International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

A Decision Tree Based Font Style/Size Independent Kannada Printed Character Recognition System

N. Shobha Rani[1], Smitha Madhukar[2]

[1]Lecturer, Department of Computer Science, Amrita Vishwa Vidyapeetham,
Mysore Campus, Mysore-570026, Karnataka, India

[2]Assistant professor, Department of Computer Science, Amrita Vishwa Vidyapeetham,
Mysore Campus, Mysore-570026, Karnataka, India

Abstract: ***Segmentation is one of the most critical as well as important stage in optical character recognition system. Especially the segmentation of South Indian scripts has become one of the challenging aspects in order to provide a standard solution to South Indian OCR's. The segmentation of Kannada and Telugu scripts are considered to be still more serious researches due to the highest number of characters and increased variability, touching characters and overlapping characters in its native characters. This paper aims at providing an efficient touching line segmentation and classification algorithm in application with multiple projection profiles, bounding box analysis, Pearson's correlation features and decision tree classifier. The algorithm has provided improved accuracy in recognizing the complex or overlapping characters and proved to be efficient by obtaining around 97% - 99% of accuracy.***

Keywords: Touching line segmentation, Decision tree classification, Pearson's correlation features, Character Recognition.

1. Introduction

Kannada is considered to be the official language of state Karnataka and spoken by more than 48 million people in South India. Even though the popularity of English language has spread across multiple states, regions, the usage of regional languages seems to be high in various part of the state Karnataka. Especially the government offices/educational institutions as lots of application need to be fulfilled in its regional language Kannada. Therefore the document automation/storage is the most primary necessity to retain the historical documents and preserve the statistics of various activities happened. The document automation may save lots of time that has been spent in data entry. Data entry may in turn introduce human errors and various other inconsistencies in the data which create wrong illusions and ambiguities about data under analysis/usage. All these critics can be handled effectively through automated system like optical character recognition system.

Optical character Recognition is the process of automating the digitized documents into editable format documents. Optical character recognition has become heart of various fields where data is of primary importance for their a smooth functioning of business needs. OCR technologies has spanned across various fields in the recent times. Some of its applications include [1] reading aid for the blind, automatic text entry into the computer for desktop publication, library cataloging, ledgering, automatic reading for sorting of postal mail, bank cheques and other documents, document data compression: from document image to ASCII format, language processing such as indexing, spell checking, grammar checking, multi-media system design etc.

Any character recognition system undergoes pre-processing, segmentation, feature extraction, classification and recognition. However segmentation, classification and recognition are considered to be most typical and important which leads towards the overall accuracy of the character recognition and judges the efficiency of the system. Especially segmentation is one of the most crucial stages and considered to be heart of any South Indian OCR system. The segmentation of South Indian languages like Kannada is very complex due to its highest number of alphabet set, connected characters, touching and overlapping characters [2]. The usual segmentation techniques like projection profiles are very much suitable for languages like English and not efficient to deal with South Indian scripts like Kannada and Telugu. Thus we intend to devise an efficient and simple algorithm that can segment Kannada printed characters containing various artifacts like touching/ overlapping/ compound characters in application with multiple projection profiles, bounding box analysis and decision tree classifier.

2. Complexity of Kannada Script

Kannada alphabets are evolved from Kadamba and Chalukya scripts which are considered as descendents of Brahmi script. The basic structure of Kannada script varies from many other Roman scripts. Kannada characters doesn't contain shirorekha and it as a very wide alphabet set. Kannada alphabet set includes 51 base characters, 16 vowels and 35 consonants. Besides these alphabets it as consonant and vowel modifiers and all these possible combinations may create 560 different combinations. The presence of consonant and vowel modifiers along with base consonants and vowels may generate a compound/connected characters/ overlapping characters. These are sources that introduce complexity in segmentation and classification of Kannada script [3]. The largest dataset of Kannada script may result in more number of classes that may even introduce performance issues of the system resulting into more time complexity and misclassification of data. All these various factors motivated us to devise an algorithm that can efficiently deal with this wide variety of characteristic

complexity of Kannada script.

3. Literature Survey

Most of the experimentations are devised for the Kannada character recognition system in the literature using various approaches approaches. Details of some of the experimentations performed and its loopholes are presented in this section.Anupama et.al. [1], Had propped a method segmentation of handwritten Telugu characters using multiple projection histograms. The method initiates by segmenting lines using horizontal projection histograms, then for word/ character segmentation vertical projection histograms are used. The method gives good results only when the line is free from connected components, overlapping and touching components. M. Swamy Das et. Al [2], had devised a method to segment the overlapping characters in printed Telugu text documents using Projection profile, connected component approach and vertical spatial relationships. The experimentation results are satisfactory but the method is complex in terms of identification of spatial vertical relationships and moreover not able to segment overlapping characters when touching characters are present. Nallapareddy priyanka et. Al.[3] had proposed a technique for segmentation of lines and words in printed Telugu text document using projection profiles and run-length smearing. The approach is a good fit to do the line and word segmentation and method may not be a best fit segment characters and it requires only the clear white gap to separate two lines. Manish Kumar Jindal et. Al., [4] proposed algorithms to segment touching characters, and overlapping lines in degraded printed Gurmukhi document. Various categories of touching characters in different zones are considered and the structural properties of script are used. The algorithm proposed for segmenting horizontally overlapping lines uses a heuristics based upon the height of a character. Siddaling Uralogin et.al [5] had devised a method for Kannada printed character segmentation using zone based projection profiles and has been tested various types of fonts with accuracy of output around 92%. Sanjeev Kunte et. al [6], devised a two stage character segmentation technique for Kannada character segmentation in which the algorithm first segments the vatthus using connected component analysis and then segments the characters at the second step. J. Bharathi et. al [7], had implemented a approach for touching character segmentation using partial projection profiles and bounding boxes. The algorithm gives around 91% accuracy and recognizes touching location in case of overlapping characters. Amit Patel [8] had propped an approach for Telugu character recognition without an explicit segmentation using convolution neural network approach that constructs a graph transformer network for the data set and performs the recognition. Vijaya Kumar Koppula et. al., [9] had proposed an algorithm for Telugu line segmentation based on the concept of fringe maps where the fringe number associated with each pixel gives the distance of nearest black pixel through which segmentation path can be sought. Dhaval Salvi et. al [10], had devised a longest path algorithm for handwritten cursive English script segmentation. The longest path algorithm uses a graph model to predict the possible touching locations in the text. M. Blumenstein et. al, [11] had designed a method of identifying segmentation points in cursive text in application with Artificial Neural Networks in which the network is trained with global features of characters prior to the determination of segmentation points.

Even though much experimentation are devised to provide a efficient solution to the Kannada character recognition system, still there are some limitations that may obstruct the accuracy of the system, especially in the segmentation stage and classification stage of the system. Thus the paper focuses on devising an enhanced algorithm for segmentation and classification of the Kannada printed character recognition system.

4. Proposed Methodology

The proposed system of Kannada printed character recognition system is composed of several stages like preprocessing, segmentation, feature extraction, classification, recognition and post processing. We mainly concentrate on resolving the conflicts that may arise during segmentation and classification stages. The figure 1 depicts the architecture of the proposed system.

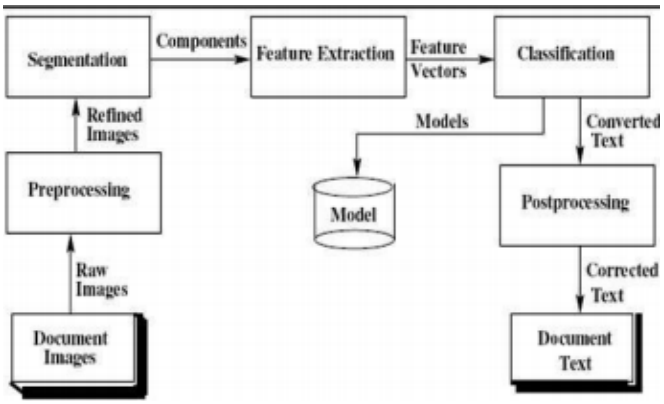

Figure 1: Architecture of Proposed System

4.1 Pre-Processing

Pre-Processing is considered to be preliminary stage of document processing. A well deskewed gray scale/color input document is binarized using quad tree based binarization approach [13] where the document undergoes iterative cycles for enhanced thresholdings that can eliminate scanning artifacts and noise in the document. Gaussian filter is used to improve quality of document image by making it suitable for further processing.

4.2 Segmentation

The pre-processed document is sent as input for segmentation stage. Segmentation is typical as concerned with Kannada script due to the presence of consonant conjuncts in each line. The consonant conjuncts may overlap with next line. The proposed system as employed an enhanced approach of using projection profiles. Assuming that not all the lines might be overlapping with one another the average height of a line is calculated using horizontal projection profiles. However in case of printed Kannada documents this assumption will be true as all printed

documents maintains constant height and clear segmentation path may exists for at least two consecutive lines in the document in worst case. Obtaining height of at least one line may help us to fix the average height of the line. Usually subscripts will occupy 1/10th portion of the line which will be in the bottom zone and considered as minimum threshold for line extraction. The proposed method uses the minimum boundary and maximum boundary of each column within the average height of the line considered as threshold and extracts that portion till all the columns in that line exhausts. The figure 2 shows the set of lines in a sample Kannada printed document image.

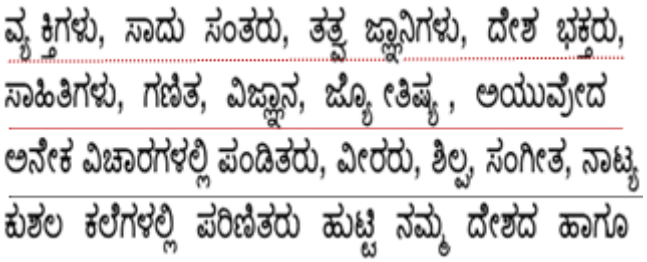

Figure 2: Sample Image with overlapping lines

Between line 1 and line 2 and line 2 and line 3 in figure 2 there is no clear segmentation boundary as indicated with red dotted line, therefore we cannot obtain the height of line. But between line 3 and line 4 there is a clear segmentation boundary as indicated with black dotted line in figure 2. This helps us to obtain the average height of the line. The average height of line is divided into two zones top zone and bottom zone. Starting with first zero valley index of the line each row in the preferred average height area (from zero valley to average line height) is analyzed vertically column by column till first non-zero pixel is encountered after crossing the first zone (i.e., $3/4^{th}$ of average line height) within the average line height.

The presence of non-zero pixel indicates the end of line and considered as maximum segmentation boundary for that corresponding column in that line. This process is repeated till all the columns corresponding to the line exhausts. Similarly beginning of next line is considered as ending of bottom zone of previous line and same process is repeated for extraction of all the lines till all rows in the image exhausts. The extracted lines are stored as separate image files and proceeded further for word segmentation.

Figure 3: Sample line with overlapping characters

Algorithm for Touching Line Extraction:
Input: Pre-Processed Image I
Output: Extracted Line

Step 1: Read the Input Image
I <= Pre-Processed Image
Step 2: Obtain Horizontal projection profile
Repeat for each row in I
Count the number of black Pixels in each row
End
Step 3: Analyze the horizontal projection profile for at least two consecutive zero valleys
Step 4: Avg Height = height of line obtained
Step 5: Starting with first zero valley index divide the image into equal number of chunks as of average line height.
Step 6: Store the starting index of each chunk in minimum boundary of each line array.
Step7: Divide each chunk into two zones as top zone and bottom zone
Midzone = Avg Height/2
Bottomzone-Start=Midzone + 1/10*(Avg Height)
Topzone= Avg Height – Bottom zone
Step 8:Inspect each chunk vertically from beginning of bottom zone till first non-zero pixel is encountered.
Step 9: Extract the line from Index of beginning of first zero valley index to non-zero pixel identified
Step 10: Store first non-zero pixel identified as maximum boundary array of each line.
Step 11: Using minimum and maximum boundary array extract each line and store it as separate image
Step 12: Repeat the process till all rows in the image I exhausts.
Step 13: Stop.

4.2.1 Word Segmentation

Each line image extracted from document is further traversed column by column to obtain vertical projection profiles. The spatial vertical relationships i.e., zero valleys from one word to another word is used for word extraction during connected component analysis.

4.2.2 Character Segmentation

Character segmentation is very complex compared to word segmentation and line segmentation, due to the presence of consonant and vowel modifiers. The connected characters may obstruct the spatial relationships from one character to another character. Unlike words in the document, uniform space is not maintained from one character to another character which is as represented in Figure 4.

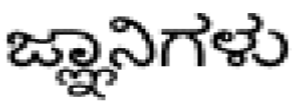

Figure 4: A word block without uniform spatial relationships between characters

Each word extracted is analyzed for connected components [6] by using connected component analysis algorithm and bounding boxes are applied to all connected components identified. Then each bounding box is considered by applying recursive XY-cut algorithm [10] and sent for further processing.

4.3 Feature Extraction:

Extracting features from an image is a fundamental procedure for classification of various characters. The Correlation features evaluate subsets of features on the basis

of the following hypothesis: "Good feature subsets contain features highly correlated with the classification, yet uncorrelated to each other". Pearson's correlation measure is used in the proposed system for feature extraction. A basic property of Pearson's r is that its possible range is from -1 to 1. A correlation of -1 means a perfect negative linear relationship, a correlation of 0 means no linear relationship, and a correlation of 1 means a perfect linear relationship. The following equation gives the merit of a feature subset S consisting of k features:

$$Merit_{S_k} = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}.$$

$\overline{r_{cf}}$ is the average value of all feature-classification correlations, and

$\overline{r_{ff}}$ is the average value of all feature-feature correlations.

In the proposed methodology, each segmented block of characters is normalized to a size of 40 x 40 and directed for feature extraction process. The normalized segmented block is subject to extraction of pixel wise correlation features. These features extracted from each segmented character are preceded to the decision tree classifier.

4.4 Classification

The classification technique is a systematic approach to build classification models from an input data set. Decision trees are powerful and popular tools for classification and prediction. Decision trees represent rules, which can be understood by humans and used in knowledge system such as database. In the decision tree, the root and internal nodes contain attribute test conditions to separate records that have different characteristics. The entire terminal nodes are assigned a class label of characters set in Kannada.

The proposed system has trained the decision tree classifier correlation features for data sample of around 6 different font styles for each character in Kannada including with vowels and consonants. The decision tree is constructed in learning phase. During the testing phase the correlation feature input is tested across each node to predict the appropriate class of the character.

4.5 Post-Processing

Once the appropriate class is predicted by the decision tree classifier, the corresponding class Unicode is printed in the editable document like notepad, MS-word.

5. Experimental Results and Discussion

We have collected data samples of around 6 different font styles for each character including with vowels, consonants, vowel modifiers, consonant modifiers and other connected component characters. In overall of 550 different classes of characters including with compound characters and 6 different font styles has generated 550 x 6 = 3300 classes of data as number of terminal nodes in decision tree classifier. Around 50 sample documents are tested and reached an overall accuracy of 97-100% in most of the cases. Except for characters like **Sree** or compound characters with more than two consonant modifiers, the system has given satisfactory results. The accuracy of the recognition is computed by number of characters recognized divided by total number of characters in the whole document.. The empirical relation for accuracy is given by,

$$\text{Accuracy} = \frac{\text{Number of characters recognized}}{\text{Total number of characters in the whole document}}$$

The experiment results are as presented in the figure 5 and figure 6.

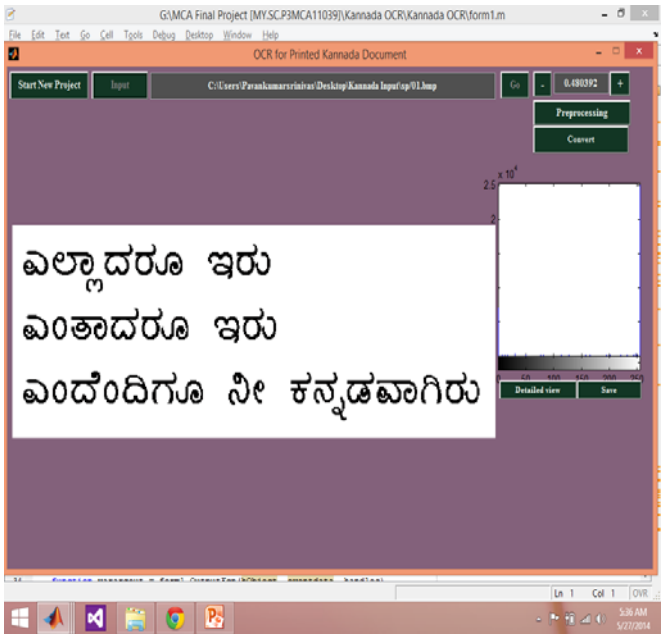

Figure 5: The Non-Editable input Image

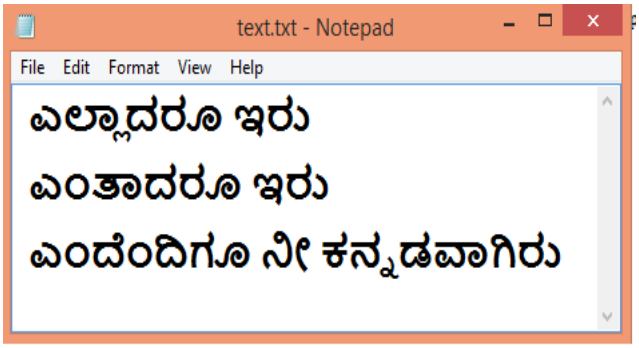

Figure 6: The Editable Output document

Table 1: The accuracy of document scanned as per font styles

Font Name	*Number of Document Samples*	*Number of characters recognized*	*Total number of characters*	*Accuracy attained*
Nudi Akshara-05	10	628	636	98.7%
Nudi Akshara-06	5	548	552	99.2%
Nudi Akshara-07	10	800	820	97.5%
Nudi Akshara-09	5	455	460	98.9%
Nudi Akshara-11	10	750	760	98.6%
Nudi Akshara-12	10	850	870	97.7%

6. Conclusions

This paper describes a simple and efficient OCR system for Kannada printed text documents. The GUI is very user friendly and accepts complex Kannada document as input image and converts it into machine editable format. The system is designed to be independent of the font and size of text. In future the system can be extended to work with various types of font styles. The touching segmentation of Kannada printed characters proves to be efficient and gives accurate character recognition in 99% of cases.

7. Future Scope

In recent times various technologies as evolved which can imitate many human behaviours and actions. Kannada printed character recognition is one pace towards it. The paper clearly as provided the scope for clear line and character segementation with simple strategies with respect to printed character recognition. Further it can be extended towards the handwritten Kannada character recognition and even approaches for broken character segmentation shall be devised.

References

[1] N. Anupama, Ch. Rupa, Prof. E. Sreenivasa Reddy, "Character Segmentation for Telugu Image Document using Multiple Histogram Projections", Global Journal of Computer Science and Technology Volume XIII Issue V Version I 2013.

[2] M Swamy Das, Dr. CRK Reddy, Dr. A Govardhan, G. Sai Krishna, " Segmentation overlapping text lines and characters in printed Telugu text document images", / International Journal of Engineering Science and Technology Vol. 2(11), 2010, 6606-6610.

[3] Nallapareddy Priyanka, Srikanta Pal, Ranju Mandal, "Line and Word Segmentation Approach for Printed Documents ",IJCA Special Issue on "Recent Trends in Image Processing and Pattern Recognition" RTIPPR, 2010.

[4] Manish Kumar Jindal, Gupreet Singh Lehal, Rajendra Kumar Sharma, " On segmentation of touching characters and overlapping lines in degraded printed Gurumukhi script", International Journal of Image and Graphics, Volume 09, Issue 03, July 2009.

[5] Siddaling Urolagin, K. V. Prema, N. V. Subba Reddy, " Document image segmentation for Kannada script using zone based projection projection profiles", AIM/CCPE 2012, pp. 137-142, 2013, Springer.

[6] R. Sanjeev Kunte, Sudhakar Samuel R.D, " A two stage character segmentation for printed Kannada text", GVIP Special Issue on Image Sampling and Segmentation, March, 2006.

[7] J. Bharathi, Dr. P. Chandra Sekhar Reddy, "Segmentation of Telugu Touching Conjunct Consonants Using Overlapping Bounding Boxes ",International Journal on Computer Science and Engineering (IJCSE), ISSN : 0975-3397 Vol. 5 No. 06 June 2013.

[8] Amit Patel, "A Hybrid Convolutional Neural Network for Telugu OCR without explicit character Segmentation", School of Computer and Information Science, University of Hyderabad.

[9] Vijaya Kumar Koppula, "Fringe Map Based Text Line Segmentation of Printed Telugu Document Images", International Conference on Document Analysis and Recognition (ICDAR), 2011.pp 1294 – 1298, ISSN : 1520-5363, E-ISBN : 978-0-7695-4520-2.

[10] Dhaval Salvi, Jun Zhou, Jarrell Waggoner, and Song Wang, "Handwritten Text Segmentation using Average Longest Path Algorithm", 978-1-4673-5052-5/12/$31.00 ©2012 IEEE.

[11] M. Blumenstein, B. Verma, "A New Segmentation Algorithm for Handwritten Word Recognition", School of Information Technology, Griffith University-Gold Coast Campus, PMB 50, Gold Coast Mail Centre, QLD 9726, Australia.

[12] K. Indira, S. Sethuselvi, " Kannada Character Recognition System a review", Inter-JRI Science and Technology, Vol. 1, Issue 2, July 2009.

[13] N. Shobha Rani, Arun Gopi, "A Quad Tree Based Binarization Approach to Improve quality of Degraded Document Images", International Journal of Computer Science Engineering (IJCSE), ISSN: 2319-7323, Vol. 3 No.01 Jan 2014.

[14] Gonzalez R C, Woods R E 1993, "Digital image processing", (Boston, MA, USA: Addison Wesley Longman Publishing Co. Inc.)

[15] C. Balletti F. Guerra, "Image matching for historical maps comparison", e-perimetron, Vol. 4, No. 3, 2009[180-186].