A Machine Editable Format for Kannada Handwritten Character Recognition Using A Dissect and Connect Technique

Aravinda. C. V\(^1\), Dr. H. N. Prakash\(^2\)

\(^1\)Asst. Professor Department of Information Science & Engineering, S.J.B.I.T BGS Health City, Kengeri, Bengaluru, Karnataka
\(^2\)Head Department of Computer Science & Engineering, Rajeev Institute of Technology, Hassan, Karnataka, India

Abstract—HCR for Indian Languages is a challenging task where there is relatively little work has been done. The typical nature of the kagunita, the compound characters in which one or more consonants combine with vowel, increases the complexity of the character for recognition. The segmentation is the major task of any OCR. The aim is to provide segmentation techniques that have been developed, 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". We try to reduce the character complexity by segmenting each character into four sub images namely main unit, right adjunct, bot- tom adjunct, top adjunct so that we can decrease the number of symbols for the classification. In this paper, we experiment with the use of moments features on Kannada kagunita. Moments and statistical features are extracted from the cut images. These subspace features are used for recognition on Neural Network Classifier. The result is obtained in the editable format which can be edited for the future purpose.

Keywords—Optical Character Recognition, Handwritten Character Recognition, Optical Handwritten Character Recognition.

I. INTRODUCTION

OCR of Indian Scripts is one of the challenging tasks in the field of Pattern Recognition. It is very valuable in terms of the variety of applications and also as an academically challenging problem. When OCR is used as a solution for inputting regional language data and also as a solution for converting paper information to soft form, OCR solutions become powerful component in addressing the digital divide. It also provides a solution for processing large volumes of data automatically. Hence extensive work is happening in this field on different scripts. Very little work is reported on Indian language scripts. The huge size of the character set, their complex shapes, similarities between different alphabets and the generally ambiguous nature of handwriting have made it difficult to attain good recognition accuracy. In spite of these difficulties, efforts are being made to interact with machines through handwriting.[2] In this direction more efforts are being made to realize OHCR systems of South Indian Languages like Telugu and Tamil than Kannada.

II. KANNADA LANGUAGE SCRIPT

Kannada, the official and administrative language of the South Indian state of Karnataka, is spoken by about 48 million people. The Kannada alphabets were developed from the Kadamba and Chalukya scripts. Descendants of Brahmī which were used between the 5th and 7th century A.D. The basic structure of Kannada script is distinctly different from Roman script. Unlike many North Indian Languages, Kannada characters do not have shirorekha (a line that connects all the characters of any word) and hence all the characters in a word are isolated.[6] This creates difficulty in word segmentation. Kannada script is more complicated than English due to the presence of compound characters. However, the concept of upper/lower case characters is absent in this script. Modern Kannada has 50 base characters, called as Varnamale. There are 13 vowels as in fig and two special characters called as Yogawahans as in fig. There are five letters, a total of twenty five letters called as grouped consonants or structured consonants as in fig.

We used Kannada script for our experiments. Kannada is the official language of Karnataka which is one of the South Indian state. The proposed system breaks down the recognition process into five fundamental sequential stages: pre-processing, segmentation, feature extraction, classification and conversion to editable format. Kagunita are curved in nature with some kind of symmetric structure observed in the shape.[4] The characters have very similar shape with minute difference among them. Hence, missing even smallest detail of the character can result in an incorrect recognition. In this paper, we address the complexity issues of Kannada Characters for recognition and also to obtain the result in editable format. The complexity of the character is decreased by segmenting each character into three sub images and extract the features of each sub images and then subject them for classification using SVM classifier. Initially the structural and the dynamic features of Handwritten Characters are exploited to segment the compound Kannada characters into 282 distinct symbols. This overcomes the huge data collection problem and also reduces the computational costs and complexity.
There are ten miscellaneous constants (unstructured constants) as in fig. Consonant take modified shapes when added with vowels. When consonant character is used alone it results in a dead consonant (mulavanyanja). Vowels modifiers can appear to the right, on the top or at the bottom of the base consonant. Fig 4 shows a consonant modified by all the 15 vowels. Such constant-vowel combinations are called live consonants (gunitakshara). When two or more constant conjuncts appear in the input they make a consonant conjuncts (vatthu) as in Fig(4). The first consonant take full form and the following consonant becomes half consonant. An example of a word with a vowel-modifier and a consonant conjunct (vatthu) is shown in Fig(6). In addition, a special character is used along with consonant conjunct or with a consonant-vowel combination and forms a word with arkavatthu. Example of a word with a consonant-vowel combination and a new character (arkavatthu) is shown in fig Therefore the total number of Kannada character combinations as mentioned in table 1 is 647921, even though some of them are not phonotactically valid. So, it is almost impractical to build a recognizer that deals with such a huge number of classes. Therefore it is feasible to break the character into their constituents and reorganize these constituents independently.

III. IMAGE PREPROCESSING TECHNIQUES

Image processing is referred to processing of a 2D picture by a computer. An image is defined in the "real world" is considered to be a function of two variables. Modern digital technology has made it possible to manipulate multi-dimensional signals with systems that range from simple digital circuits to advanced parallel computers. The goal of this manipulation can be divided into three categories:

1. Image processing (image in $\rightarrow$ image out)
2. Image analysis (image in $\rightarrow$ measurements out)
3. Image Understanding (image in $\rightarrow$ high level description out)

Before going to processing an image, it is converted into a digital form. Digitization includes sampling of image and quantization of sampled values. After converting the image into bit information, processing is performed. This processing technique may be Image enhancement, image restoration and image compression.
Image Enhancement: It refers to accentuation, or sharpening, of image features such as boundaries, or contrast to make a graphic display more useful for display & analysis.

Image Restoration: It is concerned with filtering the observed image to minimize the effect of degradations. Effectiveness of image restoration depends on the extent and accuracy of the knowledge of the degradation process as well as on filter design. Image Compression: It is concerned with minimizing the number of bits required to represent an image.

IV. IMAGE SEGMENTATION USING DISSECT AND CONNECT TECHNIQUE

Image Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. On observing the character combinations, we can notice four regions namely, middle, right, top and the bottom; there exists a vertical overlap among these four regions. In addition, the order of writing a character covering these regions is right, top, middle region vowel modifiers and then the bottom. Each region may have a single or multiple strokes. In some cases, the middle and the top regions are connected or the writing style of individuals may lead to this condition and both the regions are written using a single stroke. However, the bottom region is not connected with the middle region. In the proposed scheme, the well defined rules of combining different symbols to obtain compound characters, writing methods and the dynamic pen motion of online Kannada handwriting are exploited to segment a character into four groups. They are,

- The middle unit (MU), comprising of main character
  - The right adjunct (RA) having strokes written to the right of MU and occupying the middle region.
  - The bottom adjunct (BA) having the bottom region strokes such as conjuncts and vowel modifiers.
  - The top adjunct (TA) with different strokes.

Algorithm For Segmentation:

The epigrammatic steps to segment a given script is as follows

1. Line Segmentation: The lines of a text block are segmented by finding the valleys of the projection profile computed by a row wise sum of black pixel values.

2. Letter segmentation: Characters of a line are segmented by finding the valleys of the projection profile computed by column wise sum of black pixel values. After the letter is obtained it is dissected into four sub images There are very small variations of matra shapes with same size on the top. In bottom there are only two variations of same size. To make this information prominent for its recognition, this portion can be cut around 20% on the top and bottom respectively giving 2 different images. The right vowel matra varies in size. We can observe four different sizes of vowel matra. To recognize the right side vowel, we therefore considered 4 different size images one original image and 3 cut images so that at least one category can best represent the vowel matra. The first image category is the one with no cuts (original whole image), second category with around 20% cuts, third with around 40% cut and the fourth category with around 60% cut. Hence there will be around 6 images to help in the recognition of vowel in the Kagunita. Similarly, to recognize the consonant component in the Kagunita, as the images are size normalized, the vowel matra variation varies the size of the consonant present to the left of the vowel matra. Hence if no vowel matra to the right then consonant occupies the whole image. With 20% vowel matra the consonant occupies 80% of the total space and so on. Therefore to recognize the consonant we considered again 4 images with one original and 3 cut images as shown in the fig.
V. FEATURE EXTRACTION

In pattern recognition and in image processing, feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (e.g., the same measurement in both feet and meters) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction.

- Moment based features Think of each character as a PDF. The 2-D moments of the character are: From the moments we can compute features like: mpq
- Total mass (number of pixels in a binarized character)
- Centroid - Center of mass
- Elliptical parameters Eccentricity (ratio of major to minor axis) Orientation (angle of major axis)
- Skewness
- Higher order moments

Geometric Moments

For a digital image f(x,y) of size MxN, the geometric moment of order (p,q) is given by

\[ \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} x^p y^q f(x,y) \text{ with } p,q = 0,1,\ldots,\infty \]

All mpq with p+q <= n a positive integer, are the geometric moments of order p+q. For a binary image, the zero order moment m00 represents the total area of an image. The two first order moments, m01 and m10 represent center of mass of an image. In terms of moment values, the coordinates of center of mass also called centroid To make features translation invariant, the M x N image plane is to be mapped onto a square defined by x \( \in [-1,+1] \) and y \( \in [-1,+1] \) Invariance with respect to position of the object in the image can be achieved by calculating the central moments of the mapped digital image.

VI. CLASSIFICATION

The feature vector extracted from the segmented character is assigned a label using a classifier. Recognition of segmented characters is performed using NN Classifier.

The recognition performance of Back Propagation network will highly depend on the structure of the network and training algorithm. Feed forward back propagation neural network has been selected to train the network. The number of nodes in input, hidden and output layers will determine the network structure. All the neurons of one layer are fully interconnected with all neurons of its just preceding and just succeeding layers.

VII. CONVERSION TO EDITABLE FORMAT

The flow chart describing the process of conversion to editable format is given in fig.7.1. The group of the classified sub image is obtained. Based on this group value, the Unicode corresponding to the character is stored into a variable. This value is stored into a word array. On completion of the entire line, the content of the word array is written onto the file. On completion of recognition of the entire image, the file is saved and closed. It is then opened in read and write mode in order to perform the required editing. The file can be saved for the latter use.

VIII. RESULTS

The experiment was performed with 25 different handwritten scripts. About 30 datasets for each character were collected and was trained using NN classifier. The input image was preprocessed, segmented and applied to the classifier for recognition. Appreciable results were obtained for legible handwritten scripts with accuracy of 96%. Fair results were obtained with more vohaksharas with accuracy of 81% since segmentation was a difficult task.
IX. CONCLUSION AND FUTURE ENHANCEMENT

HCR is the process of identifying the handwritten characters. The text in an image is converted into letter codes which are usable within computer and text-processing applications. Here recognition is done for kagunita, the script of Kannada language. The proposed idea aims to reduce confusion between similar shaped characters in kagunita. It attempts to improve the overall accuracy of the HCR. It converts the image into an editable format. The editable text can be saved and opened for editing. The current system can be combined with other features to improve the performance further. An overall architecture for HCR incorporating all these features can be developed to attempt to improve the overall accuracy further. Such a structure will help to exploit further domain information in the recognition process. The current system is implemented only to recognize kagunita. This can be extended to recognize typical Vothaksharas also.

REFERENCES


AUTHOR’S PROFILE

Aravinda C.V Currently working as an Assistant Professor in Dept of IS&E at S.J.B.I.T, Kengeri Bengaluru, and Pursuing his Ph.D(Computer Science and Engineering) at V.T.U. Belgaum on “Character Recognition for South Indian Languages.”

H.N. Prakash received the B.E. degree in Electronics and Communication Engineering from the University of Mysore, Mysore, India, in 1990 and the M.Tech degree in Electronic Instrumentation from Regional Engineering College, Warangal, India in 1995 respectively. He received the Ph.D degree in Computer Science and Engineering from the University of Mysore, Mysore, India, in 2010. He is currently a Professor in the Department of Computer Science and Engineering at Rajeev Institute of Technology, Hassan, India. He is a life member of Indian professional bodies such as Institute of Engineers, Indian Society for Technical Education (ISTE) and System Society of India (SSI) and Member of Indian Institute of Engineers, India. He has authored several peer reviewed papers at national and international conferences and Journals Including IEEE Transactions. His research interest includes signature analysis and retrieval, clustering, biometrics, Image processing, Pattern recognition and symbolic data analysis.