Global + Local (Glocal) Features based Script Identification System for Indian Multi-Script Documents

[1]B.V. Dhandra, [2]Mallikarjun Hangarge

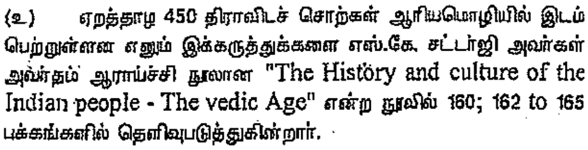
Abstract

The problem of determining the script of the text present in multi-script documents is one of the important steps as a precursor to Optical Character Recognition (OCR). In this paper, the word level script identification in bilingual or multilingual documents based on global and local features is reported. Initially, the identification of the script of words using morphological filters (global features) and regional descriptors (local features) in a bi-script scenario is considered. In the later stage, the problem is extended across tri-script to five-script scenarios. The words of different scripts are classified using K nearest neighbour algorithm with five fold cross validation on a large dataset of 27,500 word images. The proposed algorithm achieves an average accuracy of more than 94.78% and is robust for noise, word length, font styles, and sizes.

1. Introduction

An important area in the field of document image analysis is that of optical character recognition (OCR), which is broadly defined as the process of recognizing either printed or handwritten text from document images and converting it into electronic form. To date, many algorithms have been presented in the literature to perform this task for a specific language, and such OCRs will not work for a document containing more than one script. Most of the work reported in the literature relates to Roman, Arabic, Chinese, and Korean and Japanese scripts. Though, some work has already been reported involving Indian scripts, the work is still in its infant stage.

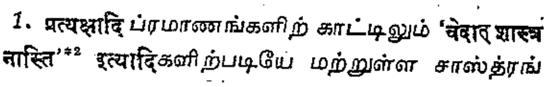
A multi-script and multi-lingual country like India, most of the official documents are multi-script in nature. Besides, there are other Asian countries where multi-script documents exist. In India, the mixing of scripts in a document may exist at paragraph or text line or word level. In India, each state has its own language/script for its official and commercial use. The people living in the border areas of the states will use bi-scripts/tri-scripts as their business languages. The document presented in Fig. 1 is an example of bi-script document. Further, under the three-language formula [11], adopted by most of the states, the document in a state may be printed in its respective official language, the national language (Hindi, uses Devnagari script) and also in English, and hence, tri-script documents co-exist. Thus, identification of the script is one of the necessary challenges for the designer of OCR systems dealing with such multi-script documents. Quite a few results have been reported in the literature, identifying the scripts in multi-script documents. However, very few of these works deals with script identification at the word level. This has motivated us to attempt the script identification problem at word level in multi-script documents.



(உ) ஏறத்தாழ 450 திராவிடச் சொற்கள் ஆரியமொழியில் இடம் பெற்றுள்ளன எனும் இக்கருத்துக்களை எஸ்.கே. சட்டர்ஜி அவர்கள் அவர்தம் ஆராய்ச்சி நூலான "The History and culture of the Indian people - The vedic Age" என்ற நூலில் 160; 162 to 165 பக்கங்களில் தெளிவுபடுத்துகின்றார்.

(a)

[1&2] P.G.Department of Studies and Research in Computer science, Gulbarga University Gulbarga, Karnataka, India
E-mail : dhandra_b_v@yahoo.co.in,mhangarge@yahoo.co.in



1. प्रत्यक्षादि ப்ரமாணங்களிற் காட்டிலும் 'वेदात् शास्त्रं परं नास्ति'*2 इत्यादिகளிற்படியே மற்றுள்ள சாஸ்த்ரங்

(b)

Figure 1, (a) and (b) are sample bi-script documents.

The work related to the script identification can be broadly classified into two categories; namely, local and global features based approaches. Local approach is an approach, which analyses a document image at the list of connected components. Global approach is an approach which employees image regions as textures. Further, the work can be subcategorized into three different categories based on the type of images considered; (1) text blocks, (2) text lines and (3) words.

1.1 Word Wise Script Identification

Peeta Basa Pati *et al.* [19] used global approach based on Gabor filter bank having three different radial frequencies and six different angles of orientation with a radial frequency bandwidth of 1 octave and an angular bandwidth of 30° They obtained a combination of 18 odd and 18 even filters with three radial frequencies and six degrees. The size of each filter mask used for experimentation is 13x13. Thus, a 36-dimensional feature vector of the total energy in each of the filtered images is used. The Linear discriminant (LD) and nearest neighbour (NN) classifiers are used to classify the word images of five different scripts namely, Roman, Devnagari, Kannada, Tamil and Oriya in bi-script, tri-script and five-script scenarios. They used prototypes to reduce the training set to smaller size and in turn saved 87% of memory and computation. However, this method assumes that a word should contain at least two characters. Thus, it is word size dependent and still it involves time complexity as it depends on 36-dimension feature vector for classification.

The other algorithms proposed for word level script identification are by Dhanya *et al.* [13] based on Gabor filters and spatial spread features, Pal *et al.* [15] based on water reservoir and conventional features and Padma *et al.* [16] based on discriminating features, have recognition rate of more than 95%. The recognition accuracy of these algorithms falls drastically for the words of size having less than three characters. Hence, the algorithms are word size dependent. Peeta Basa Pati *et al.* [17] have proposed word level script identification for Tamil, Devnagari and Oriya scripts based on 32 features using Gabor filters. They have not reported about the performance of their algorithm for various font sizes and styles. Mean while, these algorithms deal with only bi-script and tri-script identification problems. However, in Indian context, script identification in multilingual documents is to be focused, for successful design of multi-script/ multi-lingual OCR. That is what this paper is about.

1.2 Script Identification at Text block and Text line Level

The number of other approaches for automatic script identification at text blocks as well as at text lines has been proposed in the literature and is briefly presented here. Spitz [1] proposed a method for distinguishing between Asian and European languages by examining the upward concavities of connected components. Tan *et al.* [6] proposed a method based on texture analysis for automatic script and language identification from document images using multiple channel (Gabor) filters and Gray level co-occurrence matrices for seven languages: Chinese, English, Greek, Koreans, Malayalam, Persian and Russian. Hochberg, *et al.* [2,3] described a method of automatic script identification from document images using cluster-based templates and also proposed an algorithm for handwritten script identification of six scripts using statistical features extracted based on connected components. Tan [5] developed rotation invariant features extraction method for automatic script identification for six languages. Wood *et al.* [4] described projection profile method to determine Roman, Russian, Arabic, Korean and Chinese characters. A. Busch *et al.* [9] proposed a texture based script identification system using wavelet features. Pal *et al.* [11] proposed an automatic technique of separating the text lines from 12 Indian scripts. Gaurav *et al.* [12] proposed a method for identification of Indian languages by combining Gabor filter based techniques and direction distance histogram classifier for Hindi, English, Malayalam, Bengali, Telugu and Urdu. Basavaraj *et al.* [14] proposed a neural network based system for script identification of Kannada, Hindi and English. Nagabhushan *et al.* [18] discussed an intelligent pin code script identification methodology based on texture analysis using modified invariant moments. All the above methods are either based on global features or local features. In this paper an attempt is made to demonstrate the potentiality of hybridized features (i.e. combination of global and local features) for script identification at word level. The study of performance comparison of the global features versus local features is a debate [10]. Though, their performance is application dependent. In the context of word level script identification, Gabor filters are extensively used [10, 13, 17, and 16] for extraction of global features considering a word as a texture and they suffer to extract discriminating features when the word size is less than two characters. This is, because of missing essential sub patterns in a texture. Thus, they are image size dependent. However, local features are script dependent and they suffer with generalization problem. Therefore, we have hypothecated that hybridization of local and global features may give good performance rather than individual. Our hypothesis is proved to be true based on the experimental results.

In Section 2, the brief overview of data collection, preprocessing, skew correction and segmentation is presented. In Section 3, the feature extraction, features computation and K nearest neighbour classifier are discussed. The experimental details and results obtained are presented in Section 4. Conclusion is given in Section 5.

2. Data Collection and Preprocessing

2.1 Data Collection

In this paper, two data sets are used for experimentation. The first dataset of 22500 word images is accessed from Indian institute of Science, Bangalore used by [19] and it is used as a benchmark to measure the performance of our algorithm [courtesy by A. G. Ramkrishnan and Peeta Basa Pati, IISC, Bangalore]. This database has been created with an assumption that a word should contain at least two characters. The second one is, a typical data set of 5000 word images obtained by segmenting 200 document images. We have retained all the components obtained by segmentation like isolated vertical and horizontal lines, opening and closing brackets, Arabic numerals, single character words and other special characters. Fig. 2 shows few images of the second data set. Out of 200, one hundred documents are collected from various Books, Magazines and News Papers. Another 50 documents are downloaded from digital library of Indian institute of science. The remaining 50 documents are downloaded from samachar.com, then printed and finally scanned with 300 dpi. Most of these documents are bilingual in nature. These, documents contain lot of variability in terms of font size, styles and scanning resolutions varying from 300 to 600 dpi, as well as the age and nature of the document. The automatic bifurcation of second data set is carried out into one component words, two component words, three component words and more than three component words except Devnagari script. This exercise is performed to test the robustness of the proposed algorithm for word size variations. The details of the second dataset are given in Table 1.

Table 1 Second Database Details

Scripts	One comp. words	Two comp. words	Three comp. words	> 3 comp. words	No. of words
English	55	139	179	627	1000
Kannada	39	67	140	754	1000
Tamil	16	92	151	741	1000
Oriya	81	57	134	728	1000

Note: Devnagari words normally cannot be bifurcated as different component words, because sirorekha joins all the components or characters of the word and hence it yields as a single component.

2.2 Preprocessing

In general, the scanned document images are not good candidates for segmentation and feature extraction. The varying degrees of contrast in gray scale images and the presence of skew and noise will affect such features, leading to high classification error rates. In order to reduce the impact of these factors, the document images from which features are to be extracted must undergo a significant amount of preprocessing. The individual steps, which are performed in this stage, are binarization, deskewing, and segmentation. However, we assume that, the document image contains only text matter.

Binarization can be described as the process of converting a gray scale image into one, which contains only two distinct tones, that is black and white. This is an essential stage in many of the algorithms used in document analysis; especially those that identify connected components, that is, group of pixels, which are connected to form a single entity. For the purpose of this evaluation, a global thresholding approach provides an adequate means of binarization, and the method proposed by Otsu in [20] is used. The morphological area opening operation is performed to remove the noise likes, periods, commas and quotation marks of area less than are equal to 50 pixels.

2.2.1 Skew Detection and Correction

Knowing the skew of a document is necessary for many document image analysis tasks. Calculating projection profiles, for example, requires knowledge of the skew angle of the image to a high precision in order to obtain accurate results. In practicable, the exact skew angle of a document is rarely known, as scanning errors, different page layouts, or even deliberate skewing of text can result in misalignment. In order to correct this, it is necessary to accurately determine the skew angle of a document image or of a specified region of the image, and for this purpose, a number of techniques have been presented in the literature [21, 22, 23 and 24]

Dhandra et al [25] discussed an image dilation and region labeling method for skew angle detection. In order to estimate the skew angle, they dilated the input image horizontally using a line-structuring element with a length of 32 pixels to fill up the gapes between the characters and words. Further, regions are labeled and then orientation of each label has been calculated and their mean is taken as an estimate of skew angle. However, this algorithm works well for English documents and when the script of the document changes, especially for Indian scripts having isolated descenders and ascenders; it fails to estimate the skew angle accurately. Most of Indian scripts have descenders and ascenders; therefore to make this algorithm script independent, we extended it by dilating the input image in vertical and horizontal directions with a line-structuring element to fill up the gaps between characters, descenders and ascenders to produce the text lines or words as the regions. The length of the structuring element is computed using equation (1) with k=0.5 and k=0.7 for vertical and horizontal directions respectively. Then, regions are labeled and their average orientation is taken as an estimate of skew angle (,). Rotating an input image in opposite direction by , performs the skew correction.

Strel_length=K. Mean (connected components height) (1)

where, strel_length means the length of the structuring element and K varies from 0.25 to 0.8.

2.2.2 Segmentation

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.

3. Feature Extraction

The segmented word images are used to compute the eight-connected components of white pixels on the image and produce the bounding box for each of the connected components to compute the local features. The whole word image is used as a texture to compute the global features. The computation of global and local features is discussed below. The extraction of the proposed features are based on the visual discriminating factors that influences for the discrimination of the scripts by human being such as (1) vertical strokes densities (2) horizontal strokes densities (3) presence of holes, (4) the connected component or character aspect ratio, (5) the connected component or character eccentricity and extent. The computation of these features is discussed in the following.

3.1 Features Computation

3.1.1 Global Features

To extract the vertical and horizontal strokes, we have performed the opening operation on the input binary image with the line-structuring element. The length of the structuring element is computed with k=0.5 using equation (1) for both the directions.

Stroke density: The stroke length is defined as the number of pixels in a stroke as the measure of its length [7], in vertical and horizontal directions of the image. Further, the stroke density is defined as the total length of all the strokes divided by the size of the image. Throughout the discussion (section 3.1) N is referred as number of on pixels in an image. The values of 10 features extracted here, are real numbers. A line chart shown in Fig. 3 exhibits the potentiality of features for discriminating the proposed scripts.

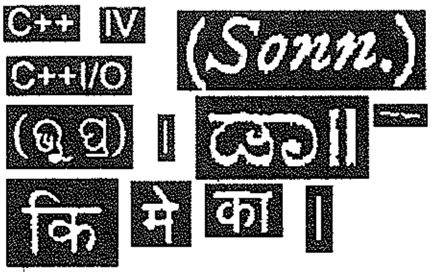


Figure 2 Sample Images of Second Data Set.

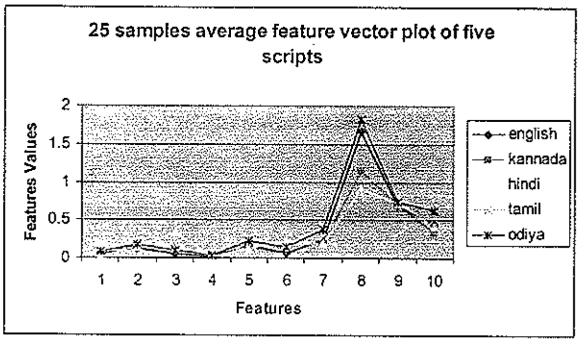


Figure 3 Average (25 samples) Feature Vector Plot of Five Scripts

1 Vertical Stroke Density (VSD):

$$VSD\ (Pattern) = \frac{\sum_{i}^{N} Onpixel(V\_Pattern_i)}{Size(V\_Pattern)} \qquad (2)$$

2 Horizontal Stroke Density (HSD):

$$HSD\ (Pattern) = \frac{\sum_{i}^{N} Onpixel(H\_Pattern_i)}{Size(H\_Pattern)} \qquad (3)$$

The remaining features 3, 4, 5 and 6 are extracted based on top hat and bottom-hat morphological filters (transformations) in vertical and horizontal directions. The features are computed in similar way as discussed in equation (2)-(3). The "top-hat" transformation, due to F.Meyer [26] aims to extract the objects that have not been eliminated by the opening. *It can be defined as the residue between the identity and an opening.* This transformation process is depicted in the Fig. 4 (from Serra lecture notes). This transformation is preferred here (to catch the residue after eliminating vertical and horizontal stokes from the image) to decompose an input image in two directions (vertical and horizontal) at two levels (top and bottom) to extract fine textural primitives for discriminating the scripts. As an illustration, morphological opening, top hat and bottom hat transformation in vertical and horizontal directions of English and Devnagari words is represented in Fig.5

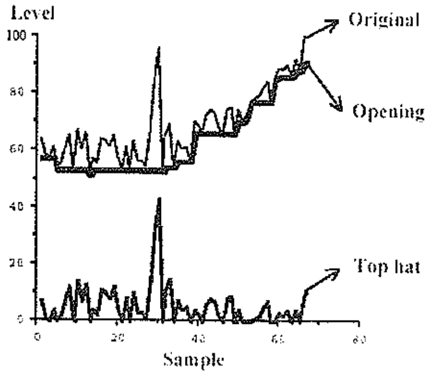


Figure 4. Top-hat Transformation Process

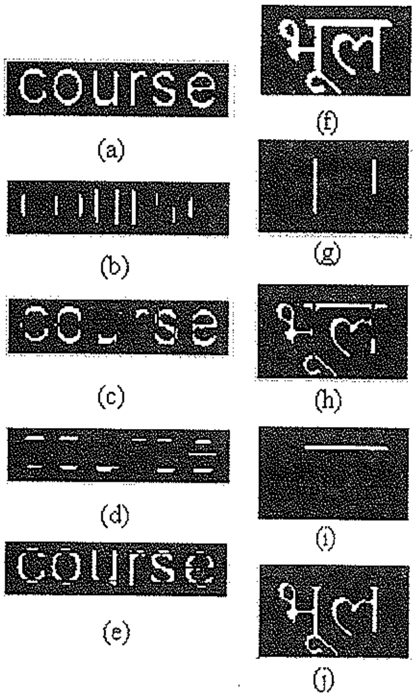


Figure 5 (a) English input word, (b) vertical opening of (a), (c) vertical top hat transform of (a), (d) horizontal opening of (a), (e) horizontal top hat transform of (a), The second column images from (f)-(j) represent the same sequence of operations as explained above with Devnagari word image.

***Pixel Density of an image after fill holes*:** This is the ratio between the numbers of on pixels left after performing fill hole operation on input pattern, to its size. 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 input word image. That is,

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

Then g = $[R_f^c\,(f_m)]^c$ has the effect of the filling the holes in f, where, R_f^c is the reconstructed image of f.

7. Pixel Density of the pattern after fill holes is defined as

$$PDH(pattern) = \frac{\sum_{i}^{N} Onpixel(g_i)}{Size(g)} \tag{4}$$

3.1.2 Local features

The extraction of following features is based on the connected components of an image and thus, they are local features.

8 ***Aspect Ratio***: - The ratio of the height to the width of a connected component of an image [3]. The average aspect ratio (AAR) is defined as

$$AAR(pattern) = \frac{1}{N}\sum_{i=1}^{N}\frac{height(component_i)}{width(component_i)} \tag{5}$$

where N is the number of connected components in an image.

The value of AAR is a real number. Note that the aspect ratio is very important feature for word wise script identification [7].

9. ***Eccentricity***: It is defined as the length of major axis divided by the length of the minor axis of a connected component of an image [8].

$$Average\ Eccentricity = \frac{\sum_{i}^{N} eccentricity_i}{N} \tag{6}$$

where N is the number of connected components in an image.

10. ***Extent***: It is a real valued function; defined as the proportion of the pixels in the bounding box that are also in the region. It can be computed as area divided by the area of the bounding box.

$$Average\ Extent = \frac{\sum_{i}^{N} extent_i}{N} \tag{7}$$

where N is the number of connected components in an image.

The sample feature vector of English, Kannada, Devnagari, Tamil and Odiya is given below.

Kannada=[0 0.1770 0.2874 0.0210 0.1560 0.3348 0.2993 1.1085 0.7103 0.3304]

English=[0.1623 0.0629 0.0337 0.0459 0.1794 0.0536 0.2967 1.7838 0.6895 0.4041]

Hindi=[0.0438 0.1368 0.1547 0.0479 0.1327 0.1672 0.2163 0.8789 0.7185 0.4436]

Tamil=[0.0384 0.1670 0.1777 0.0134 0.1920 0.3254 0.3768 0.8225 0.7384 0.4064]

Odiya=[0.0299 0.1782 0.0220 0.0225 0.1856 0.0434 0.3904 1.7517 0.7333 0.6297]

3.2 K-Nearest neighbour Classifier

In order to identify the most appropriate classifier for the problem of script identification, we chosen the KNN classifier based on its best performance reported in [10] by comparing with well known classifiers, namely, Parzen density, quadratic Byes, feed-forward neural net and support vector machine. For selection of KNN classifier, we followed [10], because of the underlying problem remains same. The 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. To test the performance of the classifier the feature set of 27,500 word images are randomly divided (approximately equal) into five groups and a 5-fold cross validation was done for 10 iterations to get optimum result as reported in Table-2 to 7.

4. Results and Discussions

For experimentation, two data sets are used. First data set consists of 22,500 word images and second data set (typical data set) consists of 5,000 word images. The typical data set consists of one, two, three and more than three components (or character) words, along with some isolated special characters, whereas first dataset involves the word images of two characters and more than two characters. To test the generality of the proposed work, we applied it on both the data set and noticed its better performance. The script identification results for first data set is shown in Table 2,3and 4. Pati *et. al* [19], claimed the highest average recognition accuracy of 99.7%, 99.0% and 96.0% for bi-script, tri-script and five-script recognition with 32 features, whereas the proposed algorithm's maximum average recognition accuracy is 99.92%, 99.6% and 98.03% for bi-class, tri-class and five-class problems with 10 features (i.e. 1/3rd of the features used by Pati). In view of this, the proposed algorithm is efficient, robust and has showed better performance in terms of accuracy and dimensionality. Furthermore, the dimensionality reduction analysis is performed based on principal component analysis with covariance to identify the dominant features of the proposed algorithm. It is noticed that the aspect ratio is the more dominant feature for recognition among the other features. The results of PCA evidences the argument of A.K.Jain [7], that aspect ratio is an important feature for word wise script identification. We also experimentally verified the contribution of each feature for script identification and it is shown in Fig 7. As a second step verification, the algorithm is applied on the second data set containing single character words, scanning resolution artifacts and noise like periods, opening brackets, closing brackets and long hyphens, which have an area of more than 50 pixels. The average recognition results of this data set are shown in Table 5, 6 and 7. The average recognition result for bi-class problem is as high as 99.30% and as low as 96.65%. For tri-script the maximum average accuracy is 97.6% and minimum is 96.86% and the average recognition accuracy for five-script is 94.78%. The average recognition rate of second data set is less to the maximum extent of 2% as compared to first data set; this is due to the noise components like isolated horizontal/ vertical strokes, periods and commas constituting an area of more than 50 pixels. It is observed that, the existence of isolated vertical stokes in Odiya and Devnagari scripts are more and long hyphens used between the words also exist in number of documents used for experimentation. Further, the influence of the word size on the recognition rate is also observed experimentally and it is depicted in Fig. 6. It is clear that as the word size increases the recognition rate also increases. Finally, we argue that the overall performance of the proposed algorithm for script identification at word level is better than the proposed algorithms in the literature.

Table 2 Average script identification results of first data set for bi-script, in percentage using K-NN classifier with k=3

Scr.	Eng	Kan	Hin	Tam	Odi
Eng	-----	99.61	99.78	99.84	99.66
Kan	99.61	-----	97.43	99.64	99.92
Hin	99.78	97.43	------	99.38	99.92
Tam	99.84	99.64	99.38	-----	99.21
Ori	99.66	99.92	99.92	99.21	-----

where Eng, Kan, Hin. Tam and Odi stands for English, Kannada, Hindi, Tamil and Odiya scripts. Similarly scr. and avg. are for script and average.

Table 3 Average recognition accuracies of first data set in percentage for tri-script of Kannada, Odiya and Tamil with Roman and Devnagari using KNN with k=3.

Scr.	Kan.	Odi.	Tam.
Roman	99.5778	99.2222	99.6667
Devnagari	97.7333	99.8444	99.1556
Local	96.3778	99.7556	99.2889
Average	97.8963	99.6074	99.3704

Table 4 Script identification results of first data set for five-script scenario, in percentage

Scr.	Eng.	Kan.	Hin.	Tam.	Odi.	Avg.
KNN with k=3	98.80	96.11	97.46	98.75	99.02	**98.03**

Table 5 Average script identification results of second data for bi-script, in percentage using K-NN with k=3

Scr.	Eng.	Kan.	Hin.	Tam.	Odi.
Eng.	-----	97.15	99.30	99.20	98.45
Kan.	97.15	-----	99.25	98.10	98.45
Hin.	99.30	99.25	-----	96.65	98.10
Tam.	99.20	98.10	96.65	-----	98.45
Odi.	98.45	98.45	98.10	98.45	-----

Table 6 Average recognition accuracies of second data set in percentage for tri-script of Kannada, Odiya and Tamil with Roman and Devnagari using KNN with k=3.

Scr.	Kan.	Odi.	Tam.
Roman	96.8000	97.5000	99.7000
Devnagari	97.7000	96.9000	94.5000
Local	96.5000	98.6000	96.4000
Average	97.0000	97.6666	96.8666

Table 7 Script Identification results of second data set, in percentage for five-script scenario, using KNN classifier

Scr.	Eng.	Kan.	Hin.	Tam.	Odi.	Avg.
KNN with k=3	95.20	95.60	92.70	94.20	96.20	94.78

5. Conclusion

In this paper, we have proved experimentally our hypothesis that the hybridization of global and local features will perform better than individual. The aim of this paper is to facilitate the designing of successful multilingual OCR. This is very simple algorithm and it works by decomposing an input image in two directions (horizontal and vertical) at two levels (top and bottom transform) to extract (global features) fine texture primitives for discrimination of scripts by considering a word as texture. The local features like aspect ratio, eccentricity and extent have much influence on the performance of the algorithm. During the extraction of features, the connected components of size less than are equal to 50 pixels are removed from the image prior to features computation. Thus, the approach is robust with respect to noise. It is also clear that the algorithm is insensitive to font styles, sizes, word length and scanning artifacts like resolution. Experimental results have showed that relatively simple technique can reach a high level accuracy for discriminating the proposed scripts. It is our future endeavor to modify this algorithm to perform script identification from multilingual document images containing more number of Indian languages. In future, we have a goal to prove this algorithm as a generalized framework for script identification at word level for Indian scripts.

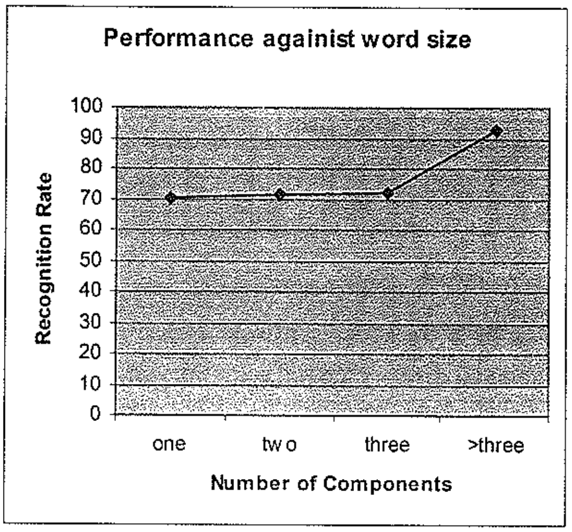


Figure 6 Performance Versus Word Size

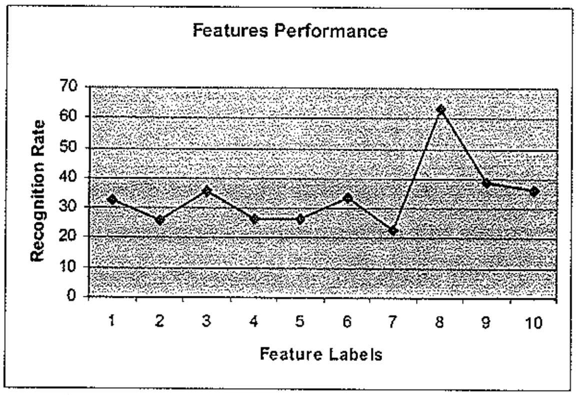


Figure 7 Performance Versus Features

6. Acknowledgment

The authors are grateful to Dr. P. Nagabushan for his valuable guidance, suggestions, critical analysis and comments through out this work. We also thankful to Dr. G Hemantha Kumar and Dr. D. S. Guru, Dept. of Computer Science, University of Mysore, for their helpful discussion and encouragement during this work.

References

[1] A.L.Spitz, *"Determination of the script and language content of document images", IEEE* Tran. on Pattern Analysis and Machine Intelligence, Vol. 19, pp 234-245, 1997.

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

[3] Judith Hochberg, Kevin Bowers, Michael Cannon and Patrick Keely, *"Script and language identification for hand-written document images",* vol.2, pp 45-52, *IJDAR-1999.*

[4] S. Wood, X. Yao, K.Krishnamurthi and L.Dang, *"Language identification from for printed text independent of segmentation",* Proc. of Int'l. Conf. on Image Processing, pp 428-431, 1995.

[5] T.N.Tan, *"Rotation invariant texture features and their use in automatic script identification", IEEE* Trans. on Pattern Analysis and Machine Intelligence, vol. 20, pp 751-756, 1998.

[6] G.S.Peake and Tan, *"Script and language identification from document images",* Proc. of Eighth British Mach. Vision Conf., vol.2, pp 230-233, Sept-1997.

[7] Annop M. Namboodri, Anil K Jain, *" Online handwritten script identification",* IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 26,no.1,pp 124-130, 2004.

[8] Dengsheng Zhang, Guojun Lu, *"Review of shape representation and description techniques",* Pattern Recognition, vol. 37, pp 1-19, 2004.

[9] A. Busch ,W. W. Boles and S. Sridharan, *" Texture for script identification",* IEEE Trans. on Pattern Analysis and Machine Intelligence, 27(11) 1720-173, 2005.

[10] Gopal Datt Joshi, Saurabh Garg and Jayanthi Sivaswamy, *"Script Identification for Indian Documents"*, In. Pro. of 7th IAPR workshop on Document Image Systems (DAS), pp 255-267, New Zealand, 2006.

[11] U.Pal and B.B.Chaudhuri, *"Script Line separation from Indian Multi- script Documents"*, In Proc. of 5th ICDAR, pp 406-409, 1999.

[12] Santanu Chaudhury, Gaurav Harit, Shekar Madnani, R.B.Shet, *"Identification of scripts of Indian languages by Combining trainable classifiers", Proc. of ICVGIP* 2000, Dec-20-22, Bangalore, India.

[13] D Dhanya, A.G Ramakrishnan and Peeta Basa pati, *"Script identification in printed bilingual documents"*, Sadhana, vol. 27, part-1, pp 73-82, 2002.

[14] S.Basavaraj, Patil and N.V.Subbareddy, *"Neural network based system for script identification in Indian documents"*, Sadhana, vol. 27, part-1, pp 83-97, 2002.

[15] U.Pal. S.Sinha and B.B Chaudhuri, *"Word-wise Script identification from a document containing English, Devnagari and Telgu Text," Proc. of NCDAR*, PP 213-220, 2003.

[16] M.C.Padma and P. Nagabhushan, *"Identification and separation of text words of Kannada Hindi and English languages through discriminating features*", *Proc. of NCDAR-2003*, pp 252-260. 2003.

[17] Peeta Basa pati, S. Sabari Raju, Nishikanta Pati and A.G. Ramakrishnan, *"Gabor filters for document analysis in Indian Bilingual Documents", Proc. of ICISIP,* pp 123-126, 2004.

[18] P. Nagabhushan, S.A. Angadi and B.S. Anami, *"An Intelligent Pin code Script Identification Methodology Based on Texture Analysis using Modified Invariant Moments", Proc. of ICCR-2005,* pp 615-623.

[19] Peeta Basa Pati and A.G.Ramakrishnan, *"HVS inspired system for Script Identification in Indian Multi-Script Documents"*, In Proc. of 7th International Workshop on Document Analysis System, Nelson Newland, Feb-13-15, pp 380-389, 2006.

[20] N. Otsu, *"A Threshold Selection Method from Gray-Level Histogram*", IEEE Trans. Systems, Man, and Cybernetics, vol.9,no.1, pp 62-66,1979.

[21] H.S. Baird, *"The Skew Angle of Printed Documents, "Document Image Analysis"*, L. O : Gorman and R .Kasturi, eds., IEEE CS Press, pp 204-208,1995.

[22] B.B.Chaudhuri and U.Pal, *"Skew Angle Detection of Digitized Indian Script Documents"*, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no.2, pp 703-712, Feb.1997.

[23] W. Postl, *"Detection of Linear Oblique Structures and Skew Scan in Digitized Documents"*, In Proc. of Int'l Conf. Pattern Recognition, pp. 687-689,1986.

[24] G.Peak and T Tan, *"A General Algorithm for Document skew Angle Estimation"*, Proc. Int'l. Conf. Image Processing, vol.2, pp 230-233, 1997.

[25] B.V.Dhandra, V.S.Malemath, Mallikarjun Hangarge, Ravindra Hegadi, "*Skew detection in Binary image documents based on Image Dilation and Region labeling Approach*", In Proceedings of ICPR 2006, V. No. II-3, pp 954-957, 2006.

[26] F.Meyer, *"Contrast Feature Extraction"*, In J-L Chermant, editor, Quantitative Analysis of Microstructures in Material Science, Biology and Medicine, Riederer Verlag, Stuttgart, Germany,1978. Special Issue of Practical Metalography.

Author's Biography

B.V.Dhandra. Professor and Chairman, P.G Department of Studies and Research in Computer Science, Gulbarga University, Gulbarga, Karnataka, INDIA. He has born in Gulbarga, India on January 1st 1955. He has received MA degree in Statistics in 1979 and M.Phil in Statistics in 1986 from Karnataka University Dharwad, Karnataka, India. He obtained his Ph.D. degree from Shivaji University, Kolhapur, India in the year 1993. He served as lecturer during 1979 to 1993, as a Reader during 1993 to 2001 and since 2001 he has been appointed as Professor in Computer Science and presently heading the PG Department of Studies and Research in Computer Science, Gulbarga University, Gulbarga, India He is member of board of studies of various Universities. He has published more than 40 research articles in peer reviewed national and international conferences and journals. His research interests are Pattern Recognition, Image Processing and Operations Research.

Mallikarjun Hangarge is a senior faculty member and Head of the Department of Computer Science, Karnataka Arts, Science and Commerce Degree College, Bidar, Karnataka, India. He has born in Gulbarga, India, on January 1st 1966. He received Msc. degree in Statistics from Gulbarga University, in 1989, India. He received Post Graduation Diploma in Computer Application in 1992 from Gulbarga University, India. He has been awarded MPhil in Computer Science in 2003 from M.S.University, Trinnelvelli, Tamil Nadu, India. He has 15 years of experience in teaching for under graduate and postgraduate students. Presently, he is carrying out his PhD under the faculty improvement program of University Grant Commission, New Delhi, India. He has published more than 20 research articles in peer reviewed National and International conferences and Journals and as part of his PhD work he has been published 10 research articles. His research interests are Pattern Recognition and Image Processing.