Recognition of Printed Kannada Numerals based on Zoning Method

Ravindra S. Hegadi
Department of Computer Science
Soapur University
Solapur-413255, India

ABSTRACT

Zoning is one of the popular methods used for the optical character recognition of documents. In this paper the zoning approach is used for recognition of printed Kannada numerals. The input scanned document image containing printed Kannada numerals is binarized. The noise present in the document in the form of tiny dots is eliminated. The row segmentation followed by the column segmentation is performed on this document to segment out every numeral. The number of regions is obtained from this segmented numeral, which will be used as one of the feature during recognition stage. A morphological thinning algorithm is applied to thin this numeral. In the next stage the number of end points and the coordinate values of each end point are obtained. The zones in which the end points lie, and the regions that each numeral generates, are used for the recognition of the numeral. The proposed algorithm is applied on the document containing the printed Kannada numerals of different fonts generated using Nudi 4.0 software. The analysis of recognition using proposed method is also presented here.

General Terms

Document image analysis, digital image processing, Kannada numeral recognition.

Keywords

Zoning, printed Kannada numerals, handwritten Kannada numerals, OCR, numeral recognition.

1. INTRODUCTION

Kannada is one of the Indian languages belonging to Dravidian family of languages. The Kannada language is mainly spoken in Karnataka by about 60 million people and it has its own script [1][2]. Kannada is known to be the third oldest languages of the world. The India's highest literary honor, the Jnanapeeth awards, has been conferred eight times to Kannada writers, which is the highest number of times for any Indian language [3]. It has been awarded the classical language status by the Government of India [4]. The evolution of Kannada script is primarily from stone carving, because of which most of the characters are round, curvy and symmetric with straight strokes/wedges. This script is also used to write the Telugu language (script derived from Old Kannada or Halegannada), Tulu Language, Banada Language, Konkani by the Konkani diaspora in coastal Karnataka [5]. Kannada language has 10 digits, as in any other languages.

In the recent years few works related to development of optical character recognition for the Kannada character has been reported. Machine replication of human functions, like reading, is an ancient dream. However, over the last five decades, machine reading has grown from a dream to reality. Optical character recognition has become one of the most successful applications of technology in the field of pattern recognition and artificial intelligence. Many commercial systems for performing OCR exist for a variety of applications, although the machines are still not able to compete with human reading capabilities.

The problem of document image analysis can be subdivided into optical character recognition, script identification and graphical object recognition. Kannada numeral recognition problem is one of the sub problems of character recognition. The problem of character recognition can have further subdivision as recognition of printed and handwritten characters. The common image processing techniques used for the character recognition are line thinning, locating lines, curves and edges. These images may also contain the pictures. The processing of such pictures is not in the domain of document image analysis. The problem of text processing includes the processing of characters, numerals and special characters. One important task carried out on text is the optical character recognition (OCR) which is used to recognize the text which includes alphabets, numerals and special characters [6].

A template matching based approach for numeral recognition was proposed by R. S. Hegadi [7], in which the resized numeral is compared with the stored templates. Based on the correlation coefficient between the two numerals, recognition is carried out. This method has reported reasonable accuracy but fail to recognize the broken numerals. A neural network based classifier using wavelet transform coefficients as features for recognition was proposed by S. R. Kunte et. al. [8]. It could address the problems associated with the template matching approaches for character recognition, but, it rate of accuracy is still significantly low and requires large database for training the network. The work proposed by T. V. Ashwin et. al. [9] for printed Kannada characters works on template matching approach for recognition mechanism and uses SVM classifier. This methodology is highly sensitive to font changes, and pre-processing stage is not framed properly, due to which the problems are found in segmentation and resizing stages. In particular, untrained fonts cannot be accurately recognized.

Kannada character recognition based on k-means clustering is reported in [10]. The authors propose a segmentation technique to decompose each character into components from 3 base classes, thus reducing the magnitude of the problem. K-means provides a natural degree of font independence and this is used to reduce the size of the training database to about a tenth of those used in related work. The accuracy of proposed work is compared with the related works.

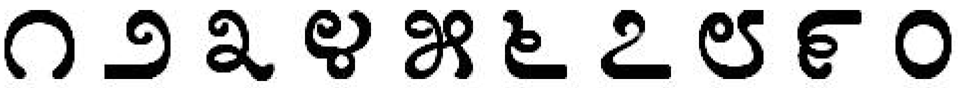


Fig 1: Printed Kannada Numerals

In neural network based classification of Kannada numerals has been proposed by R. S. Hegadi [11] in which the printed and handwritten numerals are pre-processed to eliminate the noise in the document. The edge detection based on Sobel operator is applied and then edge linking is performed to connect the broken edges. The region filling is used to fill the holes in the numerals. Then each numeral is segmented and resized to 5 × 7 pixels. This image is converted into one-dimensional array. A multi-layer feed forward neural network is used to classify the Kannada numerals. Four sets of characters are used for training the network and one set of numerals are used to testing. The classifier took more time for classifying the handwritten numerals as compared to the printed numerals.

This paper presents recognition and classification of printed Kannada characters based on zoning method. Section 2 describes the methodology where the image is pre-processed and features are extracted from each numeral image. The results of the proposed method are discussed in the section 3. Conclusions are discussed in section 4.

2. PROPOSED METHODOLOGY

The image document containing printed Kannada numerals will be converted to binary image. The noise present in the form of tiny dots is eliminated. In this document the pixel corresponding to numerals will be white in a dark background. The row segmentation followed by the column segmentation is performed to obtain each numeral. The regions generated by each numeral are computed, which will be one of the features used for the recognition and classification if numeral. Further the morphological thinning is applied. The other features such as location of end points and line junctions are obtained. Based on these features the numerals are classified. The whole process is described in the following sub sections.

2.1 Image pre-processing

The hard copy of the document containing numerals is scanned using optical scanner. The scanned document will be a color image, which will be converted to a binary one by applying thresholding. The process of throsholding basically converts the gray scale image into binary image. In this binary image the pixels corresponding to the numerals will be black and the background pixels will be white. A negative transformation is applied on this image, which will convert the pixels corresponding to the numerals to white and background pixels to black. The image may contain the noise in the form of tiny dots, due to the optical sensors. The presence of these dots will lead to miss detection of each dot as a numeral. These dots are eliminated by applying the morphological opening operation. All regions which contain less than 20 pixels are removed through this operation. The rows containing the numerals are detected from the document and these rows are segmented by applying the row clipping algorithm. It is assumed that the document contains the numerals in unrotated form. In this document the scanning is performed from left-top most corner and when a white pixel is encountered, that row is considered as the starting of the first row of the numerals. The end of the row containing this set of numerals is identified by locating immediate next row which does not have any white pixels. This process will yield us one complete row of numerals. This process is repeated further for the rest of the document to segment all the rows containing numerals. Each row containing numerals is further segmented into individual single numeral by extracting all the connected components having pixel values of 1. This segmented numeral image is subjected to feature extraction.

2.2 Feature extraction

A typical font of printed Kannada numerals is shown in Figure 1. The first stage in the feature extraction is to extract the number of regions from the numeral image. The main reason for detecting the regions is to identify the number of holes in the numeral image. On segmentation of each numeral, some of the boundary pixel values will be 1. Since this image contains one region, it is subjected to negative transform so as to get the number of holes from the image. The regions generated around the boundary will lead to false detection of holes. Hence these regions are connected to form a single region by padding rows containing zero values at the top and bottom of the image and columns containing zero values on left side and right side of the image. The number of connected components will gives us the exact number of regions. The process of extracting regions is shown in Figure 2. The numeral 6 in the image has three regions indicating that the numeral contains two holes and one region corresponds to the outer region of the numeral.

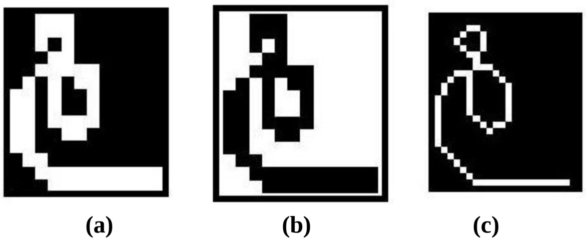


Fig. 2: Obtaining regions from the numeral, (a) Original numeral image, (b) inverted image after padding of zeros on all side which contains 3 regions, (c) image after morphological thinning.

Once the regions are obtained from the numeral image, the next process will be thinning. The morphological thinning algorithm is applied iteratively till the thickness of numeral in the image is reduced to single pixel width.

From the thinned image the end points of the numeral are obtained by applying the neighbourhood process at each pixel. If a pixel is an end point of the numeral then there will be only one white pixel in its 3 × 3 neighbourhood. Additional features such as the pixels which are junctions for three and four line joins are also found out. The pixels with three line joins will have three white pixels and four line joins will have four white pixels in their 3 × 3 neighbourhood. The number of such pixels and their coordinate values are stored for further processing. Figure 3 shows the thinned Kannada numerals which are divided into zones of five rows and five columns.

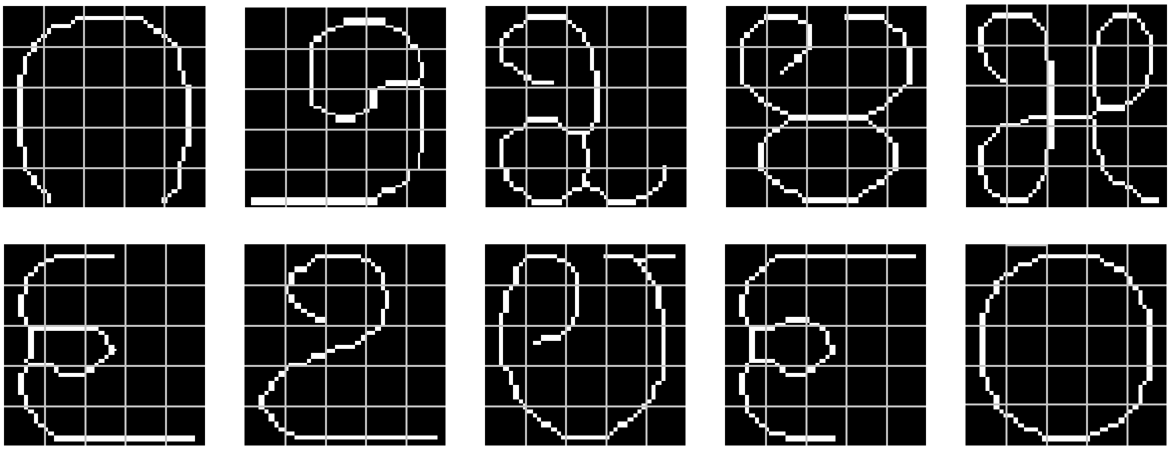


Fig 3: Kannada numerals after thinning and zoning

By studying the properties for following unique properties of each numeral are found out, which will uniquely identify that specific numeral.

The numeral one does not have any hole and it has two end points. Both the end points of this numeral are always in the lowest row of the zones, whereas, one of the end point will be located within initial two columns and other column will be located within last two columns. Numeral two has one or two end points and may have one hole. One of its end points will be located at the last row of the first column in the grid. These features will uniquely identify the numeral two.

The numeral three has similar features of seven. It has at least one end point and one hole. Its last end point will be located in the last two rows of the last column. The discriminating feature between three and seven are that the numeral seven have a line segment spanning at least 80 percent of the last row. The numeral four has at least one hole and one end point. If two holes are found then there will be only one end point and vice versa. The end points of this numeral are in the upper half zone of the grid.

The features of numeral five are that it has at least two holes. It may even have three holes; in that case it will have only one end point. It one of the end points will be located in the extreme last row and column of the zone. The numeral six will have two to three end points. If it has two end points in that case it will have one hole, otherwise it will have no holes. The top end point will be in the first row of the grids and the last end point will be in the last row and last column of the grid.

As discussed earlier, the numeral seven has similar features as three except that it does not have any hole in its centre part. Another discriminant feature is that it has a straight line component in the lower row of the zone. The numeral eight may have one, two or three end points. But all these end points are located in the upper half of the grids.

The numeral nine have two or three end points. One end point is located in the last column of the first row and another end point is in the central part of last row. It may have zero or one hole. If it has one hole then there will be only two end points. The last digit is zero, which has no end points but will have one hole, by which it can be uniquely identified.

Based on the above defined features the numerals are recognized and classified. There are possibilities of generation of false edges while applying thinning process. The generation of spurious end points are avoided by evenly applying the thinning about the centre of numeral edge. The whole process is presented in the form of algorithm as follows:

Algorithm: Classification of numerals based on zoning

Input: *Image containing printed Kannada numerals*

Output: *Text file containing the numerals corresponding to the input image*

Process:

Step 1: Convert input image into binary image

Step 2: Apply negative transformation to convert pixels of numerals to white and background to black.

Step 3: Remove the noise which will be in to form of tiny dots.

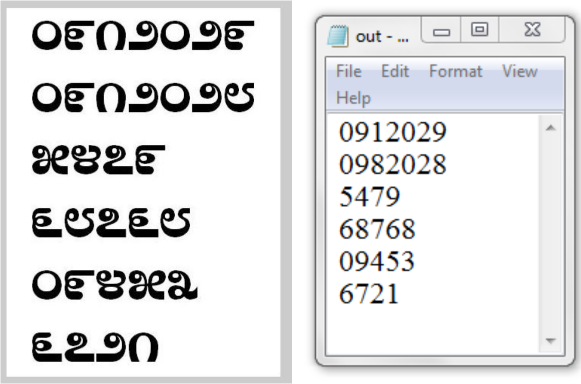
Step 4: Apply row segmentation to extract each row of numerals and then column segmentation on each row to extract each numeral.

Step 5: Apply morphological thinning over segmented numeral.

Step 6: Compute the number of end points in each numeral and also store the coordinate location of those end points.

Step 7: Based on the features discussed in this section, classify the numerals.

End.



೦೯೧೨೦೨೯
೦೯೮೨೦೨೮
೫೪೭೯
೬೮೭೬೮
೦೯೪೫೩
೬೭೨೧

```
0912029
0982028
5479
68768
09453
6721
```

Fig 4: Input image containing Printed Kannada numerals and output file containing recognized numerals.

3. RESULTS

The printed Kannada numerals are generated using Nudi 4.0 Kannada word processing software. For the implementation of proposed algorithm is coded using Matlab 7.0. Figure 4 shows a document image containing printed Kannada numerals which is the input. The proposed method generated the output as shown in the Figure. It can be noticed that for this particular font the proposed method could accurately recognize all the numerals.

4. CONCLUSION

A simple zoning concept based on the features namely holes and locations of end points are used for the reorganization and classification of printed Kannada numerals. The noise and generation of spurious end points are taken care by the proposed algorithm. The proposed method could correctly recognize most of the numerals generated using Nudi 4.0 Kannada software. But for some fonts it failed to recognize due to indiscriminate mismatch of identified features. Further, additional features like distance measures and line crossings may be used to improve the reorganization rate.

5. REFERENCES

[1] Krishnamurti, 2003, 78.

[2] Steever, 1998, 129-131.

[3] "Awardees detail for the Jnanpith Award", Official website of Bharatiya Jnanpith. Bharatiya Jnanpith. Retrieved 2008-05-12.

[4] "Declaration of Telugu and Kannada as classical languages". Press Information Bureau. Ministry of Tourism and Culture, Government of India. Retrieved 2008-10-31.

[5] "Old Kannada". Retrieved 2009-05-07.

[6] Gorman L. O' and Kasturi R. 1995. Document image analysis, IEEE Computer Society Press.

[7] Ravindra S. Hegadi 2011. Template Matching Approach for Printed Kannada Numeral Recognition. In Proceedings of the International Conference on Computational Intelligence and Information Technology (CIIT), Pune, India, 480-483.

[8] Sanjeev Kunte R. and Sudhakar Samuel R. D. 2007. An OCR system for printed Kannada text using two-stage Multi-network classification approach employing Wavelet features. In Proceedings of the IEEE Computer Society International Conference on Computational Intelligence and Multimedia Applications, India, 349–355.

[9] Ashwin T. V. and Sastry P. S. 2002. A font and size independent OCR system for printed Kannada documents using support vector machines. Sadhana, 27(1), 35-58.

[10] Karthik Sheshadri, Pavan Kumar T Ambekar, Deeksha Padma Prasad and Dr. Ramakanth P Kumar 2010. An OCR system for Printed Kannada using k-means clustering. In Proceedings of the IEEE International Conference on Industrial Technology (ICIT), Chile, 183-187.

[11] Ravindra S. Hegadi 2012. Classification of Kannada Numerals using multi-layer Neural Network. In Proceedings of the International Conference on Advances in Computing (ICADC), Bangalore.