Script Line Separation From Indian Multi-Script Documents

U. Pal and B. B. Chaudhuri
Computer Vision and Pattern Recognition Unit
Indian Statistical Institute
203 B. T. Road, Calcutta - 35, INDIA

Abstract

In a multi-lingual country like India, a document page may contain more than one script form. Under the three-language formula, the document may be printed in English, Devnagari and one of the other official Indian languages. For OCR of such a document page, it is necessary to separate these three script forms before feeding them to the OCRs of individual scripts. In this paper, an automatic technique of separating the text lines using script characteristics and shape based features is presented. At present, the system has an overall accuracy of about 98.5%.

1 Introduction

India is a multi-lingual multi-script country, where a single document page like a passport application form, examination question paper, money order form may contain words in two or more language scripts. So, there is a need for developing multi-script OCR system that would work in two stages: (1) separation of different script regions of the document (2) feeding of individual script regions to appropriate OCR system. This paper deals with the first stage.

There are 18 official Indian languages. Two or more of which may be written in one script. Thus, 12 different scripts are used for writing these languages (see Fig.1). Under three-language formula, the documents for a region are printed in English, Hindi (Devnagari) and the regional official language. Our aim is to find approaches of separating all script triplets formed in this way. To the best of our knowledge, this is a pioneering work of its kind on Indian scripts.

Among earlier studies, Spitz[6] described a method for English and Japanese text separation. Later on he [7] developed another method, based on optical density distribution of characters and frequently occurring word shape characteristics, to separate Han based from Latin based script. Using cluster based templates, an automatic script identification technique is proposed by Hochberg et al. [4]. Wood et al. [9] described an approach using filtered pixel projection profiles. Ding et al. [3] proposed a method for separating some European and Oriental scripts. Recently, Tan [8] described a fractal-based method for identification of Chinese, English, Greek, Russian, Malayalam and Persian text.

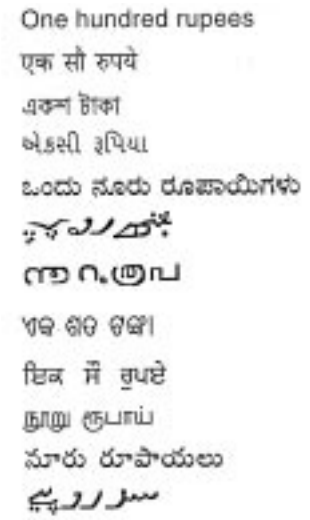

Fig.1: Examples of 12 Indian scripts. Top to bottom: English, Devnagari, Bangla, Gujarati, Kannada, Kashmiri, Malayalam, Oriya, Panjabi (Gurumukhi), Tamil, Telugu and Urdu.

2 Properties of Indian language scripts

The official languages of India are Assamese, Bangla, (Bengali) English, Gujarati, Hindi, Kankanai, Kannada, Kashmiri, Malayalam, Marathi, Nepali, Oriya, Panjabi, Rajasthani, Sanskrit, Tamil, Telugu and Urdu. Of them, Devnagari script is used to write Hindi, Marathi, Rajasthani, Sanskrit and Nepali language while Bangla script is used to write Assamese and Bangla (Bengali) languages. Our present work is concerned with script separation and not the language separation [6,9]. We note that there are 10 triplets formed by English, Devnagari with any one of the other scripts. Since character shapes of different scripts are not the same, a method which

works for scripts segmentation of one triplet may not work for other. Based on script shape characteristics, we put these 10 triplets into five different groups.

In some scripts (like Bangla, Devnagari etc.) many characters have a horizontal line at the upper part. This line is called *matra* (*sirorekha*) in Bangla (Devnagari). However, here, we shall call it as *head-line*. When two or more characters sit side by side to form a word the head-line portions touch one another and generate a long head-line, which is used as a feature to isolate one text line from the other. In some scripts (like Gujarati, Oriya etc.) the characters do not have head-lines but have vertical line-like structures. We use these heuristics also in our separation scheme.

In most Indian languages, a text line may be partitioned into three zones. The *upper-zone* denotes the portion above the head-line, the *middle zone* covers the portion below head-line and the *lower-zone* is the portion below base-line. For the text having no head-line, the mean-line separates upper-zone and middle-zone. An imaginary line, where most of the uppermost (lowermost) points of characters of a text line lie, is called as mean-line (base-line). Examples of zoning are shown in Fig.2.

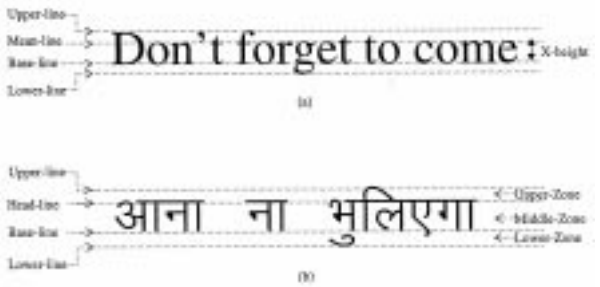

Fig.2: Example of zoning: (a) English (b) Devnagari script line. Here, head-line or mean-line and base-line separate the text line into three zones.

The English script characters can be upper-case or lower-case. But the case concept is absent in Indian scripts. We identify a text line as English upper-case if upper-line and lower-line coincides with mean-line and base-line, respectively. See Fig.2 for upper-line and lower-line.

3 Script lines separation from triplets

As mentioned earlier, we divided 10 script triplets into five groups. The groups and their separation approach are described bellow. Here we assume that the digitized images are converted into two-tone (0-1 level) images.

3.1 Separation of first group of triplets

The triplets in this group are (English, Devnagari, Bangla) and (English, Devnagari, Panjabi). Here each of the scripts, except English, have long head-lines. English script is distinguished by this head-line future.

Separation of English, Devnagari and Bangla text lines: If we take the longest horizontal run of black pixels on the rows of a text line then such run length for English line is usually much smaller than that of Bangla or Devnagari line. For example, see Fig.3. This information has been used to separate the English lines from other text lines. Separation of Bangla and Devnagari text line is tricky because of their structural similarity and we used some character level features for the purpose. For details see our paper [5].

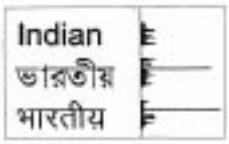

Fig.3: Longest horizontal run in each row (from top to bottom English , Bangla and Devnagari text)

It is noted that Panjabi (Gurumukhi) script is very similar to Devnagari script. For their separation, we use some extra distinctive features (like half vertical line, position of vertical line etc.) .

3.2 Separation of second group of triplets

The triplets in this group are (English, Devnagari, Urdu) and (English, Devnagari, Kashmiri). English and Urdu words do not have head-line. Using this information, at first Devnagari text lines are separated from English and Urdu. English and Urdu are separated as follows:

(a) Horizontal projection profile: If we compute the horizontal projection profile of the text lines, we get two prominent local maxima for English while only one maximum for Urdu. Also, for English text line these two maxima occur at mean-line and base-line region (Fig.4).

(b) Vertical runlength distribution: A distinctive characteristics of most of the English characters is the existence of vertical line-like structure. In Urdu alphabet, the number of such characters is very few. The vertical line-like structure is used as a separating feature. To detect this feature we compute the column-wise runs of black pixels of the components of a text line. A component has a vertical line-like structure if a black run length of that component is greater than half of the text line height. The text line height is defined as the normal distance between upper-line and lower-line. If 25% of the components in a text line have vertical line like structure then we assume that it is a English text line. Else, it is

Urdu. The 25% threshold value is obtained from an experiment over a large set of data.

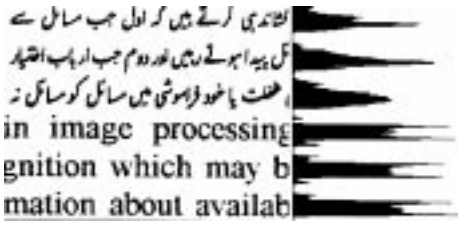

Fig.4: Horizontal projection profile of a typical document image containing Urdu a and English text.

(c) Distribution of lowermost points of the components: For the English characters having no descenders, the lower-most points lie on base-line. Otherwise, lower-most points lie on lower-line(i.e. characters g, j, p, q, y). The lowermost points of most characters of an Urdu text line do not lie on such lines. This property has been used as another feature for their separation . For a text line we compute two sets of lower-most points A and B corresponding to base-line and lower-line using nearest neighborhood rule. If the normal distance from the point to base-line is smaller than that to the lower-line, then the point belongs to the set corresponding to base-line. Else, it belongs to the set corresponding to lower-line. We separately compute the standard deviations σ_A and σ_B of the lowermost row values of the components of these two sets. The lowermost row values are normalized to make this feature size independent. It is found that if $\sigma_B + \sigma_A < 3.0$ then the text line can be considered as English. Otherwise, it is Urdu. This threshold is obtained from experimental data. Due to the dot of the characters like "i" and "j" or due to salt and pepper noise, sometimes an English text line may be wrongly identified as Urdu. To solve this problem we select only those components whose bounding box widths are greater than half of the average bounding box width of all component in the line. If at least two of the above features is obtained for a text line in favor of English then we identify that text line as English. Otherwise, we decide it as Urdu text line.

3.3 Separation of third group of triplets

In this group the triplets are (English, Devnagari, Gujarati) and (English, Devnagari, Oriya). Gujarati words do not have long head-line. Using the head-line information, Devnagari lines are separated from English and Gujarati. For the separation of Gujarati and English we segment the text lines into words and words into the characters. Word segmentation is done by vertical projection profile. Character segmentation is done as follows. Since, there is no head-line, we detect the mean-line and base-line from each line. Now, a scanning in the vertical direction from the mean-line is initiated. If during a scan, one can reach the base-line without touching a black pixel then this scan makes a boundary line between two characters.

For the separation of English and Gujarati text, the following two heuristic features are used.

(a) Position of vertical line with respect to its bounding box: Vertical lines in most of the Gujarati and Oriya characters are situated only in the right side of the character bounding box. In English, except the characters 'd', 'g' and 'q', the vertical lines are situated either in the left sides (e.g. E, F, b, h, k, p etc.) or both sides (e.g. H, M, m, U, u, N, n etc.) of the character bounding box. The first type of characters are called left-sided while the second type of characters are both sided. The right sided characters are similarly defined. For a line, we compute the numbers of right-sided characters (N_R), left-sided characters (N_L) and both-sided characters (N_B). If for a line, $(N_L + N_B) > N_R$ then we classify that line as English. Else, it is Gujarati.

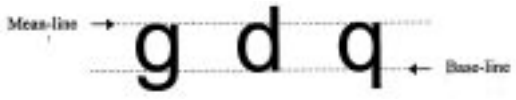

Fig.5. Example of some English character having part beyond the mean-line and base-line.

For some characters like I, i, l. t etc., it is very difficult to determine the position of their vertical lines because of their small widths. To avoid them, we select those components which have bounding box width greater than a threshold.

In Gujarati and Oriya script, the vertical lines of the characters do not go beyond base-line or mean-line. But for the characters 'd', 'g' and 'q', the vertical lines go beyond mean-line and base-line (See Fig. 5). Using this property we can restrict these three right-sided characters from falling in the Gujarati character class.

(b) Distribution of vertical component above mean-line: Here, we compute the number of characters in which the vertical line go above the mean line. In English, many characters like B, D, E, F, H, I, J, K, L, M, N, P, R, T, U, b, d, f, h, k, l, and t have vertical lines which go above the mean-line. But in Gujarati there is only one character where the vertical line goes above the mean-line. In Oriya script also, there is a few characters where the vertical line go above the mean-line . If the ratio of number of such characters to the total number of characters in a text line is greater than 0.25, then we classify that line as English. Else, it is Gujarati.

If any of above two conditions satisfy in favor of English, we classify that text line as English. Else, it is Gujarati.

Using this approach, the other triplet can be separated.

3.4 Separation of fourth group of triplets

Two triplets (Devnagari, English, Telugu) and (Devnagari, English, Kannada) belong in this group. Using the head-line Devnagari lines are separated from Telugu and English. Telugu and English text lines are segmented as follows. While English has vertical lines there is no Telugu character with vertical line-like structure. We compute the column-wise runs of black pixels of the components of a text line and we say a component has vertical line-like structure if the length of a black run of that component is greater than or equal to the *x-height* of the text line (*x-height* of a text line is shown in Fig. 2). If for 20% of the characters in a text line there exist vertical line structure then we assume that the line is a English text line. Else, it is Telugu.

The above arguments hold for Kannada script also.

3.5 Separation of fifth group of triplets

There are two triplets in this group namely, (English, Devnagari, Tamil) and (English, Devnagari, Malayalam). The head-line feature is used to separate Devnagari in this case also. For the distinction of English and Tamil text lines, we compute the following features.

(a) Distribution of vertical component above mean-line : The detection procedure of this feature is already described in 3.3.

(b) Horizontal run information : One of the most distinctive and inherent characteristics of most of the Tamil characters is the existence of three or more horizontal black runs. In English alphabet, the number of such characters is very few. Only the characters M, N and W have three or four black runs. We compute the number of components which have three or more runs in a text line. If 20% of the component in a line has 3 or more black runs then we assume that the text line is Tamil. Else, it is English.

If any of the two above features obtained from a text line is in favor of English, we classify that text line as English. Else, it is Tamil.

4 Results and Discussion

We applied our separation scheme on 120 different document images containing about 2000 text lines. We considered at least 9 document images from each triplet. The images were scanned from question papers, bank account opening application form, money order form, computer printouts, translation books etc.. We noted that the accuracy rates of script line separation of five groups were 97.6%, 99.2%, 98.7%, 99.3% and 97.7% respectively. The overall accuracy of the system was about 98.5%. We noted that most of the identification errors are obtained for short lines containing one word only. To reduce the error due to the short lines, we use a heuristic. If a line contains only one word then it is highly likely that it is the continuation of the previous line. Thus, when we get a very short line then we classify it to the class of its previous line.

Our scheme does not depend on the size of characters in the text line. Also, we noticed that this approach is font and case insensitive.

In this separation scheme, vertical line-like feature may not be detected properly for italic text lines. In that case we can use the algorithm due to Chaudhuri and Garain [1] to detect italic lines as well as to compute the slant angle of the characters. Now, instead of vertical line-like strokes, we search for slanted line-like strokes.

The work presented here is a step towards building a general OCR system that can work for all major Indian scripts. A bilingual (Bangla and Devnagari) OCR system is already working in our lab[2].

References:

1. B. B. Chaudhuri and U. Garain, "Automatic detection of italics, bold and all capital words in document images", *In Proc. 14th ICPR* , pp. 610-612, 1998.

2. B. B. Chaudhuri and U. Pal, "An OCR system to read two Indian language scripts: Bangla and Devnagari", *In proc. 4th ICDAR*, pp. 1011-1015, 1997.

3. J. Ding, L. Lam and C. Y. Suen, "Classification of oriental and European scripts by using characteristic features", *In Proc. 4th ICDAR* , pp.1023-1027, 1997.

4. J. Hochberg, L. Kerns, P. Kelly and T. Thomas, "Automatic script identification from images using cluster-based templates", *In Proc. 3rd ICDAR*, pp. 378-381, 1995.

5. U. Pal and B. B. Chaudhuri, "Automatic separation of words in multi-lingual multi-script Indian documents", *In Proc. 4th ICDAR*, pp. 576-579, 1997.

6 A. Spitz, "Multilingual document recognition", *Electronic Publishing, Document Manipulation, and Typography*, R. Furuta, ed. Cambridge Univ. Press, pp. 193-206, 1990.

7. A. Spitz, "Determination of the script and language content of document images", *IEEE PAMI.*, vol.19, pp. 235-245, 1997.

8. T. N. Tan, "Rotation invariant texture features and their use in automatic script identification", *IEEE PAMI*, vol. 20, pp. 751-756, 1998.

9. S. Wood, X. Yao, K, Krishnamurthi, and L. Dang, "Language identification for printed text independent of segmentation", *In Proc. ICIP*, pp. 428-431, 1995.