Multi-Script Line identification from Indian Documents

U. Pal, S. Sinha and B. B. Chaudhuri
Computer Vision and Pattern Recognition Unit
Indian Statistical Institute
203 B. T. Road, Kolkata-700108, INDIA
Email: umapada@isical.ac.in

Abstract

A document page may contain two or more different scripts. For Optical Character Recognition (OCR) of such a document page, it is necessary to separate different scripts before feeding them to their individual OCR system. In this paper an automatic scheme is presented to identify text lines of different Indian scripts from a document. For the separation task at first the scripts are grouped into a few classes according to script characteristics. Next feature based on water reservoir principle, contour tracing, profile etc. are employed to identify them without any expensive OCR-like algorithms. At present, the system has an overall accuracy of about 97.52%.

1. Introduction

India is a multi-lingual multi-script country, where a single document page (e.g. a passport application form, an examination question paper, a money order form, bank account opening application form etc.) may contain lines in two or more language scripts. So, there is a need for developing multi-script OCR system for such a country. To develop such multi-script OCR, it is necessary to identify different scripts before feeding them to their individual script OCR systems. This paper deals with a line-wise script identification scheme for Indian documents.

In India, there are 18 official (Indian constitution accepted) languages. Two or more of these languages may be written in one script. Twelve different scripts are used for writing these languages. Under the three-language formula, many of the Indian documents are written in three languages namely, English, Hindi and the state official language. For example, a money order form in the West-Bengal state is written in English, Hindi and Bangla, because Bangla is the state official language of West-Bengal. Previously we [5] developed a system to identify different scripts from triplets formed by English, Devnagari (Hindi language is written by Devnagari script) and a third state language scripts. A drawback of that system is that we need to know the type of script triplet before using script separation scheme. In this paper we propose a more general scheme to handle all the scripts. If a single document page contains all these twelve scripts, our present method can identify each of them without any prior knowledge of the document. To the best of our knowledge, this is the earliest work of its kind on Indian scripts.

Among the pieces of earlier work of script separation, one of the first attempts is due to Spitz[7]. He proposed a technique for distinguishing Han and Latin based scripts on the basis of spatial relationships of features related to the character structures. An identification scheme using cluster-based template of the scripts is proposed by Hochberg *et al.* [2]. Using rotation invariant multi-channel Gabor filter texture features, Tan [8] described a method for the identification of Chinese, English, Greek, Russian, Malayalam and Persian text. For details review of the pieces of other earlier work on script identification, see [6].

2. Properties of Indian Language Scripts

The official languages of India are Assamese, Bangla, English, Gujarati, Hindi, Kankanai, Kannada, Kashmiri, Malayalam, Marathi, Nepali, Oriya, Panjabi, Rajasthani, Sanskrit, Tamil, Telugu and Urdu. Among these, Hindi and Bangla are the first and secondmost popular languages in India and 4th and 5th most popular language in the world while English is the most popular language of the world. As stated above, the scripts used for the Indian languages are not all different. For example, 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. Twelve different scripts are used to write these 18 languages. These scripts are named as Devnagari, Bangla, English, Gujarati, Kannada, Kashmiri,

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 IEEE

Malayalam, Oriya, Gurumukhi (Panjabi), Tamil, Telugu and Urdu. Examples of different script lines are shown in Fig.2. Our present work is concerned with script separation and not the language separation.

In most Indian script alphabet system apart from vowel and consonant characters, called *basic characters*, there are compound characters formed by combining two or more basic characters. The shape of a compound character is usually more complex than the constituent basic characters.

In some scripts (like Bangla, Devnagari etc.) it is noted that many characters of these alphabets have a horizontal line at the upper part. In Bangla, this line is called *matra* while in Devnagari it is called *sirorekha*. However, in this paper, we shall call it as *head-line*. When two or more characters sit side by side to form a word in the language, the head-line portions touch one another and generate a long head-line, which is used as a feature for script identification (see Fig. 1(b)).

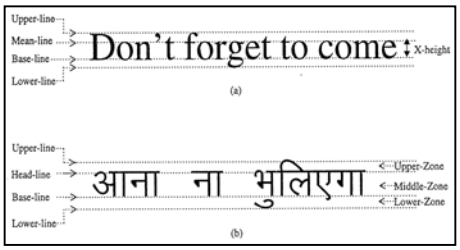


Fig.1: Different zones of English and Devnagari text line

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 of basic (and compound) characters below head-line and the *lower-zone* is the portion below base-line. Those text where script lines do not contain head-line, the mean-line separates upper-zone and middle-zone. The base-line separates middle-zone and lower-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). We call the uppermost and lowermost boundary lines of a text line as upper-line and lower-line. Examples of zoning are shown in Fig.1.

For detailed properties of Indian languages, see [5].

3. Preprocessing

Text digitization for our experiment has been done by a flatbed scanner (manufactured by HP, Model no ScanJet 4C). The digitized images are in gray tone and we have used a histogram based thresholding approach to convert them into two-tone images.

When a document is fed to the optical sensor either mechanically or by a human operator, a few degrees of skew is unavoidable. Skew angle of a text line in a digital image is the angle that the text line makes with the horizontal direction. Skew detection and correction are important for successful OCR and document structure analysis. The skew detection and correction techniques used here are described in our paper [4]. After skew correction the lines are segmented. For line segmentation we use a horizontal projection profile based technique, as described in [5].

4. Features Used for Script Identification:

The features are chosen with the following considerations: (a) Presence in characters of some scripts and absence in characters of at least one script (b) Robustness, accuracy and simplicity of detection (c) Speed of computation and (d) Independence of fonts, size and style of the text. Some of the principal features used in our identification scheme are as follows:

Head-line feature: If we take the longest horizontal run of black pixels on the rows of a text line then such run length for Bangla, Devnagari and Gurumukhi script will be much higher than that of other scripts. This is because characters in a word are connected by head-line in these scripts. For illustration, see Fig.2. Here row-wise maximum run is shown in the left part of Fig.2. This run information has been used to separate Bangla, Devnagari and Gurumukhi lines from other text lines.

Horizontal projection profile: If we compute the horizontal projection profile of the text lines, we note some distinct features among some of the scripts. For example, in some scripts (like Malayalam, Kannada, English etc.) we get two prominent local maxima whereas in some other scripts (like Telugu, Urdu, Kashmiri etc.) we obtain only one peak as shown in right side of Fig.2. From the projection profile of Urdu text line it can be noted that the peak of the projection profile occurs at the lower half of the text line. In text lines of English, Kannada and Malayalam scripts one peak occurs in upper half and the other peak occurs in lower half of the text line.

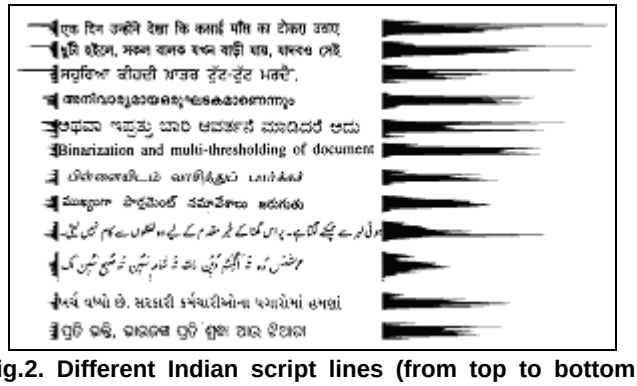


Fig.2. Different Indian script lines (from top to bottom: Devnagari, Bangla, Gurumukhi, Malayalam, Kannada, English, Tamil, Telugu, Urdu, Kashmiri, Gujrathi, Oriya) with their row-wise maximum run (left side) and horizontal profile (right side).

Water reservoir principle based feature: The water reservoir principle is as follows. If water is poured from one side of a component, the cavity regions of the component where water will be stored are considered as reservoirs [3]. By top (bottom) reservoirs we mean the reservoirs obtained when water is poured from top (bottom) of the component. (A bottom reservoir of a component is visualized as top reservoir when water will be poured from top after rotating the component by 180°). Similarly, if water is poured from left (right) side of the component, the cavity regions of the components where water will be stored are considered as left (right) reservoirs. For an illustration, see Fig.3. Here top, bottom, left and right reservoirs are shown for the English character X. The water flow levels of these reservoirs are also shown in this figure. By water flow level we mean the level above which water overflows from the reservoir.

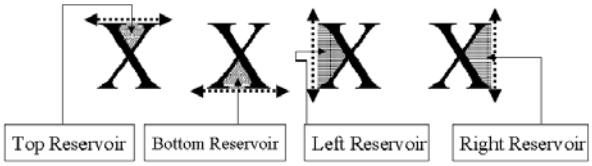


Fig.3. Top, bottom, left and right reservoirs are shown for the character X. Water flow level of reservoir is shown by dotted arrow.

In some scripts many reservoirs may be obtained from a particular side of the characters whereas in some other scripts many reservoirs may not be obtained from that side. For example, if we consider water reservoir from left then we get only five characters (a, s, x, y and z) in English text whereas in Telugu, Kannada and Malayalam scripts we will get many characters with reservoir from left. Left reservoirs rarely occur in Urdu script also. Left reservoirs obtained in Telugu and English text lines are shown in Fig.4. We use such information as a feature for script line identification.

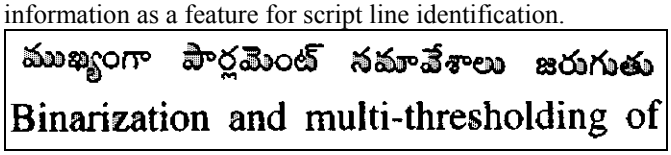
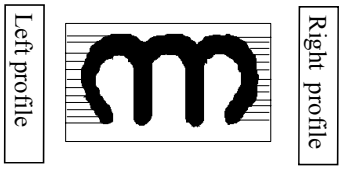


Fig.4: Left reservoirs are shown in Telugu and English lines.

Other reservoir based features like heights of the reservoirs, water flow level of the reservoirs etc. are also used for the script identification. Out of 52 (upper + lower case) characters in English only 15 characters have top reservoirs and water flow level of top reservoirs coincide either with mean-line (for lower case characters) or upper-line (for upper case characters). In Urdu there are many characters with top reservoir and water flow levels of top reservoirs do not follow any such fix position. They are random in nature. We use distribution of water flow level as a feature for script identification. We note the water flow levels (row values) of the characters of a text line. For English text line we generally get two row values. These values coincide either with mean-line row or upper-line row value. But for Urdu these row values are random. For a text line we form two sets A and B from these water flow row values. Let the largest water flow row belongs to A and the smallest row value belongs to B. Other intermediate row values are assigned either in A or B according to the nearest neighbour rule. For English text A contains the values which are similar to upper-line row whereas B contains values corresponding to mean-line row. But text lines of Urdu script do not follow this rule. We separately compute the standard deviations σ_A and σ_B of the water flow row values of the components of these two sets. We notice that for English the value of $\sigma_B + \sigma_A$ is nearly zero whereas in Urdu this value is very high.

Left and right profile: Suppose each character is located within a rectangular boundary, a frame. The horizontal or vertical distances from any one side of the frame to the character edge are a group of parallel lines which we call the *profile* (see Fig.5). If we compute left and right profile of the characters in a text line, we can notice some distinct difference in some of the scripts. For example, in both the left and right profiles of Malayalam script, most of the characters have one transition point because of their concave shape. By transition we mean change of the profiles from increasing mode to decreasing mode or vice-versa. But in some other scripts like Gujrathi, Tamil, English etc. this behaviour is absent. Similarly, if we consider profile from top then we notice one transition point in most of the Oriya characters. We use left and right profile feature for the identification of Malayalam script. We also use top profile feature for the identification of Oriya script. Left and right profile of a Malayalam character is shown in Fig.5.

Fig.5: Left and right profile of a character is shown.



Feature based on jump discontinuity: In this feature we check jump discontinuity between two consecutive boarder pixels of a component from a particular side of the component. We note that in some scripts (like Telugu, Kannada etc) such feature occurs in many characters whereas in other scripts few characters have such feature. We use this jump discontinuity occurrence frequency as a feature for identification. To compute this feature each vertical column of a particular component is scanned from top until it reaches a black pixel (P_i). Thus, for a component of width N we get N such P_is. To measure the discontinuity, we traverse from P_i to P_{i+1} clock-wise along the boarder of the component. During the traversal maximum horizontal displacement (d_{11}) and the number (m) of boarder pixels to

be traversed to go from P_i to P_{i+1} pixels are noted. This is done for all i, $(i = 1.....N-1)$. If for a character we get at least one d_{11} which is greater than 0.6 times the character width or at least one *m* which is greater than 1.3 times the character height then we decide that the feature based on jump discontinuity exist in that character**.** For an illustration, see Fig.6. Here Q_1 and Q_2 are two pixels obtained by vertical scan of two consecutive columns, and d_{11} is the maximum horizontal displacement incurred during the tracing from Q_1 to Q_2.

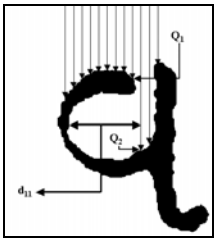


Fig.6: Example of jump discontinuity feature.

5. Script identification techniques

Based on the above principal features, we use a binary tree classifier for the script separation. The features are chosen at a non-terminal node to get optimum tree. A portion of the tree classifier is shown in Fig.7. From the tree it may be noted that some scripts belong both the sub-groups of a node. This is done to reduce the errors which may occur because of poor scanning, noise etc. We use head-line feature at the beginning of the tree. As mentioned earlier when two or more characters sit side by side to form a word in some Indian scripts like Bangla, Devnagari, Gurumukhi etc. the head-line portions touch one another and generate a long head-line. From a statistical analysis we noticed that the probability that a Bangla word will have at least one character with head-line is 0.994 and the corresponding probability for Devnagari is 0.997 [1]. Hence, the use of head-line is justified for the purpose. Sometimes because of touching some characters in Urdu and Kannada script lines may get big run. As a result these two scripts fall in both the groups. Thus, the head-line features divide the total scrips into two sub-groups containing Bangla, Devanagari, Gurumukhi, Urdu and Kannada scripts in one group and English, Tamil. Telugu, Oriya, Malayalam, Gujrathi, Urdu, Kannada in other group. To separate Kannada from Urdu, Bangla, Devnagari and Gurumukhi we check the number of peaks in the horizontal profile. We notice that in Kannada script there are two distinct peaks whereas other scripts have only one peak. To separate Urdu from Bangla, Devnagari and Gurumukhi script we note the position of the peak. Position of the peak is in the lower half of a text line in Urdu script whereas it is in the upper half for the Bangla, Devnagari and Gurumukhi scripts.

Separation of Bangla, Devnagari and Gurumukhi text line is tricky because of their structural similarity and we use some character level features for the purpose. To do so, segmentation of text line into words and word into characters is necessary. Word and character segmentation approach is similar to approach described in [5].

To separate Bangla, Devnagari and Gurumukhi lines we first distinguish Bangla from Devnagari and Gurumukhi scripts. Next, Gurumukhi script is separated from Devnagari script. To separate Bangla from Devnagari and Gurumukhi scripts we use some structural features in the characters of the words. Depending on the presence of the different structural features from the upper-zone and lower-zone of characters we identify Bangla from others. For details see our paper [4]. It is noted that Gurumukhi script is very similar to Devnagari script. For the separation between Gurumukhi and Devnagari line, we identify some distinctive features. Some of these features are based on contour tracing, number of crossings etc. For details see our paper [5].

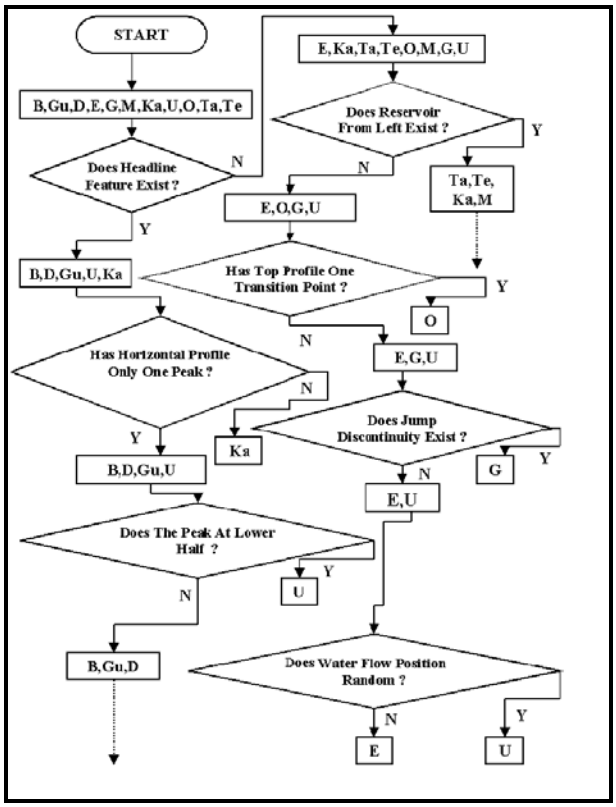


Fig.7. Flow diagram of the script identification scheme. (Here B=Bengali, D=Devnagari, E=English, Gu=Gurumukhi, M=Malayalam, Ta=Tamil, Te=Telugu, G=Gujrathi, Ka=Kannada, U=Urdu and O=Oriya).

For separation of English, Kannada, Tamil, Telugu, Oriya, Malayalam, Gujrathi and Urdu script lines we first use the left reservoir features. We notice the number of left reservoir in the characters in English, Oriya, Gujrathi and Urdu script is very few whereas in Tamil, Telugu, Kannada and Malayalam script lines there are many characters with left reservoirs. For example, see Fig.4 where left reservoirs of Telugu and English lines are shown. If 30% of the characters have at least one left reservoir then we classify that text line

in the group of Telugu, Kannada, Tamil and Malayalam. Else, we classify in the group of English, Oriya, Gujrathi and Urdu. For separation of Oriya character from English, Gujrathi and Urdu we use top profile features. It can be noted that in Oriya script most of the characters have top profile feature whereas this is absent in English, Gujrathi and Urdu. For the separation of Gujrathi script jump discontinuity feature is used. For the separation of Urdu and English we use water flow level features. For the separation of Telugu, Kannada, Tamil and Malayalam group we check number of distinct peaks in the projection profile. It can be noted that Telugu has one peak whereas Kannada, Tamil and Malayalam has two peaks. For separation Kannada, Tamil and Malayalam text lines, position of valley between two distinct peaks of the projection profile is noted. For Kannada the valley position situated in the left half of the profile while in Tamil and Malayalam the valley position situated in the right half. For example see Fig.8, where peak and valley positions are shown in these scripts. For separation of Tamil and Malayalam script we use left and right profile features. As mentioned earlier, there are many characters in Malayalam script which satisfy left and right profile features whereas in Tamil script only a few characters have this property. If more than 35% characters of a text line have left and right profile features we identify that line as Malayalam. Else the line is identified as Tamil. The threshold value is chosen from the experiment.

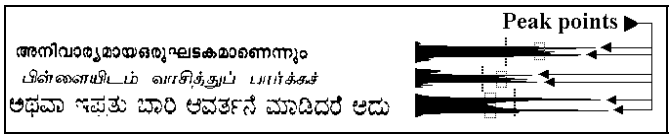


Fig.8. Peak and valley positions are shown. (top to bottom: Malayalam, Tamil and Kannada line). Valley positions are shown by dotted rectangles. Middle points of the profiles are marked by dotted lines.

6. Results and Discussion

We applied our separation scheme on 250 different multi-script document images containing about 4000 text lines. We considered at least 250 text line images from each script. The images were scanned from juvenile literature, newspaper, magazine, book, money order form, computer printouts, translation books etc. Each line contains at least 10 characters. From the experiment on the above data we noted that overall accuracy of the system is about 97.52%. Distributions of the results of different script lines are shown in Table 1. From the table it can be noted that the highest accuracy is obtained for Kannada script which is about 99.37%. This is because reservoir and profile based features show distinct behaviour in Kannada script. Second highest accuracy (99.19%) obtained from Gujrathi script. Maximum errors (5.77%) came from the Gurumukhi script. Out of 5.77% errors 4.81% are made with Devnagari script. This is because of script similarity of Gurumukhi and Devnagari .

Table 1. Distributions of script line identification accuracy. (Here B =Bengali, D=Devnagari, E=English, Gu=Gurumukhi, M=Malayalam, Ta=Tamil, Te=Telugu, G=Gujrathi, Ka=Kannada, U=Urdu and O=Oriya).

Identified as→

Script	B	D	E	Gu	M	Ta	Te	G	Ka	U	O
B	96.73	2.61			0.65						
D		96.55			0.86			0.86		1.72	
E			97.2		1.4				0.7		0.7
Gu		4.81		94.23						0.96	
M		0.81	0.81		95.93				0.81		1.63
Ta					2.54	97.46					
Te							98.80		1.20		
G								99.19	0.81		
Ka									99.37	0.63	
U			1.61							98.39	
O			1.61							0.81	97.58

From the experiment we noticed that most of the errors came for short line containing fifteen or less characters. We also note that accuracy of the system increases if number of characters increases in a line. From the experiment we noted that accuracy of the system is about 99.2% if the number of characters in a line is forty or more.

This scheme does not depend on the size of characters in the text line. Also, this approach is insensitive to font, style and case variation. Also, because of the use of robust features the scheme can identify a script line if some of characters are connected (touching) in the line because of noise/poor scanning.

References

[1] B. B.Chaudhuri and U. Pal, "Skew angle detection of digitized Indian Script documents", IEEE PAMI, vol.19, pp.182-186, 1997.

[2] J. Hochberg, P. Kelly, T Thomas and L. Kerns, "Automatic script identification from document images using cluster-based templates", IEEE PAMI, vol. 19, pp. 176-181, 1997.

[3] U. Pal, A. Belaïd and Ch. Choisy "Touching numeral segmentation using water reservoir concept", Pattern Recognition Letters, vol.24, pp. 261-272, 2003.

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

[5] U. Pal and B. B. Chaudhuri, "Script line separation from Indian multi-script documents", In Proc. 5th ICDAR, pp 406-409, 1999.

[6] U. Pal and B. B. Chaudhuri , "Identification of different script lines from multi-script documents", Image and Vision computing, Vol. 20, no.13-14 pp. 945-954 2002.

[7] L. 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.