author Adabala Venkata Srinivasa Rao obtained his B.Tech degree in Electronics and Communication Engineering from JNT University, Kakinada,AP, India, AMIE Electrical from Institute of Engineers (India), Kolkotta,India. and M.Tech in Instrumentation and Control Systems from JNT University, Kakinada, AP, India. He was worked in various engineering colleges at different positions. He is currently working as an Associate Professor in AKRG college of engineering and technology, Nallagerla, WG Dist, Andhra pradesh, India. He has 9 years of teaching and 4 years of industrial experience. He has 13 publications in various International Conferences and journals. His area of interests include Pattern Recognition , Image Processing and VLSI. He is an active member in professional bodies like AMIE, IACSIT

Second Author D Rajesh Sandeep obtained his B.Tech degree in Electronics and Communication Engineering from Sir CRR college of Engineering,Eluru, AP,India. and M.Tech in DECS from Gudlavalleru Engineering college, Gudlavalleru, AP,India. He is currently working as an Assistant Professor in Sri Vasavi Engineeering college, Tadepalligudem, Andhra pradesh, India. He has 6 years of teaching experience. He has 6 publications in various International Conferences and journals. His area of interests include Pattern Recognition , Image Processing and VLSI.

Third Author Velivela B Sundheep obtained his B.Tech degree in Electronics and Communication Engineering from G.H.Raisoni College Of Engineering, Dig Doh hills, Nagpur India, and M.Tech in Embedded System and Technology, SRM University, Potheri, Chennai, India. He was worked in various engineering colleges. He is currently working as an Assistant Professor in Vijnan's lara Institute of Technology and Science, Vadlamudi, Andhra pradesh, India. He has 4 years of teaching experience. He has 3 publications in various International Conferences and journals. His area of interests include Pattern Recognition , Image Processing and Embedded Systems.

Fourth Author S.Dhanamjaya receivied his Bachelor degree in 2001 and Master degrees (CSE) in 2003 from NIEIT,Newdelhi . He completed M.Tech in CSE from JNTUK in 2010, He is presently pursuing his doctorate. He has a total experience of 9 years. He is working as an Associate Professor in AKRG College of Engineering & Technology ,Nallajerla, Andhrapradesh . His areas of interest include Image Processing, Network Security and Cryptography, Advanced Computer Architecture, DataBase Management System, Computer Organization, Computer Networks.