Word-wise Script Identification from Bilingual Documents Based on Morphological Reconstruction

B.V.Dhandra, H.Mallikarjun, Ravindra Hegadi, V.S.Malemath
P.G.Department of studies and research in Computer Science, Gulbarga University, Gulbarga India
dhandra_b_v@yahoo.co.in, mhangarge@yahoo.co.in

Abstract

In a multi-lingual country like India, English has proven to be the binding language. So, a line of a bilingual document page may contain text words in regional language and numerals in English (printed or handwritten). For Optical Character Recognition (OCR) of such a document page, it is necessary to identify different script forms before running an individual OCR of the scripts. In this paper an automatic technique for script identification at word level based on morphological reconstruction is proposed for two printed bilingual documents of Kannada and Devnagari containing English numerals (printed and handwritten). The technique developed includes a feature extractor and a classifier. The feature extractor consists of two stages. In the first stage, shape (eccentricity, aspect ratio) and directional stroke features (horizontal and vertical) are extracted based on morphological erosion and opening by reconstruction using the line structuring element. The average height of all the connected components of an image is used to threshold the length of the structuring element. In the second stage, average pixel distribution is obtained from these resulting images. The k-nearest neighbour algorithm is used to classify the new word images. The proposed algorithm is tested on 2250 sample words with various font styles and sizes. The results obtained are quite encouraging.

1. Introduction

With recent emergence and widespread application of multimedia technologies, there is an increasing demand to create a paperless environment. Hence, document image processing in general and Optical Character Recognition (OCR) in particular is playing an important role in transformation of the traditional paper based environment to truly paperless electronic environment.

In India, English has proven to be the binding language due to the diversity of languages and scripts. Therefore, a bilingual document page may contain text words in regional language and numerals in English. So, bilingual OCR is needed to read these documents. To make a bilingual OCR successful, it is necessary to separate portions of different script regions of the bilingual document at word level and then identify the different script forms before running an individual OCR system. Among the works reported in this direction, to distinguish between various Indian languages/scripts at word level [4, 7, 9, 12 and 14] are addressed only alphabet-based script identification, whereas numeral script identification is ignored. But, it is the fact that, the large number of bilingual documents contains text words in regional languages and numerals in English. For example, Newspapers, Magazines, Books, Application forms, Railway Reservation forms etc. This has motivated us to develop a method for automatic script identification of text words and numerals (printed and handwritten) in bilingual documents. Here, we also made an attempt for script identification of handwritten English and Kannada numerals.


In Section 2, we provide an overview of some discriminating features in the characters of Kannada, Devnagari and English numerals. In Section 3, a proposed new method for identifying the three scripts, Kannada, Devnagari and English numerals is described. In Section 4, the proposed algorithm is presented. The experimental details and results obtained are presented in Section 5. Conclusion is given in Section 6.

2. Discriminating features in characters of Kannada, Devnagari and English numerals

The feature extraction is the integral part of any recognition system. The aim of feature extraction is to identify patterns by means of minimum number of features that are effective in discriminating pattern classes. The proposed algorithm is inspired by a simple observation that every script/language defines a finite set of text patterns, each having a distinct visual appearance [18], and hence every language could be identified based on its discriminating features.

To develop the feature set, we first studied the document images and determined which visual features guided our human script identification. This analysis focused on two discriminating factors: the structural primitives like strokes (i.e. distribution of strokes in different directions) and global shape features (aspect ratio, eccentricity).

Most of the Hindi (Devnagari) language characters (alphabets) have a horizontal line at the upper part. In Devnagari, this line is called sirorekha. However, we shall call them headlines. When two or more Devnagari characters sit side by side to form a word, the sirorekha (shown in Fig. 1 (b)) or headline touch one another and generates a big headline [11], which is used as the major feature to distinguish the Kannada text words and English numerals. It can be observed that a distinct property of the English numerals is the existence of the vertical strokes like structure. By the experiment, we noticed that the vertical strokes in digits like 1, 3, 4, 6, 8, 9, and 0 are more dominant than that of horizontal strokes. It is also observable fact that the Kannada characters have more number of horizontal strokes than the vertical strokes (at a specified threshold). Hence, these directional features of strokes are considered to be more potential features to distinguish every script. These features are extracted using morphological opening by reconstruction with fill holes.

3. Proposed Method

The proposed method is designed based on the connected component analysis. These are used for thresholding the length of the structuring element for morphological opening by reconstruction as well as for global shape features extraction as discussed in Section 3.2 and 3.3.
3.1. Pre-processing

The documents are scanned using HP Scanner at 300 DPI, which usually yields a low noise and good quality document image. The digitized images are in gray tone and we have used Otsu’s global thresholding approach to convert them into two-tone images. Threshold is a normalized intensity value that lies in the range \([0, 1]\). Otsu’s method chooses the threshold to minimize the interclass variance of the thresholded black and white pixels. The two-tone images are then converted into 0-1 labels where the label 1 represents the object and 0 represents the background. The small objects like, single or double quotation marks, hyphens and periods etc. are removed using morphological opening. The next step in pre-processing is skew detection and correction. However, we assume that the skew correction has been performed before pre-processing.

3.2. Segmentation, Aspect Ratio and Eccentricity

To segment the document image into several text lines, we use the valleys of the horizontal projection computed by a row-wise sum of black pixels. The position between two consecutive horizontal projections where the histogram height is least denotes one boundary line. Using these boundary lines, document image is segmented into several text lines. Similarly, to segment each text line into several text words, we use the valleys of the vertical projection of each text line obtained by computing the column-wise sum of black pixels. The position between two consecutive vertical projections where the histogram height is least denotes one boundary line. Using these boundary lines, every text line is segmented into several text words. The word wise segmentation is illustrated in Fig. 1. These word images are then used to compute the eight-connected components of white pixels on the image and produce the bounding box for each of the connected components. Further, the average aspect ratio [6] and eccentricity of all the connected components of an image (word) is found. The eccentricity is a contour based global shape feature [21]. It is defined as the length of major axis divided by the length of the minor axis [21].

3.3 Reconstruction of connected components and fill holes

To extract the characters or components containing strokes in vertical and horizontal directions, we have performed the erosion operation on the input binary image with the line structuring element. The length of the structuring element is thresholded to 70 % (experimentally fixed) of average height of all the connected components of an image. The resulting image is used for opening by reconstruction in the vertical and horizontal directions using a fast hybrid reconstruction algorithm [8]. The reconstructed images of three scripts are illustrated in Fig. 2. Further, the reconstructed images and input image are used for fill holes. For fill holes, we choose the marker image (erode image), \(f_m\), to be 0 everywhere except on the image border, where it is set to 1-f. Here \(f\) is the original image.

\[
f_m(x, y) = \begin{cases} 1-f(x, y) & \text{if } (x, y) \text{ is on the border of } f \\ 0 & \text{otherwise} \end{cases}
\]

Then \(g = [R^c_f (f_m)]^c\) has the effect of the filling the holes in \(f\) as shown in Fig. 2. Where, \(R^c_f\) is the reconstructed image.

Finally, these resulting images are used for feature extraction (as shown in Fig. 2). The feature values are defined as the ratio between the number of on pixels remaining in third, fifth and sixth row images of Fig. 2 to the total number of pixels in the input image. Thus, we obtained a set of five features including the aspect ratio and eccentricity for each word image.

3.4 K-Nearest neighbour Classifier

K-nearest neighbour is a supervised learning algorithm. It is based on minimum distance (Euclidian distance metric is used) from the query instance to the training samples to determine the k-nearest neighbours. After determining the k nearest neighbours, we take simple majority of these k-nearest neighbours to be the prediction of the query instance. The experiment is carried out by varying the number of neighbours (\(K=3, 5, 7\)) and the performance of the algorithm is optimal when \(K = 3\).

4. Proposed Algorithm

The various steps involved in the proposed algorithm are as follows:
1. Pre-process the input document image i.e. binarisation using Otsu’s method, and remove speckles using morphological opening.
2. Carry out the line wise and word wise segmentation based on horizontal and vertical projection profiles.
3. Compute the average aspect ratio and eccentricity of all the connected components of an input image.
4. Carry out the Morphological erosion and opening by reconstruction using the line structuring element in vertical and horizontal directions.

5. Perform the hole fill operation on the input image and the reconstructed images of step 4.

6. Compute the average pixel densities of the resulting images of step 5.

7. Classify the new word image based on the k-nearest neighbour classifier.

Figure 1. Word-wise segmentation of (a) Kannada and (b) Devnagari scripts

Figure 2. (a), (g) and (m) are input images of Kannada, Devnagari and English numeral. (b), (h) and (n) are the images of vertical strokes, (c), (i) and (o) are the reconstructed images of characters that contain vertical strokes with fill holes, (d), (j) and (p) are the images of horizontal strokes, (e), (k) and (q) are the reconstructed images of characters that contain horizontal strokes with fill holes and (f), (l) and (r) represents original image with fill holes of Kannada, Devnagari and English numerals respectively.

Figure 3. (a), (b) and (c) are the sample test images of Kannada, Devnagari and English numerals

5. Results and Discussion

For experimentation, 400 document pages obtained from various magazines, newspapers, books and other such documents containing variable font styles and sizes are used with an assumption that the document pages contain only text lines. These document pages are scanned using a flat-bed HP scanner at a resolution of 300 dpi. A sample image of size 256x256 pixels is selected manually from each document page and created a first data set of 1850 word images by segmentation. Out of these 1850 word images, Kannada, Devnagari are 750 each and 350 are English numerals. Another data set of 250 handwritten numerals of Kannada and English (each 125) is used by obtaining handwritten document pages from 50 writers and scanned at the above said resolution. With this an attempt is made to test the feasibility of the proposed algorithm for script identification of handwritten numerals in addition to printed text words with handwritten numerals. The classification accuracy achieved in identifying the scripts of first and second data set is presented in Table 1, 2, 3, 4 and 5. The average script identification results of KNN classifier presented in Table 1,2,3,4 and 5 are 96.10%, 98.61%, 94.2%, 92.89% and 98.53 respectively. Although the primary aim of this paper is achieved, that is the word-wise script identification in bilingual documents; it is a fact that, normally printed documents font sizes and styles are less varied. We therefore conducted a third set of experiment on 150 word images to test the sensitivity of the algorithm towards different font sizes and styles. These words are first created in different fonts using DTP packages, and then printed from a laser printer. The printed documents are scanned as
The ISM, DTP package is used for Kannada and Devanagari, Microsoft word for English numerals. On most commonly used five fonts of Kannada, Hindi and English are considered for experiment. For each font 10 word images are considered varying in font size from 10 to 36. Out of these 150 word images, Kannada, Devanagari and English numerals are 50 each. The Kannada font styles used are KN-TTKamanna, TTUma, TTNandini, TTPadmini and TT-Pampa. The Devanagri font styles considered are DV-TTAakash, TTBhima, TTNatraj, TTRadhika, and TTsurekh. Times New Roman, Arial, Times New Roman italic, Arial Black and Bookman Old Style of English numerals are used for font and size sensitivity testing. It is noticed that, script identification accuracy achieved for third data set is consistent.

In the reported work of [4, 9, 12], it is mentioned that, the error rate is more when the word size is less than 3 characters. Our algorithm works for even single character words, but it fails when words like च, ङ, marks like “)” and broken sirorekha’s are encountered in Devnagari. The touched and broken components of Kannada word images are not recognized correctly because of loss in aspect ratio. Arial Black numerals of size more than 16 points of English also misclassified. The proposed algorithm is implemented in MATLAB 6.1. The average time taken to recognize the script of a given word is 0.1547 seconds on a Pentium-IV with 128 MB RAM based machine running at 1.80 GHz. Since, there is no work reported for script identification of numerals at word level, to the best of our knowledge, the results of this work could not be compared.

6. Conclusion

In this paper, we investigated a tool of structural stroke primitives of the connected components present in different directions and global shape features for script identification at word level. The simplicity of the algorithm is that, it works only on basic morphological operations and shows its novelty for font and size independent script identification. Furthermore, our method overcomes the word length constraint of [4, 9, 12] and works well even for single component words. In Indian context, the problem addressed here is quite relevant, robust and efficient for word-wise script identification of numerals and text words in bilingual documents. This work is first of its kind to the best of our knowledge. This work can also be extended to other Indian regional languages.

7. Acknowledgement

The authors are grateful to Dr.P.Nagabhushan, Dr. G.Hemanth Kumar and Dr. D.S. Guru, Dept. of Computer Science, Mysore University, Mysore, for their helpful discussion and encouragement during this work.

### Table 1: Recognition results of printed Kannada words and English numerals

<table>
<thead>
<tr>
<th>Script/Language</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kannada</td>
<td>96.28%</td>
</tr>
<tr>
<td>English</td>
<td>4.07%</td>
</tr>
</tbody>
</table>

### Table 2: Recognition results of printed Devnagari (Hindi) words and English numerals

<table>
<thead>
<tr>
<th>Script/Language</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hindi</td>
<td>98.12%</td>
</tr>
<tr>
<td>English</td>
<td>0.9%</td>
</tr>
</tbody>
</table>

### Table 3: Recognition results of printed Kannada, Devnagari words and English numerals

<table>
<thead>
<tr>
<th>Script/Language</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kannada</td>
<td>90.3%</td>
</tr>
<tr>
<td>English</td>
<td>3.1%</td>
</tr>
<tr>
<td>Hindi</td>
<td>3.1%</td>
</tr>
</tbody>
</table>

### Table 4: Recognition results of printed Kannada words and handwritten English numerals

<table>
<thead>
<tr>
<th>Script/Language</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kannada</td>
<td>92.16%</td>
</tr>
<tr>
<td>English</td>
<td>6.38%</td>
</tr>
</tbody>
</table>

### Table 5: Recognition results of printed Devnagari (Hindi) words and handwritten English numerals

<table>
<thead>
<tr>
<th>Script/Language</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hindi</td>
<td>98.12%</td>
</tr>
<tr>
<td>English</td>
<td>1.06%</td>
</tr>
</tbody>
</table>
Table 6: Script identification results of handwritten numerals of Kannada and English

<table>
<thead>
<tr>
<th>Script/Language</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Kannada</td>
</tr>
<tr>
<td>Kannada</td>
<td>89.58%</td>
</tr>
<tr>
<td>English</td>
<td>10.64%</td>
</tr>
</tbody>
</table>

Figure 4. The first column shows vertical and horizontal reconstruction with fill holes of English numerals and second’s vertical and horizontal reconstruction with fill holes of Kannada numerals.

8. References
