Identification of Scripts of Indian Languages by Combining Trainable Classifiers

Santanu Chaudhury santanuc@ee.iitd.ernet.in
Gaurav Harit g_harit@hotmail.com
Shekhar Madnani s_madnani@hotmail.com
R.B. Shet rbshet@hotmail.com

Department of Electrical Engineering
Indian Institute of Technology, Delhi
Hauz Khas, New Delhi 110016
India

Abstract

Identification of the script in an image of a document page is of primary importance for a system processing multi-lingual documents. In this paper two trainable classification schemes have been proposed for identification of Indian scripts. Both the schemes use connected components extracted from the textual region. The first classifier uses the novel Gabor filter-based feature extraction scheme for the connected components. We have also found that the pixel distribution of connected components can be used to capture the shapes of connected components and thus form the basis for script recognition. It has been experimentally found that the features extracted by Gabor filter-based scheme provides the most reliable performance. The other technique is simple, computationally more efficient and gives reasonably good performance. The decisions of the two classifiers designed in this paper are combined using Logistic Regression method. The combination has shown an improved recognition performance as compared to that by individual classifiers.

1 Introduction

Identification of the script in a document text page image is of primary importance for a system processing multi-lingual documents. Documents can be classified on the basis of the scripts and further, knowing the script of the text in the page image, the script specific OCR system can be applied to extract textual information. The basic problem involved here is that of classification of the textual regions into script categories using image based features.

Among the earlier works in this area, Spitz[1] has proposed a technique for distinguishing Han and Latin based scripts on the basis of spatial relationships of features related to the character structures. Language identification within the Han script class (Chinese, Japanese, Korean) is performed by the analysis of the distribution of optical density in text images. Latin-based languages are identified using a technique based on character shape codes, and finding the frequency of language specific patterns. Hochberg et al.[2] have described a technique for identifying 13 scripts. Their method is based on creation of script specific templates by a technique of clustering textual symbols from a training set. A textual symbol is defined as a connected component meeting certain size requirements. All the textual symbols are normalized to a pre-specified size. Symbols from new images are compared to the templates to find the best script. Wood et al.[3] have used projection profiles for script separation. Ding et al.[4] have also proposed a method for separating European and Oriental scripts. Pal and Chaudhuri[5] have proposed a decision tree based strategy for separating English, Urdu, Bengali and Devanagari. They have used projection profile, statistical, topological and stroke based features. However, their strategy requires reliable segmentation of the textual region into lines and words into characters. These are difficult problems for Devnagari based scripts because of the presence of the head line, and ascenders and descenders. The template-based clustering scheme suggested in [2] needs to deal with a large number of clusters (108 to 804 per language). In order to overcome this difficulty, we have formulated two schemes based on connected component which do not use shape templates but characterize connected components in terms of other shape-based characteristics. Our first scheme uses a new approach for Gabor filter-based feature extraction to characterize connected components in terms of the constituent primitives. In the other scheme we have made use of the shapes of the the connected components by taking into

account the distribution of black pixels around every pixel of a connected component.

In this work we have considered document images in portrait orientation with almost zero skew. We have experimented with document images of newspapers and magazines scanned at 200 dpi. Samples of various font sizes and styles were considered for testing individual classifiers.

2 Gabor Filter based Recognition

Language classification based on Gabor filter-based feature extraction has been done in [6] and [11]. In [6], a 16-channel Gabor filter was used to classify each pixel as belonging to Chinese or English after suitable training of the neural networks with the outputs of the filter. This approach advocated use of a number of filters with a set of fixed frequencies and direction sensitivities. However, this strategy would require a large number of Gabor frequency channels and mask sizes to accommodate various font sizes. The Gabor filter based approach we have used is local and adaptive and so is not affected by changes in font size, line and word spacing (unlike [12]).

It can be seen that an alphabet in a script is characterized by a particular organization of line segments and curve segments. A script is distinguished by the nature of the organizational pattern of the primitives. These patterns occur at spatial frequencies which are typical to the language/script. Our Gabor filter-based feature detection scheme is motivated by this observation.

The transfer function of the circular Gabor filter used is given by

$$g(x,y) = af^2 e^{\frac{-4\,ln(2)\,f^2\,(x^2+y^2)}{\omega^2}}\, cos[2\pi f(x\cos\theta + y\sin\theta)+\phi]$$

where,

a=$\frac{4\,ln(2)}{\omega^2\,\pi}$, here a is constant for fixed ω

ω, is the number of cycles of sinusoid contained within the half height width of the Gaussian envelope

f is the spatial frequency of the sinusoid in cycles per image width

θ is the orientation of the sinusoid

ϕ is the phase of the sinusoid

An elemental entity for a script is defined as a connected component in the text image of the script. It is assumed that majority of the characters of a text body correspond to the same point size. The font may vary. A six channel Gabor filter bank is used. The directions chosen are 0, 30, 60, 90, 120, 150-degrees. The frequency is made equal to two cycles over the average height of the connected components in the text block of a script i.e. spatial period of the Gabor function is set equal to half of the average alphabet height. This allows for a constant relative frequency analysis of all font sizes. The Gabor function chosen allows one-and-a-half sinusoidal cycles to be placed within the half height of the Gaussian envelop. The mask size is made equal to the average height of the connected components in the text body.

The bounding box of a connected component is convolved with each of the six Gabor filters. The output response of the filters is full-wave rectified. The average value and the standard deviation of the response of each filter for the bounding box of the connected component are used as feature values for the connected component. Each connected component is characterized by a 12-dimensional feature vector corresponding to Gabor filter channels.

It has been found that English and Malayalam respond to the Gabor filters in a similar fashion although these two scripts are visually dissimilar (see figure 1). Samples of both the scripts respond sharply to the zero degree Gabor filter. This is because a large number of characters of these scripts are composed of vertical straight segments. English and the other languages also show a peak at the coefficient corresponding to the 90-degree Gabor filter. This is due to the horizontal line segments present in nearly all scripts. Hindi or Devanagari based scripts have a header line in most of its characters, while English has alphabets which are having horizontal segments in either the top, middle or base of the character. Telugu or Karnatic scripts which are mostly circular have nearly equal response to all the Gabor filter directions.

In general, the bounding box area of connected components for the Devanagari scripts is large (because connected components do not correspond to a character because of the top horizontal bar). Hence, the mean response per pixel will be lower than that for the other more compact alphabetic scripts having similar response to a particular Gabor filter channel. English has a number of narrow characters like 'i','l','t','j',etc. which gives rise to a higher per pixel mean value for the response of a Gabor filter channel. Hence, the size dependent differences in the connected components of a script are also captured by this technique. Therefore, the feature vector proposed here characterizes the structure of the script and also provides script specific scaling information.

A labelled feature space is formed using these feature vectors for all the connected components in the training set. A query sample is analyzed for each of its connected components. Features for each of these

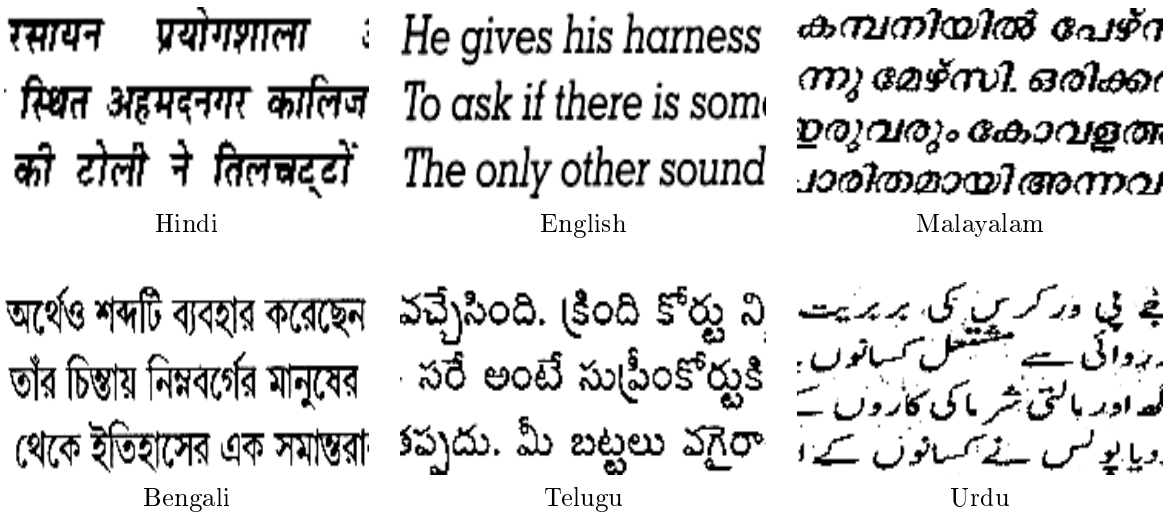

Figure 1: Script samples

components are extracted using a similar analysis. Classification is performed by computing the Mahalanobis distance with each of the script cluster.

Mahalanobis distance D_L of the unknown pattern Q from class L is given as

$$D_L = (Q - C_L)^T \wedge^{-1} (Q - C_L)$$

where,

C_L is the centroid of class L

$\wedge$ is the Covariance matrix for all the training samples.

The sample and training set contained images at 200dpi. During computation of the average height of the connected components, very small elements (connected components of less than 9-pixels) are rejected. This is to reject dots and other non script-specific markings. Also only those connected components are analysed whose height is greater than two-thirds of the average height and less than one-and-a-half times the average height. This ensures that we consider only typical connected components and at the same time we can avoid special graphical symbols or caligraphic characters embedded in the text .

For categorization of a text region, we have considered the class label of the connected components of the textual block. A textual block is classified on the basis of the majority label of the connected components.

3 Direction-distance histogram classifier

This classifier uses features extracted from the shapes of the connected components in the script image. For every pixel in the connected component, we fix up some directions in the first two quadrants with pixel at the origin.

Figure 2: Diagram showing directions and check points used around a pixel

As shown in the figure 2, the lines chosen in the first quadrant were the following : $x = 0$, $y = 0$, $3x = y$, $3y = x$, $2x = y$, $2y = x$, $3x = 2y$, $2x = 3y$, $x = y$. Lines in the second quadrant were mirror images of these lines. To avoid symmetry, the directions in the third and the fourth quadrant were not considered. In each such direction we move along and check for existence of black pixels at certain distances from the origin but within the bounding box of that connected component. These "check points" are shown as small dots in figure 2.

Since the connected components are thick, the initial point is taken slightly far off the origin and the subsequent check points follow closely. We construct a *direction-distance histogram* as shown in figure 3, which tells how many times a black pixel has been found at a particular distance at a particular direction from the origin with the origin being moved along all the pixels of the connected component.

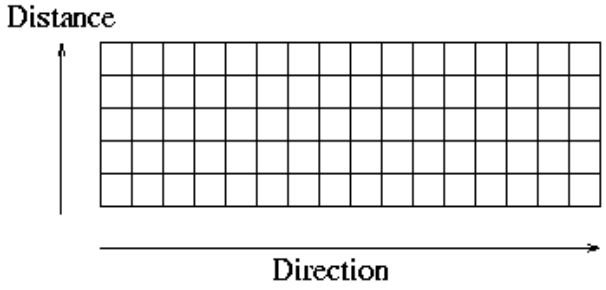

Figure 3: Two dimensional direction-distance array

We then construct a global histogram by adding up the histograms of all the connected components in the document image. This *global direction-distance histogram* is a two dimensional histogram which is then mapped to a single dimensional histogram, by traversing along the columns in a sequence, which constitutes our feature vector. This histogram carries information about the shapes of the connected components. The histogram is then normalised so that it represents a probability distribution function.

The classifier was trained using 150 samples of document images and all the feature vectors were stored with their respective labels. The script of an unknown sample is recognized by extracting its feature vector and finding its Mahalanobis distance (described in the previous section) with the stored features of each class. The class which gives the minimum distance is declared as the true of the query sample.

4 Combination Scheme Used

The previously discussed classification schemes are following different methodologies and thus may be having some advantages and disadvantages. Keeping this in mind it looks logical to combine these classifiers in some meaningful way to get better classification capability.

Previous methods for combining classifiers include intersection of decision regions [7], voting methods [8], prediction by top choice combinations[9] and use of Dempster-Shafer theory [10]. We have used the method of *Logistic Regression* [7] to combine the above two classifiers. This method tries to exploit the consistencies of the individual classifiers. For example, if a classifier is consistently misclassifying English as Malayalam, this fact an be made use of while computing the combined decision of the classifiers.

In this paper, we have considered the two classifiers described in the previous sections for combination. Six language scripts *Hindi, English, Telugu, Malayalam, Bengali, Urdu* are considered for classification. A training set consisting of 25 samples of each script is used for designing classifiers. For Logistic Regression analysis a test sample set consisting of 40 patterns of each script is obtained.

For a given test sample, the Gabor filter based classifier awards ranks to each language based on the number of connected components of the test sample whose features are found nearest to those of that language (class). The language having the maximum number of connected components near to its features is awarded the highest rank. The direction-distance histogram classifier gives the highest rank to the language whose features have the shortest Mahalonobis distance to those of the test sample. The two classifiers will award a rank score $x_i = \{x_{1i}, x_{2i}\}$ to every language (class) C.

The probability $\Pi_c(x_i)$, that the test sample belongs to class C, given the rank score x_i to the class C, is estimated statistically using the formula :

$$\Pi_c(x_i) = \frac{\textit{Number of times any true class gets score } x_i}{\textit{Normalizing constant}}$$

For all possible rank scores $x_i = \{x_{1i}, x_{2i}\}$, the following sets of linear equations are formed,

$$\log \frac{\Pi_c(x_i)}{1 - \Pi_c(x_i)} = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i}$$

where α, β_1, β_2 are constant parameters. The term $\log \frac{\Pi_c(x_i)}{1-\Pi_c(x_i)} = L_c(x_i)$ is called as logit and is linearly related to x_{1i}, x_{2i}.

x_{1i} and x_{2i} range over all possible ranks awarded by the two classifiers. The number of such equations depends on the number of ranks awarded by the classifiers. For example, if the classifiers award ranks such as 0, 1, 2, 3 then there will be 16 such linear equations. These equation are solved using the *Singular Value Decomposition* method to obtain the values of the constant parameters α, β_1, β_2. These parameters are actually the weights assigned to the marks awarded by each classifier. Now, to classify a test sample, we obtain the rank scores awarded to each language C by the two classifiers. Using these rank scores and the constants, we compute the logit values L_c and Π_c for each language. The language having the highest value of Π_c is declared as the true class of the document.

5 Results

About 50 samples of each of the languages have been used for our experiments. In the absence of a standard database, we have selected samples from

Language	Hindi(%)	Telugu(%)	English(%)	Malayalam(%)	Bengali(%)	Urdu(%)
Hindi	90.0	2.0	2.0	2.0	4.0	0.0
Telugu	0.0	90.0	2.0	0.0	8.0	0.0
English	2.0	4.0	80.0	4.0	8.0	2.0
Malayalam	4.0	4.0	16.0	52.0	20.0	4.0
Bengali	2.0	0.0	2.0	4.0	92.0	0.0
Urdu	0.0	0.0	0.0	0.0	2.0	98.0

Table 1: Classification results with Gabor filter based classifier

Language	Hindi(%)	Telugu(%)	English(%)	Malayalam(%)	Bengali(%)	Urdu(%)
Hindi	66.0	8.0	0.0	8.0	18.0	0.0
Telugu	14.0	56.0	4.0	20.0	4.0	2.0
English	4.0	4.0	66.0	10.0	16.0	0.0
Malayalam	6.0	10.0	12.0	54.0	18.0	0.0
Bengali	22.0	10.0	0.0	6.0	62.0	0.0
Urdu	8.0	4.0	0.0	0.0	0.0	88.0

Table 2: Classification results for Direction-distance histogram based classifier

published newspapers and magazines. These documents have been processed for noise removal and segmentation. Some sample text elements are shown in figure 1.

5.1 Gabor filter based recognition

While building the training set we have avoided degraded samples which can produce improper connected components. We have used 25 samples of each language for building up the training set. The training set contains test samples of different font sizes and styles.

In table-1, we present classification results for document image samples of different languages.

From table-1, it is clear that for the complete test set, the Hindi documents have maximum of correct identification and Malayalam the least. We also find reasonably large mis-classification among English and Malayalam connected components. Malayalam characters also have curved shape similar to Telugu and this characteristic is also observed in the classification results of the connected components.

5.2 Direction-distance histogram based recognition

The training set as well as the testing set for this classifier were the same as that used for the Gabor filter based classifier. The classification results for the various languages have been shown in table-2. This classifier has shown improved results in the case of Malayalam. Although the performance of this classifier was not impressive, this was combined with the Gabor filter based classifier to exploit consistent misclassifications for improving the overall accuracy using two different kinds of features.

5.3 Classification using the combination scheme

The results after the combination of classifiers are shown in table-3. We see improvements in the case of English and Malayalam. However the performance for Telugu and Bengali have suffered because of the inconsistent performance of the Direction-distance histogram classifier. If English and Malayalam are not the major concern, then the Gabor filter is the obvious choice. If classification of English and Malayalam is critical for an application, then the combination scheme turns out to be the preferred choice over Gabor filter based classification scheme because of acceptable performance for majority of the language classes. In particular, performance of the combination scheme for English has been found to be useful because in the multi-lingual environment in India, English documents are more common than those of other languages.

6 Conclusions

In this paper we have presented two trainable strategies for script recognition. Although we have worked with Indian scripts, the basic approaches are general and can be applied to other languages. Performance of these schemes is dependent on the nature of the training sets chosen. By making choice of appropriate training set, these schemes can be adapted for suitable applications. Of the two strategies proposed we have found the Gabor filter based scheme

Language	Hindi(%)	Telugu(%)	English(%)	Malayalam(%)	Bengali(%)	Urdu(%)
Hindi	90.0	0.0	0.0	4.0	6.0	0.0
Telugu	2.0	76.0	2.0	10.0	6.0	4.0
English	0.0	4.0	88.0	2.0	4.0	2.0
Malayalam	12.0	2.0	16.0	60.0	10.0	0.0
Bengali	8.0	0.0	2.0	4.0	86.0	0.0
Urdu	0.0	0.0	0.0	0.0	2.0	98.0

Table 3: Classification results after combining classifiers

to be the most promising and reliable. It can also work for regions where text and graphics are intermingled (eg. advertisements). The direction-distance histogram based scheme can work only for text regions. But, this technique is computationally less expensive than the Gabor filter based approach. The combination scheme of the above two classifiers using Logistic Regression method has shown improvements in the case of English and Malayalam.

References

[1] A.Spitz, *Determination of the Script and Language Content of Document Images*, IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol.19, No.3, pp. 235-245, 1997.

[2] J. Hochberg, L. Kerns, P. Kelly and T. Thomas, *Automatic Script Identification From Document Images Using Cluster-Based Templates*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.2, pp. 176-181, 1997.

[3] S. Wood, X. Yao, K. Krishnamurthy and L. Dang, *Language Identification for the printed text independent of segmentation*, In Proc. Int'l Conf. on Image Processing, pp. 428-431, 1995.

[4] J. Ding, L. Lam and C.Y. Suen, *Classification of oriental and European Scripts by using Characteristic features*, In Proc. 4th Int'l Conf. Document Analysis and Recognition, pp. 1023-1027, 1997.

[5] U.Pal and B.B. Chaudhuri, *Automatic Separation of Different Script Lines from Indian Multi-script Documents*, In ICVGIP 1998, pp. 141-146.

[6] Anil K. Jain and Yu Zhong, *Page Segmentation Using Texture Analysis*, Pattern Recognition, Vol. 29, No.5, pp. 743-770, 1996.

[7] Tin Kam Ho, Jonathan J. Hull, Sargur N.Srihari *Decision combination in multiple classifier systems*, IEEE transactions on Pattern Analysis and Machine Intelligence. pp. 66-7, Vol. 16, No.1 Jan 1994.

[8] Lei Xu, Adam Krzyzak and Ching Y. Suen, *Methods of combining multiple classifiers and their application to handwriting recognition*, IEEE transactions on Systems, Man, and Cybernatics Vol. 22 No. 3, June 1992 pp. 418-435.

[9] Mehdi Mostaghimi, *Bayesian estimation of a decision using Information theory*, IEEE transactions on Systems, Man, and Cybernatics Vol. 27 No. 4, July 1997 pp. 506-517.

[10] Josef Kittler, Mohamad Hatef, Robert P.W.Duin and Jiri Matas, *On combining classifiers*, IEEE transactions on Pattern Analysis and Machine Intelligence. pp. 226-239, Vol. 20, No.3 March 1998.

[11] Santanu Choudhury, Ravinder Shet, *Trainable script identification strategies of Indian Languages*, ICDAR 99, Bangalore, India pp. 657-660

[12] T.N Tan, *Rotation invariant texture features and their use in Automatic script identification*, IEEE transactions on Pattern Analysis and Machine Intelligence. pp. 751-756, Vol. 20, No.7 1998.