Wavelet Packet Based Texture Features for Automatic Script Identification

M.C. Padma padmapes@gmail.com
Dept. of Computer Science & Engineeriing,
PES College of Engineering,
Mandya-571401, Karnataka, India

P. A. Vijaya pavmkv@gmail.com
Dept. of E. & C. Engineering,
Malnad College of Engineering,
Hassan-573201, Karnataka, India

Abstract

In a multi script environment, an archive of documents printed in different scripts is in practice. For automatic processing of such documents through Optical Character Recognition (OCR), it is necessary to identify the script type of the document. In this paper, a novel texture-based approach is presented to identify the script type of the collection of documents printed in ten Indian scripts - Bangla, Devanagari, Roman (English), Gujarati, Malayalam, Oriya, Tamil, Telugu, Kannada and Urdu. The document images are decomposed through the Wavelet Packet Decomposition using the Haar basis function up to level two. Gray level co-occurrence matrix is constructed for the coefficient sub bands of the wavelet transform. The Haralick texture features are extracted from the co-occurrence matrix and then used in the identification of the script of a machine printed document. Experimentation conducted involved 3000 text images for learning and 2500 text images for testing. Script classification performance is analyzed using the K-nearest neighbor classifier. The average success rate is found to be 98.24%.

Keywords: Document Processing, Wavelet Packet Transform, Feature Extraction, Script Idenmtification.

1. INTRODUCTION

The progress of information technology and the wide reach of the Internet are drastically changing all fields of activity in modern days. As a result, a very large number of people would be required to interact more frequently with computer systems. To make the man–machine interaction more effective in such situations, it is desirable to have systems capable of handling inputs in the form of printed documents. If the computers have to efficiently process the scanned images of printed documents, the techniques need to be more sophisticated. Even though computers are used widely in almost all the fields, undoubtedly paper documents occupy a very

important place for a longer period. Also, a large proportion of all kinds of business writing communication exist in physical form for various purposes. For example, to fax a document, to produce a document in the court, etc. Therefore, software to automatically extract, analyze and store information from the existing paper form is very much needed for preservation and access whenever necessary. All these processes fall under the category of document image analysis, which has received significance as a major research problem in the modern days.

Script identification is an important problem in the field of document image processing, with its applications to sort document images, as pre processor to select specific OCRs, to search online archives of document images for those containing a particular language, to design a multi-script OCR system and to enable automatic text retrieval based on script type of the underlying document. Automatic script identification has been a challenging research problem in a multilingual environment over the last few years. All existing works on automatic language identification are classified into either local approach or global approach. Ample work has been reported in literature using local approaches [1, 3, 7-10]. The local features are extracted from the water reservoir principle [1, 3], morphological features [8], profile, cavities, corner points, end point connectivity [13], top and bottom profile based features [11, 12]. In local approaches, the features are extracted from a list of connected components such as line, word and character, which are obtained only after segmenting the underlying document image. So, the success rate of classification depends on the effectiveness of the pre-processing steps namely, accurate Line, Word and Character segmentation. It sounds paradoxical as LWC segmentation can be better performed, only when the script class of the document is known. Even when the script classes are known from the training data, testing requires the performance of LWC segmentation prior to script identification. But, it is difficult to find a common segmentation method that best suits for all the script classes. Due to this limitation, local approaches cannot meet the criterion as a generalized scheme.

In contrast, global approaches employ analysis of regions comprising of at least two text lines and hence fine segmentation of the underlying document into line, word and character, is not necessary. Consequently, the script classification task is simplified and performed faster with the global approach than the local approach. So, global schemes can be best suited for a generalized approach to the script identification problem. Adequate amount of work has been reported in literature using global approaches [2, 4, 6]. Santanu Choudhuri, et al. [4] have proposed a method for identification of Indian languages by combining Gabor filter based technique and direction distance histogram classifier considering Hindi, English, Malayalam, Bengali, Telugu and Urdu. Gopal Datt Joshi, et. al. [6] have presented a script identification technique for 10 Indian scripts using a set of features extracted from log-Gabor filters. Dhanya et al. [14] have used Linear Support Vector Machine (LSVM), K-Nearest Neighbour (K-NN) and Neural Network (NN) classifiers on Gabor-based and zoning features to classify Tamil and English scripts. Recently, Hiremath et al. [15] have proposed a novel approach for script identification of South Indian scripts using wavelet based co-occurrence histogram features. Ramachandra Manthalkar et.al. [16] have proposed a method based on rotation-invariant texture features using multichannel Gabor filter for identifying seven Indian languages namely Bengali, Kannada, Malayalam, Oriya, Telugu and Marathi. They [16] have used multichannel Gabor filters to acquire rotation invariant texture features. From their experiment, they observed that rotation invariant features provide good results for script identification. Srinivas Rao Kunte et al. [17] have suggested a neural approach in on-line script recognition for Telugu language employing wavelet features. Peeta Basa Pati et al. [18] have presented a technique using Gabor filters for document analysis of Indian bilingual documents.

Sufficient amount of work has also been carried out on non-Indian languages [2, 22-30]. One of the first attempts in automatic script and language recognition is due to Spitz and his coworkers [23]. Assuming connected components of individual characters have been extracted from the document image, Spitz first locates upward concavities in the connected components. He then discriminates Asian languages (Japanese, Chinese, and Korean) against European languages (English, French, German, and Russian) based on the vertical distributions of such concavities. The three Asian languages are differentiated from each other by comparing the statistics of the optical densities (the number of black pixels per unit area) of the connected components, whereas the European languages are discriminated by means of the most frequent occurring

word shape tokens also derived from the connected components [23]. Tan [2] has developed a rotation invariant texture feature extraction method for automatic script identification for six languages: Chinese, Greek, English, Russian, Persian and Malayalam. Peake and Tan [19] have proposed a method 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, Korean, Malayalam, Persian and Russian.

Hochberg et al. [25] have presented a method for automatically identifying script from a binary document image using cluster-based text symbol templates. The system develops a set of representative symbols (templates) for each script by clustering textual symbols from a set of training documents and represents each cluster by its centroid. "Textual symbols" include discrete characters in scripts such as Cyrillic, as well as adjoined characters, character fragments and whole words in connected scripts such as Arabic. To identify a new document's script, the system compares a subset of symbols from the document to each script's templates, screening out rare or unreliable templates and choosing the script whose templates provide the best match. Later, they have extended their work on thirteen scripts - Arabic, Armenian, Burmese, Chinese, Cyrillic, Devanagari, Ethiopic, Greek, Hebrew, Japanese, Korean, Roman, and Thai. Chew Lim Tan et al. [24] presents a technique of identification of English, Chinese, Malay and Tamil in image documents using features like bounding boxes of character cells and upward concavities. Andrew Busch et al. [22] have exploited the concept of texture features for script identification of English, Chinese, Greek, Cyrillic, Hebrew, Hindi, Japanese and Persian. Wood et al. [24] have proposed projection profile method to determine Roman, Russian, Arabic, Korean and Chinese characters. Zhu et al. have presented an unconstrained language identification technique using a shape codebook concept [26]. Later, Zhu et al. have extended their work on language identification of handwritten document images [27]. Lu et al. have explored the challenging problem of script and language identification on degraded and distorted document images [28, 29]. Stefan Jaeger et al [30] have proposed a multiple classifier system for script identification at word level using Gabor filter analysis of textures. Their system identifies Latin and non-Latin words in bilingual printed documents. The classifier system comprises of four different architectures based on nearest neighbors, weighted Euclidean distances, Gaussian mixture models, and support vector machines. Global approaches make use of the texture-based features. These texture features can be extracted from a portion of a text region that may comprise of several text lines.

Texture could be defined in simple form as “repetitive occurrence of the same pattern”. Texture could be defined as something consisting of mutually related elements. Another definition of texture claims that, “an image region has a constant texture if a set of its local properties in that region is constant, slowly changing or approximately periodic”. Texture classification is a fundamental issue in image analysis and computer vision. It has been a focus of research for nearly three decades. Briefly stated, there are a finite number of texture classes Ci, i = 1, 2, 3, n. A number of training samples of each class are available. Based on the information extracted from the training samples, a decision rule is designed, which classifies a given sample of unknown class into one of the n classes [2]. Image texture is defined as a function of the spatial variation in pixel intensities. The texture classification is fundamental to many applications such as automated visual inspection, biomedical image processing, content-based image retrieval and remote sensing. One application of image texture is the recognition of image regions using texture properties. From the literature survey, it is observed that sufficient work has been carried out using texture features [11-14, 18, 20]. Existing methods on Indian script identification use the texture features extracted from the co-occurrence matrix, wavelet based co-occurrence histogram [15] and Gabor filters [17, 18]. Hiremath and Shivakumar [31] have considered Haralick features for texture classification using wavelet packet decomposition. Very few works are reported on script identification particularly using wavelet transform based features. In this paper, the texture features useful for script identification are extracted from the co-occurrence matrix constructed from the sub band coefficients of the wavelet packets transforms. As such, no work has been reported that uses the wavelet packet based texture features for script identification.

The rest of the paper is organized as follows. The Section 2 briefs about the wavelet packet transform. The database constructed for testing the proposed model is presented in Section 3. In

Section 4, complete description of the proposed model is explained in detail. The experimental results obtained are presented in section 5. Conclusions are given in section 6.

2. WAVELET PACKET TRANSFORM

Research interest in wavelets and their applications has grown tremendously over the past few years. It has been shown that wavelet-based methods continue to be powerful mathematical tools and offer computational advantage over other methods for texture classification. The different wavelet transform functions filter out different range of frequencies (i.e. sub bands). Thus, wavelet is a powerful tool, which decomposes the image into low frequency and high frequency sub band images.

The Continuous Wavelet Transform (CWT) is defined as the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet function ψ given in Equation (1).

$$C(scale, position) = \int_{-\infty}^{-\infty} f(t)\psi(scale, position, t)dt \tag{1}$$

The results of the CWT are many wavelet coefficients C, which are functions of scale and position. The wavelet transform decomposes a signal into a series of shifted and scaled versions of the mother wavelet function. Due to time frequency localization properties, discrete wavelet and wavelet packet transforms have proven to be appropriate starting point for classification tasks. In the 2-D case, the wavelet transform is usually performed by applying a separable filter bank to the image. Typically, a low filter and a band pass filter are used. The convolution with the low pass filter results in the approximation image and the convolutions with the band pass filter in specific directions result in the detail images.

In the simple wavelet decomposition, only the approximation coefficients are split iteratively into a vector of approximation coefficients, and a vector of detail coefficients are split at a coarser scale. That means, for n-level decomposition, n+1 possible ways of decomposition are obtained as shown in Figure 1. The successive details are never reanalyzed in the case of simple wavelet decomposition.

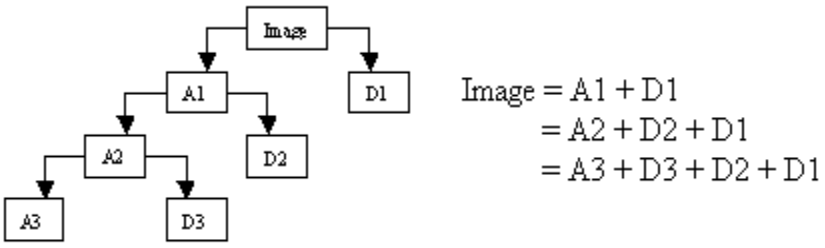

FIGURE 1. Wavelet Decomposition Tree

The concept of wavelet packets was introduced by Coifman et.al. [23]. In the case of wavelet packets, each detail coefficient vector is also decomposed as in approximation vectors. The recursive splitting of both approximate and detail sub images will produce a binary tree structure as shown in Figure 2.

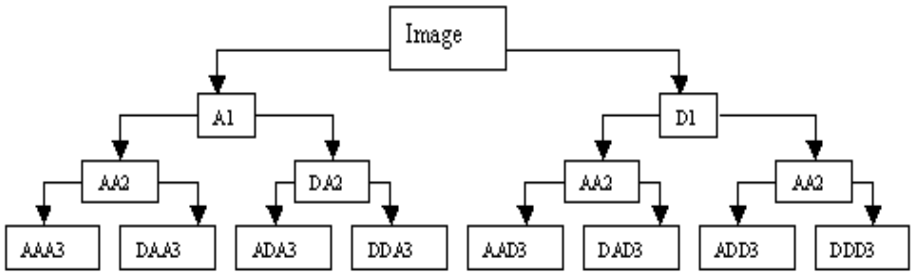

FIGURE 2. Wavelet Packet Decomposition Tree

Generally, in the case of wavelet transforms, the features are extracted from only the approximate sub band coefficients. But, in the case of wavelet packet transforms, the features are extracted from both approximate and detail sub band coefficients, as the detail sub band images can also be decomposed recursively. The features derived from a detail images uniquely characterize a texture. The combined transformed coefficients of the approximate and detail images give efficient features and hence could be used as essential features for texture analysis and classification. In this paper, the features are extracted from the sub bands of the transformed images for script identification and the complete description of the proposed model is given in Section 4.

3. DATA COLLECTION

Standard database of documents of Indian languages is currently not available. For the proposed model, the data set was constructed from the scanned document images. The printed documents like books, newspapers, journals and magazines were scanned through an optical scanner to obtain the document image. The scanner used for obtaining the digitized images is HP Scan Jet 5200c series. The scanning is performed in normal 100% view size at 300 dpi resolution. The image size of 256x256 pixels was considered. The training data set of 300 images and test data set of 250 images were used from each of the ten scripts.

4. OVERVIEW OF THE PROPOSED MODEL

Scripts are made up of different shaped patterns to produce different character sets. Individual text patterns of one script are collected together to form meaningful text information in the form of a text word, a text line or a paragraph. The collection of the text patterns of the one script exhibits distinct visual appearance. So, a uniform block of texts, regardless of the content, may be considered as distinct texture patterns (a block of text as single entity) [2]. This observation implies that one may devise a suitable texture classification algorithm to perform identification of text language. Hence, the proposed model is inspired by this simple observation that every language script defines a finite set of text patterns, each having a distinct visual appearance. In this model, the texture-based features are extracted from the sub band coefficients of wavelet packet transforms.

4.1 Preprocessing of Input Images

In general, text portion of the scanned document images are not good candidates for the extraction of texture features. The varying degrees of contrast in gray-scale images, the presence of skew and noise could all potentially affect such features, leading to higher classification error rates. In addition, the large areas of white space, unequal word and line spacing, and variable height of the text lines due to different font sizes can also have a significant effect on the texture

features. In order to reduce the impact of these factors, the text blocks from which texture features are to be extracted must undergo a significant amount of preprocessing. In this paper, preprocessing steps such as removal of non-text regions, skew-correction, noise removal and binarization is necessary. In the proposed model, text portion of the document image was separated from the non-text region manually. Skew detection and correction was performed using the technique proposed by Shivakumar [21]. 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. In this work, a global thresholding approach is used to binarize the scanned gray scale images, where black pixels having the value 0's correspond to object and white pixels having value 1's correspond to background. It is necessary to thin the document image as the texts may be printed in varying thickness. In this paper, the thinning process is achieved by using the morphological operations.

4.1.1 To Construct Uniform Block of Text

In general, the scanned image may not necessarily have uniform height of the text line Height of the text line is the difference between the topmost black pixel to the bottommost black pixel of a text line and it is obtained though horizontal projection profile. To obtain text lines of uniform height, the necessary threshold values are computed by calculating the mean and the standard deviation of the text line heights. All those text lines, which fall outside the threshold values (fixed through experimentation), are removed. This process is repeated by calculating the new mean and standard deviation of the remaining text block, until no remaining text lines are removed.

4.1.2 Inter-line Spacing Normalization

In a block of text, the white space between the text lines may vary and hence, it is necessary to normalize these white spaces. The width of the white spaces between the text lines is obtained by computing the distance between the horizontal runs of black pixels of the two text lines. The white space width of the text lines greater than eight pixels is reduced to eight pixels, which is fixed by experimental study. Thus, the spacing between the text lines is normalized to have uniform inter-line spacing.

4.1.3 Inter-word Spacing Normalization

Generally, some text lines might have varying inter-word spacing. So, it is necessary to normalize the inter-word spacing to a maximum of 5 pixels. Normalization of the inter-word spacing is achieved by projecting the pixels of each text line vertically; counting the number of white pixels from left to right and reducing the number of white pixels greater than 5 pixels to 5. Inter-word gaps smaller than 5 pixels are allowed to remain, as that does not affect to get text blocks.

4.1.4 Padding

A text line that is smaller than the average width of a text line generally appears in a block of text. This is due to the presence of headings or the text line that happens to be the last line of a paragraph. So, it is necessary to obtain a collection of text lines of equal length by padding the text lines of smaller length. Padding of text lines is done by copying and replicating the text content of the line so that the length of the text line would be increased to the average length of text line.

Then, the text block of size 256X256 pixels is extracted from the preprocessed text block. From each block of preprocessed text image, the texture features are extracted for the purpose of script identification.

4.2 Texture Feature Extraction

In this work, the known input images are decomposed through the Wavelet Packet Decomposition using the Haar (Daubechies 1) basis function to get the four sub band images namely Approximation (A) and three detail coefficients - Horizintal (H), Vertical (V) and the Diagonal (D). The Haar wavelet transformation is chosen because the resulting wavelet bands are strongly correlated with the orientation elements in the GLCM computation. The second reason is that the total pixel entries for Haar wavelet transform are always minimum. Through experimentation the Haar basis function up to the level two is found to be best, yielding distinct features and hence Haar basis function up to level two is used in this method. This result in a total of 20 sub bands, four sub bands at the first level and sixteen sub bands (four for each sub band) in the next level as shown in Figure 3.

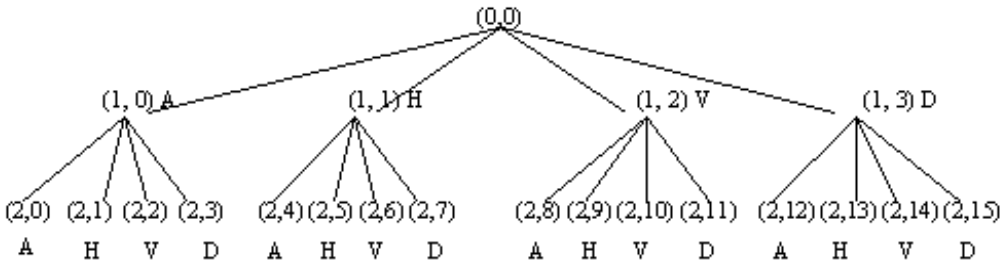

FIGURE 3. Wavelet Packet Tree up to Level - 2. (A–Approximation, H–Horizontal, V–Vertical and D–Diagonal)

It is not necessary to consider all the twenty sub bands for feature extraction. This is because, the four types of sub bands – approximation, horizontal, vertical and diagonal obtained from the wavelet transforms retain specific type of information by filtering out other information. Therefore, when the horizontal, vertical and diagonal sub bands are decomposed further into four bands, all the four sub bands may not be necessarily used as some information in the second level is lost. So, it is necessary to consider only the relevant sub bands at the second level. For example, in the sub band (1, 1) of level one which gives only horizontal detail coefficients, it is sufficient to consider only the approximation and horizontal detail coefficients of its second level. The vertical and diagonal sub bands of (1, 1) are not considered as they exhibit less or poor information. Thus, the sub bands that exhibit the similar type of coefficients from the two levels are selected. In the proposed method, the sub bands of the two levels are combined to form four groups as given below.

Group 1: Approximation sub bands: (1, 0), (2, 0) = (A, AA)
Group 2: Horizontal sub bands: (1, 1), (2, 1), (2, 4), (2, 5) = (H, AH, HA, HH)
Group 3: Vertical sub bands: (1, 2), (2, 2), (2, 8), (2, 10) = (V, AV, VA, VV)
Group 4: Diagonal sub bands: (1, 3), (2, 3), (2, 12), (2, 15) = (D, AD, DA, DD)

Thus, only fourteen sub bands - two approximate sub band, four horizontal sub band, four vertical sub bands and four diagonal sub bands are selected out of the twenty sub bands.

Many researchers have used the coefficient values of the approximate and detail sub band images as a texture feature vector [15]. According to Andrew Busch [22], a better representation of natural textured images can be obtained by applying a nonlinear function to the coefficients of the wavelet transform. In this paper, the nonlinear transform function proposed by Andrew Busch [22] is applied on the wavelet packet coefficients. So, the wavelet packet coefficients are quantized using the quantization function derived by Andrew Busch [22]. Then, gray level co-occurrence matrices are constructed for the quantized wavelet sub bands. The description of the gray level co-occurrence matrix is briefed out in the next section.

4.2.1. Gray Level Co-occurrence Matrices (GLCMs)

Gray Level Co-occurrence Matrix (GLCM) has been proven to be a very powerful tool for texture image segmentation. Gray-level co-occurrence matrices (GLCMs) are used to represent the pair wise joint statistics of the pixels of an image and have been used for many years as a means of characterizing texture [22]. GLCM is a two dimensional measure of texture, which show how often each gray occurs at a pixel located at a fixed geometric position relative to each other pixel, as a function of its gray level. GLCMs are in general very expensive to compute due to the requirement that the size of each matrix is NxN, where N is the number of gray levels in the image. So, it is necessary to reduce the number of discrete gray levels of the input image in order to obtain co-occurrence matrix of smaller size. So, if the gray levels are divided into fewer ranges, the size of the matrix would be reduced, thus leading to less noisy entries in the matrix.

In this paper, the gray levels of the quantized sub bands are divided into fewer ranges to obtain a new transformed sub band which results in reduced size of the co-occurrence matrix. Then, from

the new transformed sub bands, GLCMs are constructed using the values d=1, where d represents the linear distance in pixels. The value of θ is fixed based on the type of the sub band. For the approximate sub bands i.e., group1 ((1, 0), (2, 0)), GLCMs are constructed with the value θ = {00, 450, 900, 1350}. The value of θ is taken as 00 for horizontal sub bands (group2), 900 for vertical sub bands (group3) and, 450 and 1350 for diagonal sub bands (group4). Thus, totally, twenty four GLCM (eight GLCM for group1, four GLCM for group2, four GLCM for group3 and eight GLCM for group4) are constructed.

Haralick [32] has proposed the textural features that can be extracted from the co-occurrence matrix. In this paper, Haralick texture features [32] such as inertia, total energy, entropy, contrast, local homogeneity, cluster shade, cluster prominence, and information measure of correlation are extracted from the gray level co-occurrence matrices obtained from the coefficients of the sub bands. These texture features are given in Table 1. These features are known as the wavelet packet co-occurrence features.

TABLE 1. Wavelet Packet Co-occurrence Features Extracted from a Co-occurrence Matrix C(i, j).

Inertia:
$$F1 = \sum_{i,j=0}^{n} (i-j)^2 C(i,j)$$

Total Energy:
$$F2 = \sum_{i,j=0}^{n} C^2(i,j)$$

Entropy:
$$F3 = -\sum_{i,j=0}^{n} C(i,j) \log C(i,j)$$

Contrast:
$$F4 = -\sum_{i,j=0}^{n} C(i,j) \left| i-j \right|^k, k \in \mathbf{Z}$$

Local Homogeneity:
$$F5 = \sum_{i,j=0}^{n} \frac{1}{1+(i-j)^2} C(i,j)$$

Cluster Shade:
$$F6 = \sum_{i,j=0}^{n} \left(i - M_x + j - M_y \right)^3 C(i,j)$$

Cluster Prominence:
$$F7 = \sum_{i,j=0}^{n} \left(i - M_x + j - M_y \right)^4 C(i,j)$$

Information Measure of Correlation:
$$F8 = \frac{-\sum_{i,j=0}^{n} C(i,j) \log C(i,j) - H_{x,y}}{\max\left(H_x, H_y\right)}$$

where
$$M_x = \sum_{i,j=0}^{n} iC(i,j) \quad \text{and} \quad M_y = \sum_{i,j=0}^{n} jC(i,j)$$

$$H_{x,y} = -\sum_{i,j=0}^{n} C(i,j) \log\left(\sum_{j=0}^{n} C(i,j) . \sum_{i=0}^{n} C(i,j) \right)$$

$$H_x = -\sum_{i=0}^{n} \left\{ \sum_{j=0}^{n} P(i,j) . \log \sum_{j=0}^{n} P(i,j) \right\}$$

$$H_y = -\sum_{j=0}^{n} \left\{ \sum_{i=0}^{n} P(i,j) . \log \sum_{i=0}^{n} P(i,j) \right\}$$

The eight Haralick texture features are extracted from the twenty four GLCMs resulting in a total of 192 features. In order to reduce the dimension of the features, the mean values of the eight features are computed individually from each of the four groups. That means, eight features of any sub band is obtained by taking the average of the eight features computed from the GLCMs of that sub band. For example, from the four GLCMs of the sub band (1, 0), average of the eight features are computed. Thus, eight features of the sub band (1, 0) and eight features of sub band (2, 0) of group 1, results in 16 texture features. But, the average value of each of the eight features is computed individually from the sub bands of the corresponding groups. For example, the average value of the feature – 'Inertia' for the Group-1 is computed from the inertia of sub band (1, 0) and the inertia of sub band (2, 0). Similarly, the average value of each of the eight features is computed from sub bands of the individual groups, resulting in 32 features (8 features each from four groups), thus reducing the dimensionality of the features from 192 to 32. As the features are extracted from the two levels of wavelet packet transforms and then the average of the feature values are computed, the features could be considered as optimal features. These optimal features are strong enough to discriminate the ten script classes considered in this paper. The 32 optimal features are obtained from a training data set of 300 images from each of the ten Indian scripts - Bangla, Devanagari, Roman (English), Gujarati, Malayalam, Oriya, Tamil, Telugu, Kannada and Urdu. These features are stored in a feature library and used as texture features later in the testing stage.

4.3 Classification

In the proposed model, K -nearest neighbor classifier is used to classify the test samples. The features are extracted from the test image X using the proposed feature extraction algorithm and then compared with corresponding feature values stored in the feature library using the Euclidean distance formula given in equation (2),

$$D(M) = \sqrt{\sum_{j=1}^{N} [f_j(x) - f_j(M)]^2} \tag{2}$$

where N is the number of features in the feature vector f, fj(x) represents the jth feature of the test sample X and fj(M) represents the jth feature of Mth class in the feature library. Then, the test sample X is classified using the k-nearest neighbor (K-NN) classifier. In the K -NN classifier, a test sample is classified by a majority vote of its k neighbors, where k is a positive integer, typically small. If K =1, then the sample is just assigned the class of its nearest neighbor. It is better to choose K to be an odd number to avoid tied votes. So, in this method, the K -nearest neighbors are determined and the test image is classified as the script type of the majority of these K-nearest neighbors. The testing algorithm employed in this paper for script identification consists of the following steps.

Algorithm Testing ()
Input: Text portion of the document image containing one script only.
Output: Script type of the test document.

1. Preprocess the input document image.
2. Analyze the test image using 2-d Wavelet Packet Transform with Haar wavelet up to level 2 and obtain the wavelet packet tree.
3. Quantize the wavelet coefficients using the quantization function derived by Andrew Busch [24].
4. Select the sub bands from the wavelet packet tree as given below:
   Group 1: Approximation sub bands: (1, 0), (2, 0) = (A, AA)
   Group 2: Horizontal sub bands: (1, 1), (2, 1), (2, 4), (2, 5) = (H, AH, HA, HH)
   Group 3: Vertical sub bands: (1, 2), (2, 2), (2, 8), (2, 10) = (V, AV, VA, VV)
   Group 4: Diagonal sub bands: (1, 3), (2, 3), (2, 12), (2, 15) = (D, AD, DA, DD)
5. Construct the Gray Level Co-occurence Matrix of each sub band.

6. Extract the eight Haralick texture features given in Table 1 from the selected sub bands.
7. Compute the average value for each of the eight features from the four groups as explained in the previous section.
8. Classify the script type of the test image by comparing the feature values of the test image with the feature values stored in the knowledge base using K-nearest neighbor classifier.

The experiment is conducted for varying number of neighbors like K = 3, 5 and 7. The performance of classification was best when the value of K = 3. Thus the script type of the test image is classified by comparing the feature values of the test image with the feature values stored in the feature matrix using K-nearest neighbor classifier.

5. EXPERIMENTAL RESULTS

To test the proposed model test data set was constructed from the scanned images of newspapers, magazines, journals and books at 300 dpi grayscale. Around 250 images were chosen from each of ten Indian scripts namely Bangla, Devanagari, Roman (English), Gujarati, Malayalam, Oriya, Tamil, Telugu, Kannada and Urdu. Only the text portion of the scanned images was selected. However, images that would contain multiple columns and variable inter-line and inter-word spacing were also considered as the preprocessing steps are used to prepare them for further processing. Sample images of Bangla, Hindi, Kannada, English, Gujarati, Malayalam, Oriya, Telugu, Urdu and Tamil scripts are shown in Figure 4. The proposed algorithm is implemented using Matlab R2007b. The average time taken to identify the script type of the document is 0.2843 seconds on a Pentium-IV with 1024 MB RAM based machine running at 1.60 GHz.

FIGURE 4. Sample images of Bangla, Hindi, Kannada, English, Gujarati, Malayalam, Oriya, Telugu, Urdu and Tamil scripts.

5.1 Optimal Size of Training Data Set

It is necessary to determine the optimal size of the training data set to obtain best performance of the proposed model. In the proposed model, the test samples are tested with varying number of training data set. The overall performance of recognition verses size of the training data set is shown in Figure 5. From the Figure 5, it is observed that the proposed method attains best performance of 98.24% with an optimal training data set of 300 samples.

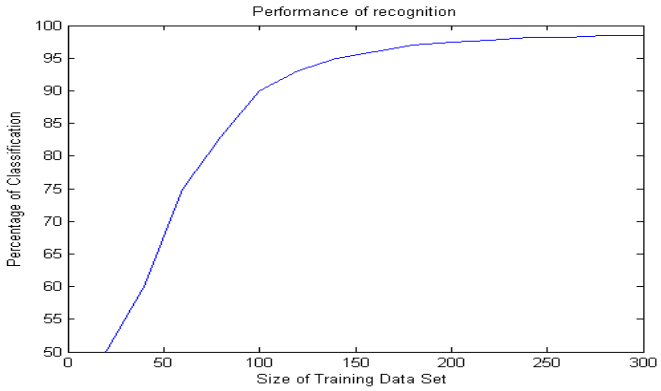

FIGURE 5. The overall performance of recognition verses size of the training data set.

The confusion matrix of the proposed method for classifying twelve Indian scripts through extensive experimentation is given in Table 2. The average classification accuracy of the proposed wavelet based method is 98.24%. The experimental results demonstrate the effectiveness of the proposed texture features.

Test images having some special characters like numerals, punctuation marks and italicized text were also considered. However, the presence of these symbols does not affect the recognition rate as they are rarely seen in a meaningful text region. A small amount of page skew was inevitably introduced during the scanning process. This skew was compensated by using the method outlined in [21].

TABLE 2. Percentage of Recognition of 10 Indian scripts (Ban=Bangla, Dev=Devanagari, Rom=Roman (English), Guj=Gujarati, Mal=Malayalam, Ori=Oriya, Tam=Tamil, Tel=Telugu, Kan=Kannada and Urd=Urdu)

Input Script Type	Classified Script Type										
	Ban	Dev	Rom	Guj	Ori	Mal	Tam	Tel	Kan	Urd	Rejected
Ban	246	3	-	1	-	-	-	-	-	-	-
Dev	2	248	-	-	-	-		-	-	-	-
Rom	-	-	245	-	2	-	2	-	-	-	1
Guj	-	-	2	246	1	-	-	-	-	-	1
Ori	-	-	2	1	245	-	1	-	-	-	1
Mal	-	-	1	-	-	244	2	-	2	-	1
Tam	-	-	3	-	2		245	-	-	-	-
Tel	-	-	-	-	-	2	1	244	3	-	-
Kan	-	-	-	-	-	2	-	2	246	-	-
Urd	-	-	-	-	-	-	-	-	-	247	3
Percentage of classification	98.4	99.2	98.0	98.4	98.0	97.6	98.0	97.6	98.4	98.8	-
Error	1.6	0.8	2.0	1.6	2.4	2.8	2.0	2.8	1.6	1.2	-

6. CONSLUSION

In this paper, a global approach for script identification that uses the wavelet based texture features is presented. The texture features are extracted from the GLCMs constructed from a set of wavelet packet sub band coefficients. The experimental results demonstrate that the new approach can better perform in classifying ten Indian scripts. The performance of the proposed model shows that the global approach could be used to solve a practical problem of automatic script identification.

7. REFERENCES

1. U.Pal, B.B.Choudhuri, : Script Line Separation From Indian Multi-Script Documents, 5th Int. Conference on Document Analysis and Recognition(IEEE Comput. Soc. Press), 406-409, (1999).
2. T.N.Tan,: Rotation Invariant Texture Features and their use in Automatic Script Identification, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 7, pp. 751-756, (1998).
3. U. Pal, S. Sinha and B. B. Chaudhuri : Multi-Script Line identification from Indian Documents, In Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003) 0-7695-1960-1/03 © 2003 IEEE, vol.2, pp.880-884, (2003).
4. Santanu Choudhury, Gaurav Harit, Shekar Madnani, R.B. Shet, : Identification of Scripts of Indian Languages by Combining Trainable Classifiers, ICVGIP, Dec.20-22, Bangalore, India, (2000).
5. S. Chaudhury, R. Sheth, “Trainable script identification strategies for Indian languages”, In Proc. 5th Int. Conf. on Document Analysis and Recognition (IEEE Comput. Soc. Press), pp. 657–660, 1999.
6. Gopal Datt Joshi, Saurabh Garg and Jayanthi Sivaswamy, :Script Identification from Indian Documents, LNCS 3872, pp. 255-267, DAS (2006).
7. 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. © Printed in India, (2002).
8. B.V. Dhandra, Mallikarjun Hangarge, Ravindra Hegadi and V.S. Malemath,: Word Level Script Identification in Bilingual Documents through Discriminating Features, IEEE - ICSCN 2007, MIT Campus, Anna University, Chennai, India. pp.630-635. (2007).
9. U. Pal and B. B. Chaudhuri, “Automatic separation of Roman, Devnagari and Telugu script lines”, Advances in Pattern Recognition and Digital techniques, pp. 447-451, 1999.
10. Lijun Zhou, Yue Lu and Chew Lim Tan,: Bangla/English Script Identification Based on Analysis of Connected Component Profiles, in proc. 7th DAS, pp. 243-254, (2006).
11. M. C. Padma and P.Nagabhushan,: Identification and separation of text words of Karnataka, Hindi and English languages through discriminating features, in proc. of Second National Conference on Document Analysis and Recognition, Karnataka, India, pp. 252-260, (2003).
12. M. C. Padma and P.A.Vijaya,: Language Identification of Kannada, Hindi and English Text Words Through Visual Discriminating Features, International Journal of Computational Intelligence Systems (IJCIS), Volume 1, Issue 2, pp. 116-126, (2008).
13. Vipin Gupta, G.N. Rathna, K.R. Ramakrishnan,: A Novel Approach to Automatic Identification of Kannada, English and Hindi Words from a Trilingual Document, Int. conf. on Signal and Image Processing, Hubli, pp. 561-566, (2006).
14. D. Dhanya, A. G. Ramakrishnan, and P. B. Pati, “Script identification in printed bilingual docuements”, Sadhana, vol. 27, pp. 73-82, 2002.
15. Hiremath P S and S Shivashankar, “Wavelet Based Co-occurrence Histogram Features for Texture Classification with an Application to Script Identification in a Document Image”, Pattern Recognition Letters 29, 2008, pp 1182-1189.
16. Ramachandra Manthalkar and P.K. Biswas, “An Automatic Script Identification Scheme for Indian Languages”, IEEE Tran. on Pattern Analysis And Machine Intelligence, vol.19, no.2, pp.160-164, Feb.1997.

17. R. Sanjeev Kunte and R.D. Sudhakar Samuel, “On Separation of Kannada and English Words from a Bilingual Document Employing Gabor Features and Radial Basis Function Neural Network", Proc. of ICCR, pp. 640-644, 2005.
18. Peeta Basa Pati, S. Sabari Raju, Nishikanta Pati and A. G. Ramakrishnan, “Gabor filters for Document analysis in Indian Bilingual Documents”, 0-7803-8243-9/04/ IEEE, ICISIP, pp. 123-126, 2004.
19. G. S. Peake and T. N. Tan, “Script and Language Identification from Document Images”, Proc. Workshop Document Image Analysis, vol. 1, pp. 10-17, 1997.
20. Rafael C. Gonzalez, Richard E. Woods and Steven L. Eddins,: Digital Image Processing using MATLAB, Pearson Education, (2004).
21. Shivakumar, Nagabhushan, Hemanthkumar, Manjunath, 2006, “Skew Estimation by Improved Boundary Growing for Text Documents in South Indian Languages”, VIVEK-International Journal of Artificial Intelligence, Vol. 16, No. 2, pp 15-21.
22. Andrew Busch, Wageeh W. Boles and Sridha Sridharan, “Texture for Script Identification”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 11, pp. 1720-1732, Nov. 2005.
23. A. L. Spitz, “Determination of script and language content of document images”, IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 19, No.3, pp. 235–245, 1997.
24. Chew Lim Tan, Peck Yoke Leong, Shoujie He, “Language Identification in Multilingual Documents”, International symposium on intelligent multimedia and distance education, 1999.
25. J. Hochberg, L. Kerns, P. Kelly and T. Thomas, “Automatic script identification from images using cluster based templates”, IEEE Trans. Pattern Anal. Machine Intell. Vol. 19, No. 2, pp. 176–181, 1997.
26. Guangyu Zhu, Xiaodong Yu, Yi Li and David Doermann, “Unconstrained Language Identification Using A Shape Codebook”, The 11th International Conference on Frontiers in Handwritting Recognition (ICFHR 2008), pp. 13-18, 2008.
27. Guangyu Zhu, Xiaodong Yu, Yi Li and David Doermann, “Language Identification for Handwritten Document Images Using A Shape Codebook”, Pattern Recognition, 42, pp. 3184-3191, December 2009.
28. Lu S and C.L. Tan, “Script and Language Identification in Degraded and Distorted Document Images,” Proc. 21st Nat’l Conf. Artificial Intelligence, pp. 769-774, 2006.
29. Lu Shijian and Chew Lim Tan, “Script and Language Identification in Noisy and Degraded Document Images”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 30, No. 1, pp. 14-24, 2008.
30. Stefan Jaeger, Huanfeng Ma and David Doermann, “Identifying Script on Word-Level with Informational Confidence”, 8th Int. Conf. on Document Analysis and Recognition, pages 416-420, August 2005.
31. Hiremath P S and S Shivashankar, “Texture classification using Wavelet Packet Decomposition”, ICGSTs GVIP Journal, 6(2), 2006, pp. 77-80.
32. R.M.Haralick, K. Shanmugam and I.Dinstein, “Textural features for Image Classification”, IEEE Transactions on Systems, Man and Cybernetics, Vol.3, pp. 610-621, 1973.