Optical Character Recognition (OCR) for Kannada numerals using Left Bottom 1/4\textsuperscript{th} segment minimum features extraction

K.S. Prasanna Kumar
Research Scholar, JJT University, Jhunjhunu, Rajasthan, India
prasannakumarks@acharya.ac.in

Abstract

In this paper an algorithm to Optical Character Recognition (OCR) for Kannada numerals is discussed. The novelty exists in segmentation of the numeral into four equal parts and using one of these parts i.e., left bottom segment to extract recognition features. The algorithm also proposes a single conflict resolution technique to resolve conflicts while conflicting features are encountered. A minimum number of features are extracted by the algorithm so as to improve the response time.

1. Introduction

The history of Kannada literature can be traced back to even before the Rastrakuta King Nripatunga or Amoghavarsha (c. 815). The earliest work extract is a treatise or poetics called Kavirajamarga by the above rulers. The King even makes reference to writings of his earlier periods, which indicates the existence of language even before his time[1]. Considering the rich literary heritage, the numerical system of the language has undergone number of transformations to reach the present stage/shapes. Unlike other languages, the different fonts make little effect on the overall shapes of the numbers. Printed numeral recognition is an integral part of the OCR and hence this study.

There are a number of authors who have developed algorithms/systems for numerical recognition and a number of papers published based on feature extraction. But this paper proposes to extract minimum number of features and achieve better results.

2. Data Set

All pattern recognition systems have the following main components for information extraction and processing, which forms the kernel of any OCR.

- Image Acquisition: acquiring the input which could be in gray, colour or binary format.
- Binarization: the candidate input is to be converted into binary form.
- Noise removal: various techniques of cleaning, smoothing, enhancing, pre-segmentation, normalization, etc are used to remove noise.
- Data extraction: this is also called as feature extraction; gives data from perspective areas.
- Character recognition: identify the character to help post-processing.

There are a number of word processors available today, with each one of them providing number of fonts. It is easy to get the Kannada numbers printed and use them in various applications. The candidate characters are then scanned with 300 DPI to binaries it using global threshold and store it in binary file format. These scanned segments of numerical images are prone to noise due to paper quality, scanners, tilt, etc. Hence these binary files are filtered to remove noise before they could be used for processing the numerical images. The noise is removed by using morphological opening operations.

Pre-processing is the preliminary step which transforms the data into a form that is more easily and effectively processed. The main task in pre-processing is to capture data and to decrease the noise that causes a reduction in the recognition rate and increases the complexities. Hence, pre-processing is an essential stage prior to feature extraction, as it controls the suitability of the results for the successive of the algorithm.

3. Related work

Pattern recognition’s oldest technique was used for recognising characters. In the early days work was more concentrated on Chinese [2], Japanese and Latin characters [3]. The OCR study in this paper deals with analysing the shapes and visual property of Kannada characters and defines a set of features, which help in identifying candidate character.

Spitz[4] used optical density distribution of characters and frequently occurring word shapes characteristics for recognising Latin set of character.

In Zone based hybrid approach,[5] the character centroid is computed and the image is further divided into fifty equal parts. The average distance from the character centroid to the each pixel presents in the image is computed. Similarly the zone centroid is computed and average distance from the zone centroid to each pixel present in the zone is computed. This is repeated for all fifty zones. Zones that are empty are considered as null zones.
and omitted. Hundred features are extracted. Nearest neighbour classification and feed forward back propagations neural network classification is used.

When a list of feature extraction methods for character recognition is reviewed, feature extraction method are seen to includes Unitary Image transform, Template matching, Deformable templates, Projection Histograms, Contour profiles, Zoning Geometric moment invariants, Zernike moments, Spline curve approximation, Fourier descriptors, Gradient feature and Gabor feature[6].

4. Pre-processing

Global image thresholding is the process of distinguishing the image from its background, so thresholding is applied to grey-level candidate scanned image. Thresholding is categorized into two main categories i.e., global and local threshold. Global thresholding methods choose one threshold value for the entire document image, which is often based on the estimation of the background level from the intensity histogram of the image; hence, it is considered a point processing operation and local adaptive thresholding uses different values for each pixel according to the area information. There are hundreds of thresholding algorithms which have been published.

Global thresholding method is used to reduce the grey-level image to a binary image. The candidate image is assumed to have two classes of pixels called foreground and background. The purpose of a global thresholding method is to automatically specify a threshold value, T, where the pixel values below it are considered foreground and the values above are background. A simple method is chosen where the mean or median value of all the pixels in the input image, the mean or median will work well as the threshold, however, this may not be the case if the pixels are not uniformly distributed in an image. A more sophisticated approach is to use a histogram of the image pixel intensities and use the valley point (minimum) as the threshold.

Local thresholding technique is used with the candidate image having non-uniform background illumination or different backgrounds. If the global thresholding method fails to separate the foreground from the background then this is due to the fact that the histogram of such image has more than two peaks making it difficult for a global thresholding technique to separate the objects from the background. Hence local thresholding method is used. The local thresholding technique developed for customized applications may not be consistent. The results could be over thresholding or under thresholding depending on the contrast and illumination. A number of surveys have compared different thresholding techniques. The work of Trier and Jain evaluated the performance of well established locally adaptive binarization methods [7]. These techniques were compared using a criterion based on the ability of an OCR module to recognize handwritten numerals from hydrographical images. In this evaluation, the Niblack’s method, [8], appears to be the best. This was applied for a specific application on certain hydro- graphic images using an OCR system. However, as stated by the authors, if different sets of images use different feature extraction methods and classifiers, then this observation may not be accurate and another method could outperform the Niblack’s method. The Niblack’s method calculates the threshold by shifting a window across the image, and using local mean, µ, and standard deviation, σ, for each center pixel in the window. The threshold value for a pixel within fixed neighbourhood is a linear function of the mean and standard deviation of the neighbourhood pixels, with a constant gradient of T(x, y), to separate objects effectively. Then the threshold is given by equation (1). The value of k is used to get/vary the size of candidate object.

\[ T(x, y) = \mu (x, y) + k \sigma (x, y) \]  

(1)

Noise refers to stochastic variations as opposed to deterministic distortions, such as shading or lack of focus. There are different types of noise that are related to the electronic capturing devices or the light source used; types of noise are photon, thermal, On-Chip electronic and quantization. Most of the noise may be eliminated by the capturing sensors.

OCR is helped from the reduction of noise at the pre-processing stage. This provides a substantial improvement in the reliability and robustness of the feature extraction at recognition stages of the OCR system. A commonly seen noise in binary images is of isolated pixels, salt-and-pepper noise or speckle noise. Thus, the processing of removing this type of noise is called filling, where each isolated pixel salt-and-pepper “island” is filled in by the surrounding “sea” [9]. In grey level images or median filters and low-pass filters such as average or Gaussian blur filters proved to eliminate isolated pixel noise. Gaussian blur and average filters are a better choice to provide smooth texture to the image.

Dilation and Erosion are morphological operations which increase or decrease the object size. Erosion makes an object smaller by removing or eroding away the pixels on its edges. Dilation makes an object larger by adding pixels around its edges. There are two general techniques for erosion and dilation: the threshold and masking techniques.

The threshold technique looks at the neighbours of a pixel and changes its state if the number of differing neighbour pixels exceeds a threshold. Basically, if the number of zero pixels in

\[ T(x, y) = \mu (x, y) + k \sigma (x, y) \]  

(1)
the neighbourhood of a pixel exceeds a threshold parameter then the pixel is set to zero[10].

The dilation process does the opposite of erosion. It counts the value of pixels next to a zero pixel, if the count exceeds the threshold parameter, then the zero pixel is set to the value of the pixel.

The masking technique uses an n x n (3x3, 5x5, etc.) array of 1s and 0s on top of an input image and erodes or dilates the input. Using masks, the direction of erosion or dilation can be controlled. Square masks are more widely used such sizes are 3x3, 5x5, 7x7 and so on with other sizes is being used [11].

Mask erosion is the opposite of dilation. It applies an n x n mask on the image so that the centre of the array is on a zero. If any of the 1s coefficients in the mask overlap a white pixel (255) in the image then it is set to zero. Vertical mask erosion removes the top and bottom rows from an object, horizontal mask removes the left and right columns and the horizontal and vertical masks remove pixels from all edges.

5. Feature Extraction Method

A simple property of the pixel is its intensity. One of the natural ways is to extract this characteristic through thresholding. This is a technique that separates a pixel to object or background by comparing its gray level. Any binarization technique is usually based on gray levels, which could be divided into global and local thresholding. Global thresholding uses a single threshold for the entire image. A pixel with gray level lower than the threshold value is considered black and other as white. Adaptive binarization method computes the threshold for each pixel based on a neighborhood of the pixel. If a pixel in the original image has a gray level higher than the threshold surface value at the corresponding location, it is considered as black.

All pattern recognition systems have two functions viz. the description of the pattern or feature extraction and the second the identification of the pattern or pattern classification. This paper concentrates on the feature extraction stage followed by a dimensionality reduction procedure for lowering the computational complexity.

The importance of the method is seen in the least number of features extracted and its success rate in recognising the candidate character. There are a number of methods extracting hundreds to thousands of features and processing it to arrive at a known character. This paper however presents the least number of features required to recognise the character, to be precise only eight values.

The principal features extracted are based on the accuracy and simplicity of detection, computational speed, independence of size & font and classifier design. The following is the method to detect features.

6. Method

Kannada language is one of the major Indian languages. The list of Kannada numbers is shown in table 1.

<table>
<thead>
<tr>
<th>English numerals</th>
<th>:</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kannada numerals</td>
<td>:</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

Table 1: English and Kannada numerals

The method begins with drawing a rectangular boundary which is considered as a pre-processing tasks wrt this paper. The character under consideration is shown in fig.1.

![Figure 1: Rectangular boundary](image)

The segmented number is normalized to a convenient grid size. The candidate pattern is divided into four equal quarters and named as top left (Q1), top right (Q2), lower right (Q3) and lower left (Q4) as shown in fig 2.

![Figure 2: Candidate numeral divided into 4 quarters](image)

The fourth quarter Q4 is considered for extracting the characteristic features of numerals 0 to 9 as shown in fig.3.

![Figure3: Q4 of 0 to 9 from top left to bottom right](image)
The fourth quarter Q4 is segmented to various regions by an alternate numbering of edges and nodes as in fig 4.

Figure 4: Numbering method

The above alternate numbering of edges and nodes is utilised to derive a vector. The vector consists of 1 if there exist phenomenal number of black pixels in the edge or a node and a 0 otherwise.

7. Algorithm:
This algorithm provides the basis for the development of Knowledge base used in OCR. The algorithm is as follows:

Step 1: Let I be the candidate image/numeral; X=true.
Step 2: I is divided into four quarters viz. Q1(I), Q2(I), Q3(I) & Q4(I).
Step 3: If X, Q4(I) is bounded with rectangular box to get B(Q4(I))
else Q2(I) is bounded with rectangular box to get B(Q2(I))
Step 4: Vector V is extracted from the resultant of step 3 as follows:
   i. V is a vector of order 1 X 8
   ii. $v_i = 1$: if the segment i has contrasting (black) pixels.
   iii. $v_i = 0$: if the segment i has non contrasting (white) pixels.
Step 6: If V = {10001000} OR V = {01010101} Set X = false; GOTO Step 3
Step 5: V is compared with the knowledge base to get A
Step 6: Display A
Step 7: END

8. Knowledge Base construction with the help of the above algorithm.
The knowledge base is a repository that contains of vector for each of the 10 numeric literals. The vectors along with Q4 are shown in figure 5.

Figure 5: Q4 with its corresponding vector

The tabulation of the above is shown in the table 1:

<table>
<thead>
<tr>
<th>Digit</th>
<th>Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>[1,0,0,0,1,0,0,0]</td>
</tr>
<tr>
<td>1</td>
<td>[1,0,0,0,0,1,0,0]</td>
</tr>
<tr>
<td>2</td>
<td>[0,1,0,1,0,1,0,0]</td>
</tr>
<tr>
<td>3</td>
<td>[0,1,0,1,0,0,0,1]</td>
</tr>
<tr>
<td>4</td>
<td>[1,0,1,1,0,1,0,0]</td>
</tr>
<tr>
<td>5</td>
<td>[0,1,0,1,0,1,0,1]</td>
</tr>
<tr>
<td>6</td>
<td>[1,1,1,0,1,0,0,0]</td>
</tr>
<tr>
<td>7</td>
<td>[0,1,0,0,1,0,0,1]</td>
</tr>
<tr>
<td>8</td>
<td>[1,0,0,0,1,0,0,0]</td>
</tr>
</tbody>
</table>
The system however encounters a conflict as expected. This is due to the fact that only minimum numbers of features are extracted. The conflict occurs when lower left quarter of numbers 0 & 8 and number 5 & 9 are considered. This is not a regular conflict in recognition, but was found to happen in number of fonts that we tried this algorithm during the testing phase.

The algorithm not only helps in recognition of the numeral but also incorporates a conflict resolution without making much change to the algorithm. The conflict resolution procedure considers the top right quarter i.e., Q2 and goes through the same series of steps. The procedure is as follows.

Considering Q2 of 0 and 8 sown in figure-6

By the algorithm the vector for 0 is calculated to be \([1,0,0,0,1,0,0,0]\) and for 8 it is calculated to be \([0,1,1,0,1,0,0,0]\) which is distinct. The algorithm takes advantage of this factor to differentiate between 0 and 8. Similarly it could be seen that for a 5 and a 9, the same procedure is helpful. The figure 7 indicates the Q2 of 5 and 9.

By the algorithm the vector for 5 and 9 is \([0,1,0,1,0,1,0,0]\) and \([1,1,0,0,1,0,1]\) which is again distinct and helps to recognize the numeral.

**9. RESULTS AND DISCUSSIONS**

The experiments have illustrated that the minimum feature extraction method concept can be applied successfully to resolve the Optical Character Recognition problem. As it is seen, a few hundred features are extracted by almost all the OCR algorithms; however it is the first time an attempt to reduce the number of features is experimented successfully. As illustrated only eight values act as an indicator to each candidate image, except for two numbers which are resolved by the same method using conflict resolution incorporated in the algorithm. There are many factors that affect the performance of OCR system for Kannada Script. The recognition rate of OCR system with the candidate image of Kannada numbers is quite high. An encouraging 98% results are achieved. However, other kinds of preprocessing and feature extraction models with Q_i may be tested for a better recognition rate in the future research in OCR System. The test set used in this experiment is of six popularly available different types of fonts and sizes.

**10. References**


[5]. S.V.Rajashekararadhya and Dr. P. Vanaja Rajan, “Efficient Zone based feature extraction algorithm for handwritten numeral recognition of four popular south Indian scripts”, JITIT, pp 1171-1181.


