SCRIPTS AND NUMERALS IDENTIFICATION FROM PRINTED MULTILINGUAL DOCUMENT IMAGES

Abirami.S[1]and Murugappan. S[2]
[1]Department of Information Science and Technology, Anna University, Chennai
abirami_mr@yahoo.com
[2]Department of Computer science & Engineering, Annamalai University
drmryes@gmail.com

ABSTRACT

Identification of scripts from multi-script document is one of the important steps in the design of an OCR system for successful analysis and recognition. Most optical character recognition (OCR) systems can recognize at most a few scripts. But for large archives of document images containing different scripts, there must be some way to automatically categorize these documents before applying the proper OCR on them. Much work has already been reported in this area. In the Indian context, though some results have been reported, the task is still at its infancy. This paper presents a research in the identification of Tamil, English and Hindi scripts at word level irrespective of their font faces and sizes. It also identifies English numerals from multilingual document images. The proposed technique performs document vectorization method which generates vectors from the nine zones segmented over the characters based on their shape, density and transition features. Script is then determined by using Rule based classifiers and its sub classifiers containing set of classification rules which are raised from the vectors. The proposed system identifies scripts from document images even if it suffers from noise and other kinds of distortions. Results from experiments, simulations, and human vision encounter that the proposed technique identifies scripts and numerals with minimal pre-processing and high accuracy. In future, this can also be extended for other scripts.

KEYWORDS

Document Images, Script Recognition, Classification, Document Image Understanding

1. INTRODUCTION

With recent emergence and widespread application of computers and multimedia technologies, there is an increasing demand to create a paperless environment [3]. Therefore all printed documents are converted into document images. In the early 90s, several document analysis systems appeared are able to handle single language documents. Identification of script is relatively little attention in that document analysis field because one can normally deduce a document's script from its country of origin, or by examining the document or it can be performed manually based on people's experience.

Multi-lingual country like India (India has 18 regional languages derived from 12 different scripts) uses multi script documents like bus reservation forms, passport application forms, examination question papers, bank- challan, language translation books and money-order forms

David Bracewell, et al. (Eds): AIAA 2011,CS & IT 03, pp. 129–146 , 2011.
 DOI : 10.5121/csit.2011.1312

containing text words in more than one language. In such environment, multi lingual OCR system is needed to read the multilingual documents.
Though most existing OCR systems are equipped with multiple OCR engines, manual routing is still required to switch incoming document images to the proper OCR engine for large achieves of document images. But with the knowledge of the underlying script and language, incoming document images can be switched automatically and this significantly reduces the human involvement. So, there must be some way to automatically categorize these documents before applying the proper OCR on them. To make a multi-lingual OCR system successful, it is necessary to separate portions of different language regions of the document before feeding to individual OCR systems.

Since the amount of multimedia data captured and stored is increasing rapidly with the advances in computer technology. Such data include multi-lingual documents. for example, magazines, museums, etc… that may store images of all old fragile documents having scientific or historical or artistic value and written in different scripts which are stored in typical large databases. [9][16][1]. in such environment the large volume of data and variety of scripts makes such manual identification unworkable[9][16]. In such cases the ability to automatically determine the script, and further, the language of a document, would reduce the time and cost of document handling. So the development of script identification from multi-lingual document image systems has become an important task.

Script and language identification aims to determine the underlying script of document text, either in an imaged or a character-coded format [10]. This is one of the fundamental issues in document analysis. The capability of recognizing multi-lingual document is both novel and useful, with such capability, many potential applications can be supported including multi-lingual access to patent, business and regulatory information, document sorting in support of character recognition, translation and keyword finding in document images[4] [2]. Dealing with multi-lingual document raises many challenges including script identification, language determination, text reading direction and differing character sets and Most of the document images are affected by noise. In addition to this, the large numbers of multilingual documents contains numerals in addition to multiple scripts. This has motivated us to develop a method for automatic script identification of text words and numerals in multilingual documents images.

In this paper, we present a document Vectorization technique to detect scripts from document images. The proposed document Vectorization technique converts the first character of each word in a document image into a nine bit vector. In particular, a vector is first constructed by using the density of the character. The underlying scripts of document images are then determined by using rule based classifiers and its sub classifiers which are created by using set of training document images. Since we intend to develop a script recognizer to discriminate between Tamil, English, Hindi scripts and numerals, an overview about the Tamil, English and Hindi scripts, English numerals and its character set has been explained here.

1.1 Tamil Script

Tamil is a South Indian language spoken widely in Tamil Nadu, India. Tamil has the longest unbroken literary tradition amongst the Dravidian languages. Tamil is inherited from Brahmi script. The Tamil script has 12 vowels, 18 consonants, 1 special character and 216 consonant based vowels. Hence a set of 247 symbols exists in the Tamil script. The modifier symbols occupy specific positions around the base characters. While the modifiers that get added on the left or the right side remain disjoint from the base character, the modifier symbols that are added either at the top or the bottom get connected to the base character and spread all to the upper,

middle and lower zones respectively. There is a dominance of horizontal and vertical strokes in the Tamil script. Tamil script spread over its width and height.

1.2 Roman Script

Roman script has 26 each of upper and lower case characters. In addition, there are some special symbols and numerals. While the capital letters of the Roman script occupy the middle and the upper zones, most of the lower case characters have a spatial spread that covers only the middle zone or the middle and the lower zones. The structure of the Roman alphabet contains more vertical and slant strokes and less horizontal transition in the middle zone compared to Tamil.

1.3 Hindi Script

In Hindi (Devanagari) language, many characters have a horizontal line at the upper part. This line is called sirorekha in Devanagari [20]. This is also called as head-line. It could be seen that, when two or more characters sit side by side to form a word, the character head-line segments mostly join one another in a word resulting in only one component within each text word and generates one continuous head-line for each text word. Since the characters are connected through their head-line portions, a Hindi word appears as a single component and hence it cannot be segmented further into blocks, which could be used to recognize this script.

1.4 Numerals

English numerals possess similar characteristics of English characters with less numbers of horizontal transitions. Tamil, English, Hindi Scripts and numerals which are considered for identification is shown in figure 1.

(a) Tamil Script

(b) Roman Script

(c) Hindi words

(d) Numerals

Figure 1 Tamil, English, Hindi scripts and Numerals

The remainder of this paper is organized as follows: Related Script and language identification work has been reviewed in Section 2. Proposed Language identification technique is discussed in

Section 3. Experimental Results and performance measures are dealt in Section 4 and 5. Concluding remarks has been given in Section 6.

2. RELATED WORK

In general, Script recognition work can be classified into two categories: Statistics based approach (local Approach), Texture based approach (Global Approach)[Abirami].Local approach follows LWC i.e. Line, Word and Character segmentation. Here the components are available only after the line, word and character segmentation. Analysis of connected components like line, word and character in the document image is to identify the script of the document images at text line level or word level. [2] [14]. Gabor filter based classification is used for global approach.[21][26][11].

In texture based approach, Busch et al. [2] use the texture as a tool for determining the script. It was performed by designing a log-mean Gabor filter and wavelet energy features are extracted from them. Ma et al. [18] have done script identification at the word level by Gabor filters with multi-class classifier. Qiao et al [26], detected scripts from document images with different styles and fonts using a 2D Gabor filter designed with the spatial orientation to determine the text area and for script identification.

He et al [11have performed script identification, using centrally symmetric matrices and non separable bivariate filter banks. Zhu et al performed font recognition using different combinations of Gabor channels. Peake and Tan [25] have proposed a method for automatic script and language identification using multiple channel (Gabor) filters and gray level co-occurrence matrices for seven languages. Tan [30]has developed rotation invariant texture feature extraction method for automatic script identification for six languages.

Pal and Chaudhuri [20] have proposed an automatic technique for separating the text lines from 12 Indian scripts. Chaudhuri et al. [20] have proposed a method for identification of Indian languages by combining Gabor filter based technique and direction distance histogram. Pal and Chaudhuri [22] have proposed an automatic separation of Bangla, Devanagari and Roman words in multilingual multiscript Indian documents. Pal et. al. [23] has suggested a word-wise script identification model from a document containing English, Devanagari and Telugu text.

In the work of Pati et al. [24] script recognition in bilingual document script (between Tamil and English) was performed using Gabor filters. Dhanya et al [9], has performed script classification using Gabor filters for bilingual document images in frequency domain. In the work of Joshi et al. [14], different Indian scripts are identified with a Log Gabor filter. Regarding token based approach Judith Hochberg et al [8], [9] exemplars are grouped into clusters based on a similarity measure. A representative template is then extracted from each cluster to identify those characters.

In statistics based approach, Spitz [27] reports an identification method which is used for distinguishing Asian and European languages by examining the upward concavities of connected components. Wood [31] has proposed a method to identify scripts using horizontal and vertical projections of the document image. Spitz [28] has developed a scheme for automatic language identification based on character density or optical density distribution. The method first classifies the script into two broad classes: Han-based and Latin (or Roman)-based. This classification is based on the spatial relationships of features corresponding to the upward concavities. Language identification within the Han script class (Chinese, Japanese, and Korean) is performed by analysis of the distribution of optical density in the text image while language

determination within the Latin class is based on the most frequently occurring characteristic word shapes. It uses character level identification.

Pal & Chaudhuri [21] have proposed a method based on a decision tree for recognizing the script of a line of text.(textline level). They have used the projection profile besides statistical, topological and stroke-based features. Anoop [1] has proposed a method to classify words and lines in an online handwritten document into one of the six major scripts. The classification is based on 11 different spatial(HID,ASL, Shirorekha Strength, Shirorekha Confidence, Stroke Density, Aspect Ratio, Reverse Distance, Average Horizontal Stroke Direction, Average Vertical Stroke Direction, Vertical Interstroke Direction (VID), Variance of Stroke Length) and temporal features extracted from the strokes of the words. This is word level identification.

Two different approaches have been proposed by Dhanya [9]. In the first method, words are divided into three distinct spatial zones. The spatial spread of a word in upper and lower zones, together with the character density, is used to identify the script. The second technique analyses the directional energy distribution of a word using Gabor filters with suitable frequencies and orientations.

Dhandra et al [5] has proposed a method which is used to identify scripts in bilingual document page containing text words in regional language and numerals in English. It examines the use of discriminating features (aspect ratio, strokes, eccentricity, etc,). Dhandra et al [8]developed a system that includes a feature extractor using morphological operation and a classifier. The feature extractor consists of two stages. Dhandra et al [6][7] are continuation of paper [8] demonstrates the feasibility of morphological reconstruction approach for script identification at word level.

Cheng et al [3] performed language identification for Chinese, Japanese, English and Russian languages based on the spatial distribution. This is done by area segmentation of normalized histogram. In the work of Chew Lim Tan et al, [29] script was discriminated as Chinese, Latin and Tamil scripts by boundary box elongation and upward concavities projection, which could work at certain threshold levels. Lu et al [17] performed script identification for Roman, Latin, Korean, Chinese, Arabic scripts at document level. Script is identified by document vectorization method. But this could work only at the document level and not at the word level. Padma et al[19] has proposed a line-wise and word-wise identification models to identify Kannada, Hindi and English text words from Indian multilingual machine printed documents. The proposed models are developed based on the four visual discriminating features, which serve as useful visual clues for language identification.

With respect to regional Languages, Pal et al. and group, Padma et al., etc… proposed different schemes for identifying English and Hindi. Tamil and English scripts has been done by a group of people [4],[14],[28], etc.. Among the works reported in this direction, to distinguish between various Indian scripts at word level is addressed only alphabet-based script identification, whereas numeral script identification is ignored. Numeral identification techniques were reported by Dhandra et al. Noise removal techniques were reported by Lu et al [17].

Our survey for previous research work in the area of document script/language identification shows that much of them rely on script/languages followed by other countries and few from our country. There is no work reported for the identification of Indian scripts such as Tamil, English, Hindi and Numerals. Moreover, no script recognizer for Tamil and English could produce a better discrimination among them in spatial approach.

The work reported in [28] fail if both English and Tamil words comes in the same area. Dhanya et al [14], has achieved better performance in the latter method rather than the Pati et al [24]. But this would fail for some English characters which are descender dominant. Dhanya's approach [9] produces better results but it fails when the quality degrades and it cannot be extended further. Most of the script identification methods use global approaches which do not require fine segmentation. These methods are fast and inexpensive to handle documents but gives poorer classification accuracy for low quality. On the other hand, spatial approaches could work well even if the document quality degrades. This works well irrespective of the document quality. It can easily adaptable to different environments since computational complexity is very less.

Because of these problems, we devise a new framework, which employs the spatial approach for identifying Tamil, Roman, Hindi scripts and Numerals from Multilingual document images. The proposed technique identifies the script using the first character of a word. Script or Language identification has been attempted in this work at character level. A Rule based classifier is then used to classify the proper script of the characters. This system can successfully act as a plug in to route the documents automatically to OCR engine.

3. SCRIPT RECOGNITION ARCHITECTURE

Classification of script of word images in multilingual documents (Tamil, English, Hindi and Numerals) has been reported in this paper. The proposed system identifies scripts by using the first character of each word in the multilingual document image. A rule based Classifier is used to classify the script at character level. Initially, characters of the word images are segmented. After extracting its features they are represented by vectors. A rule-based classifier is then used to classify the language of a character from its vectors. Figure 2 depicts the architecture of script identification

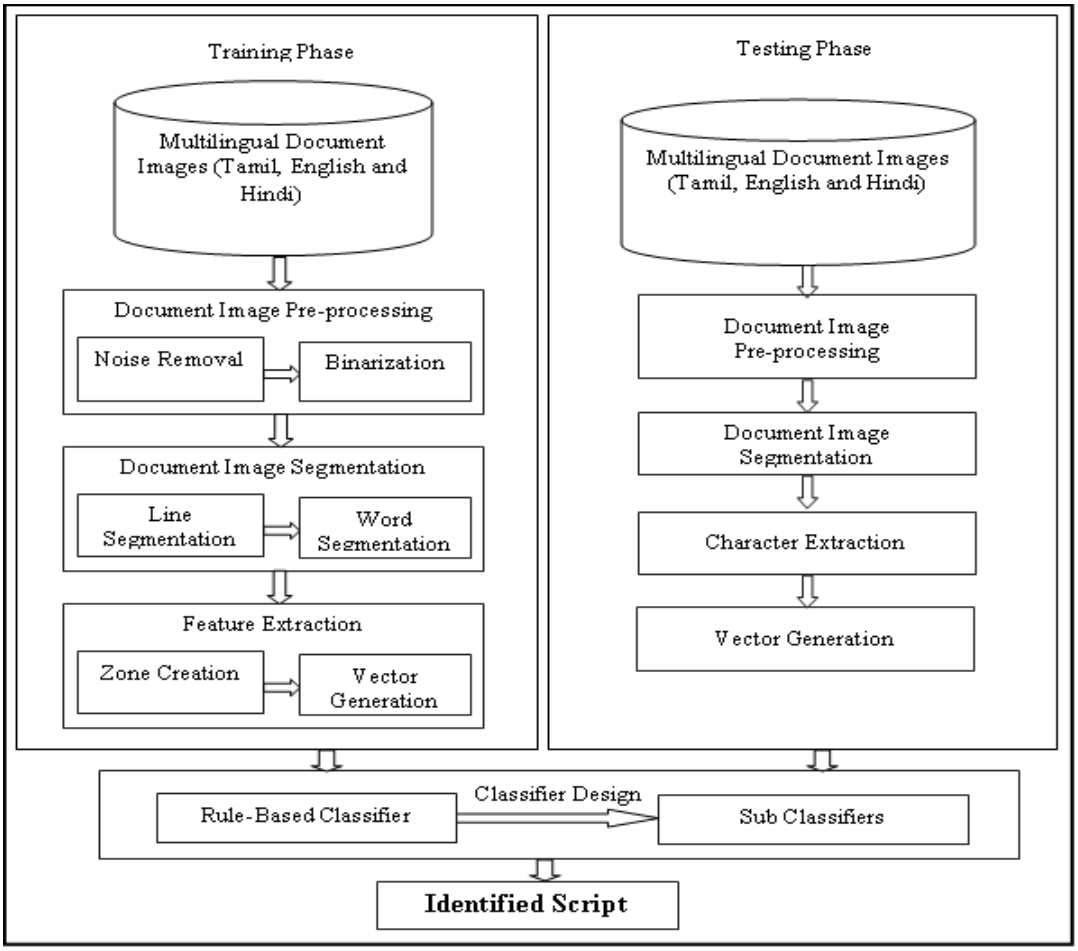

Figure 2. Script Identification Architectur

The input is a gray scale image obtained by scanning the Multi lingual newspapers. The input document is assumed to contain text and thus free from graphics, figures, maps and tables. Then the scanned images pass through the two phases such as Training Phase and Testing Phase.

3.1 Training Phase of Script Identification

Training Phase acquires multi lingual document image corpus (mixture of Tamil, English and Hindi words) constituting various font faces and sizes by scanning the hard copy of magazines in order to adapt the variations. This phase includes preprocessing of document images, word and character image segmentation, Tetra Bit generation and Classification technique.

3.1.1 Preprocessing

Any document images often suffer from noise of different sizes together with those small-connected components. All documents are scanned using HP Scanner at 300 DPI, which usually yields a low noise and good quality document image. So the document images can be binarized directly. But most of the document images are affected by salt and pepper noise as illustrated in the following figure 3.

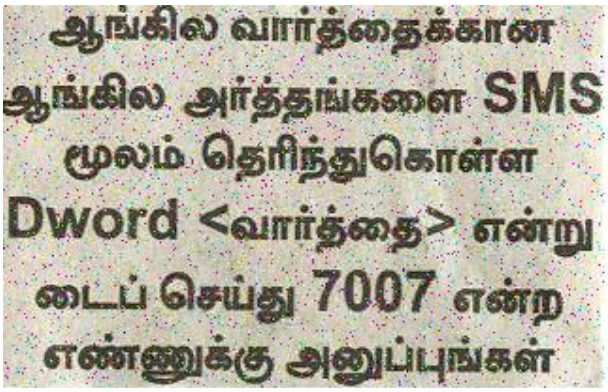

Figure 3 Noisy Document Image

Noise spikes are normally significantly brighter or darker than their neighboring pixels. Center weighted median filter [Lu et al] is used to keep the shape of the character stroke much better. Binarization uses Otsu Global Thresholding method which is used to convert gray scale images in to binary images. The most common method is to select a proper threshold for the image and then convert all the intensity values above the threshold intensity to one intensity value representing either "black" or "white" value. All intensity values below a threshold are converted to one intensity level and intensities higher than this threshold are converted to the other chosen intensity.

3.1.2 Segmentation

After pre-processing, the noise free image is passed to the segmentation phase, where the image is decomposed into words. To segment the document image into several text lines, horizontal projection profile is used which is 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.

3.1.3 Tetra Bit Generation

Once the characters are extracted from words, spaces on both the sides of the characters are trimmed to extract the exact black pixel density. Next, total character has been divided into nine zones and black tone & white tones of every zone have been grabbed. Applying two horizontal and two vertical bisections over the character produces tetra zones. The character image of I*J pixels is divided into M*N grids as shown in figure 4.

1	2	3
4	5	6
7	8	9

Figure 4. Tetra zone

Horizontal projections over the characters comprising of ascender, middle and descender zone has been reported in earlier works. Vertical projection is required to analyze further, the spatial spread of every character and we resulted in a 3*3 grid. This is illustrated for a sample Tamil and English character in figure 5.

Figure 5. Grid Formation for Sample characters

Bits for every zone have been generated based on its black pixel or white pixel density in that particular zone. Generation of bits for every zone has been represented in the following algorithm.

1. Repeat steps 2 to 4 for every zone of a character c.
2. Along the height h and width w zone n, traverse the zone n along x direction, (x=0…w) for every y. (y=0...h).
3. For every occurrence of black pixel along the traversal, increment the black pixel counter bpc by1.
4. For every occurrence of white pixel along the traversal, increment the white pixel counter wpc by 1.
5. Set the value of zone n to 1, if the ratio of wpc over the total number of black and white pixels in zone n exceeds the threshold. (1- represents the domination of white pixels over black pixels)
6. Set the value of zone n to 0, if the ratio of bpc over the total number of black and white pixels in zone n exceeds the threshold. (0- represents the domination of black pixels over white pixels).

Vectors (Tetra bits) generated for a set of sample English and Tamil characters have been represented in table.1

Table 1 Sample Vectors

Zones (Binary form)									
C	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***
⊔	0	1	0	0	1	0	0	0	0
ஒ	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	1	1
4	1	1	0	1	1	0	1	1	0

3.1.4 Classification

The task of discrimination can be carried out successfully by a supervised learner, which tries to predict the value of the function for any valid input object after seeing a number of training examples (i.e. pairs of input and target output). The proposed technique uses rule based classifier to discriminate different scripts.

Here, classification decisions are made by using any of the following three algorithms

- Rule based classifier
- Horizontal transition
- Headline Analysis

Headline Analysis

Since, Headline would be available for most of the Hindi words occupying the top three zones and this act as a discriminating factor for Hindi and Tamil script. In order to discriminate the English characters who have headlines from Hindi, a vertical line associated with the headline in the centroid area or right zone have been analyzed to differentiate Hindi script. Headline analysis algorithm is explained below:

Algorithm

1. Consider the upper three zones of the character
2. i) Initially the no of pixels in a row r=0 and t is the total width of that character
   ii) Traverse each and every pixel in that row and count number of every black pixels and increment r value
   iii) Repeat step (ii) for all rows in that zone and find the r which has maximum number of black pixels in that zone.
3. Script can be identified by comparing r and t values. If the character has continuous pixels equal to its width then it is Hindi else it is Tamil.
   Similarly vertical line analysis is used to differentiate English and Hindi scripts.

Rule Based Classifier

To discriminate Tamil and English scripts, first the total number of zones having less dominance of black pixels is identified from the generated document vectors and patterns are generated from it.

Patters are three bit values. Tetra bit values generated in the above section gets transformed into a tri-digit pattern to act as an input for three feature classes C1, C2 and C3. A subsequent decision has been brought as output for these input objects. For example, If the training vector is less dominant in 2nd , 5th and 8th zone, tri digit pattern would be “258” and if the training vector is less dominant in 4th zone, tri digit pattern would be “004”.

Since decision trees are large and difficult to interpret, Rule-based classifiers are used which uses only IF-THEN rules. This classifier is designed based on the result of pattern analysis.Based on the training set of characters, following rules have been framed to classify the vectors generated by zones as English or Tamil.

1. Set Lang attribute to null. Lang =””.
2. Receive the vectors for all the nine zones.(z1….z9)

3. Traverse the following rules (a..e) with the nine vectors to classify the language of a character:
   a) if ((z2∧z5)∨ (z5∧z8)∨ (z2∧z8)∨ (z2∧z5∧z8) equals '1') Lang = "E"
   b) else if (z4 equals '1') Lang = "E"
   c) else if ((z2∧z3) equals '1') Lang = "E"
   d) else if ((z8∧z9) equals '1') Lang ="E"
   e)else if (z4 equals '1') then Lang = "E"
   f) else if (z3 equals '1') then Lang="T"
   g)else if ((z4 and z6) equals '1') Lang = "E"
   h) else if ((z1 and z2) equals '1') Lang = "T"
   i) else if ((z4 and z6 and z9) equals '1')
      Lang = "T"
   j) else if ((z3 and z5 and z9) equals '1')
      Lang = "E"
   k) else if ((z1 and z2nand z4 and z5 and z7 and z8)
      equals '1') Lang="N"
      else
      Horizontal Transition is applied.
4. Attribute Lang represents the classified language. (E – English, T- Tamil and N - Numeral).

This classifier is used to differentiate most of the Tamil and English characters except few such as ஐ, B,etc…because of distribution in all zones. Since vectors are same for these types of characters, sub classifier is required. If number of zones having less dominance of black pixels=0, then another sub classifier, horizontal transition is applied directly since the rule based classifier cannot differentiate some of Tamil and English words having same vectors.

Horizontal Transition

Sub classifier which is used to classify some of tamil, english scripts and numerals which are distributed in all nine zones or having same density structures is called as Horizontal Transition. Horizontal Transition rate detects the horizontal disposition rate by accounting every black to white and white to black dispositions over the middle zone. It was also observed that from the training samples that the transition rate of middle zones is used to classify both tamil, roman scripts and numerals. For example the transition rate of English characters is less than the transition rate of Tamil Characters.

Algorithm

1. Consider the middle zone of the character
2. i) Initially the no of horizontal transition h=0
   ii) Traverse each and every pixel in that row and count number of every black to white and white to black dispositions and increment h value for each transition
   iii) Repeat step (ii) for all rows in that zone.
   iv)Make decision based on the h value. If h>2, Then the script is Tamil else it is English.

3.2 Testing Phase of Script Recognition

In this paper, scripts are reported for word images using their initial character. They are classified sharply based on the density of black pixels in every zone. Accuracy as well as Time factors are better in this approach. Because of the differences in character densities i.e. spatial spread per unit

area/ per zone, features of the characters of both the language gets varied. Like previous methods, this does not concentrate the spatial spread as ascender, middle and descender zones. This reveals the identity of different characters through the accumulation of black pixel in every zone as either dense or sparse.

When a testing sample is provided to the system, words and characters are segmented, tetra zones are formed and in turn features of every zone gets extracted. Features are transformed into nine bits as explained above. Nine bits goes through the above rule based classifier to classify the language.

For instance, English word represented in figure 6 is properly classified as English. Initially, the first character gets segmented and the bits generated for that character is 000101101.

TAMIL

Figure 6 Sample English word

For instance, Tamil word represented in figure 7 is properly classified as Tamil. Initially, the first character gets segmented and the bits generated for that character is 000000000. Here the classification decision is made as "HT". Horizontal transition procedure clearly discriminates this as Tamil.

Figure 7 Sample Tamil word

For instance, Numbers represented in figure 8 properly classified as Numeral. Initially, the first character gets segmented and the bits generated for the first character is 110110110. Here the classification decision discriminates as Numeral. For instance, the word  is identified as Hindi. Initially, the first character gets segmented and the bits generated for the first character is 000000110. Here the classification decision discriminates as hindi because of its headline and vertical line structure.

40

Figure 8. Sample Numerals

4. EXPERIMENTAL RESULTS

This system has been implemented using Java Advanced Imaging. Inputs were obtained from various magazines, newspapers and other such documents containing variable font sizes. The training and test patterns have been taken, consisting of variable English and Tamil words. Figure 9 shows a sample multilingual image. Figure 10 shows the identified Hindi script. Figure 11, and 13 shows the sample outputs of Numerals, Tamil and English Script identification. Figure 14 and 15 shows the noisy document image and clean image. Figure 16 and 17 shows the sample Tamil and English script identified from the noisy image.

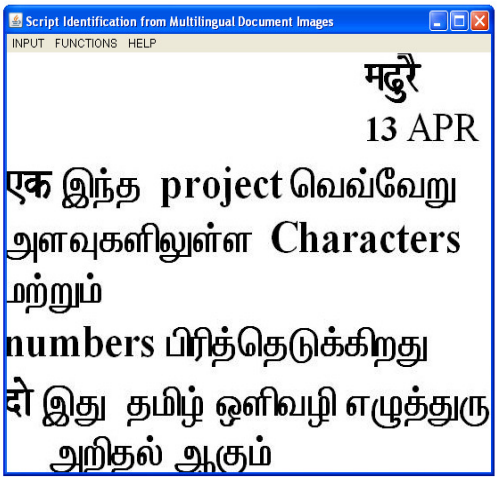

Figure 9. Sample Image

Figure 10 Identified Hindi Script

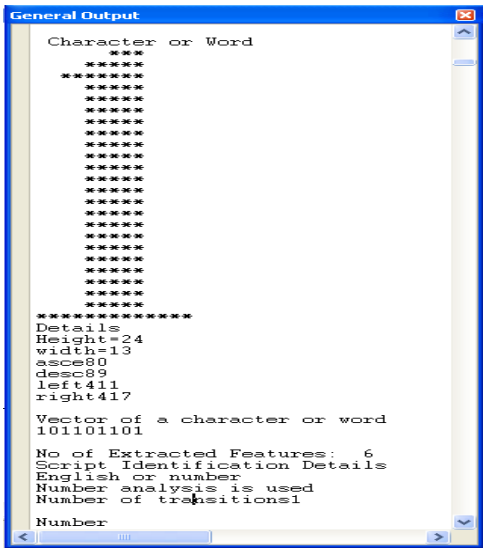

Figure 11 Numeral Script

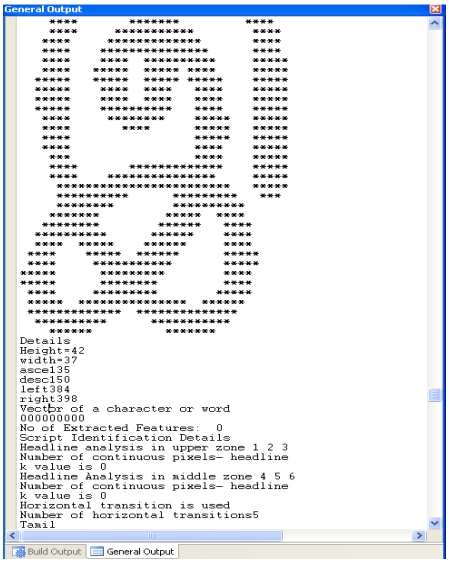

Figure 12. Tamil Script

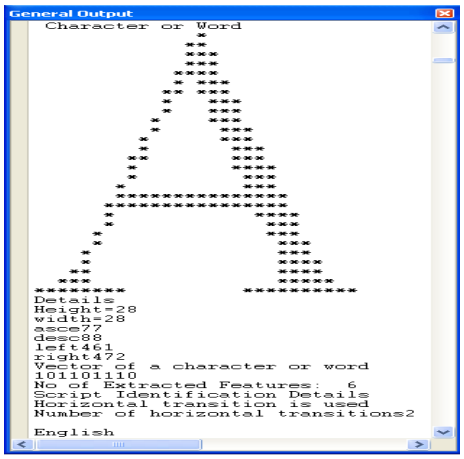

Figure 13. Identified English Script

Figure 14. Noisy document Image

Figure 15. Document Image without Noise

Figure 16. Identified Tamil Script

Figure 17. Identified English Script

5. PERFORMANCE ANALYSIS

Quantitative performance of the classification scheme is computed over a sample document image collection of more than 250 document images. Considerable number of words is used for testing in document images. During the classification of words, priority is given to bilingual document (Tamil and English) words and numerals mixed in the same document image. This system performs well for document images even from quality of 200 dpi when comparing to other systems. This is not constrained to a single image type and could read images of types BMP, JPEG and TIFF. Further, this system is well normalized to act upon different font sizes.

We computed recall and precision for the classified words. Percentage of relevant words that are perfectly classified as either of the script (Tamil or English) is computed It is found that a promising precision and recall measures gets registered in our experiment.

Figure 18 shows the Precision Measure of script recognizer over Tamil and English word images of different document images, each containing an average of 150 words. Figure 19 shows the Recall measure of Script Recognizer over Tamil and English word images of different document images.

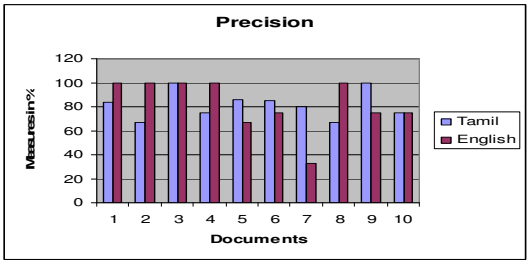

Figure 18. Precision Measures

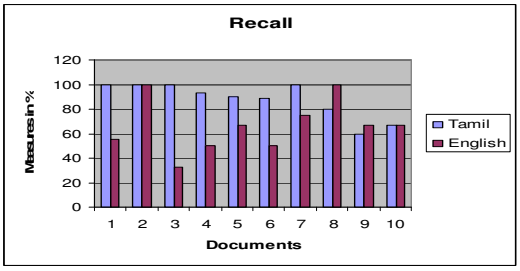

Figure 19. Recall Measures

Table 2 and 3 shows the Precision and Recall measures of Script recognizer over Tamil, English, Hindi words and numeral images of different document images, each containing an average of 150 words. Figure 20 and 21 shows the Precision and Recall values graphically.

Table 2. Precision Measures of Multi-Scripts

Images	Tamil	English	Hindi	Numerals
1	100	100	100	100
2	93.8	95	97	100
3	95.2	93.3	75	94
4	100	93	100	100
5	100	77.8	85.7	100
6	97.6	90	96	98
7	97.7	83	91	100
8	97.4	91	95	100
9	98.1	85	89.5	100

Table 3. Recall Measures of Multi-Scripts

Images	Tamil	English	Hindi	Numerals
1	100	100	100	100
2	93.3	100	80	69.2
3	90.9	93.3	90	100
4	100	100	100	88.9
5	90	100	100	83.3
6	94.5	68.8	100	84.6
7	92.7	85	95	85.7
8	87	80.4	100	100
9	88.4	84.1	100	86.7

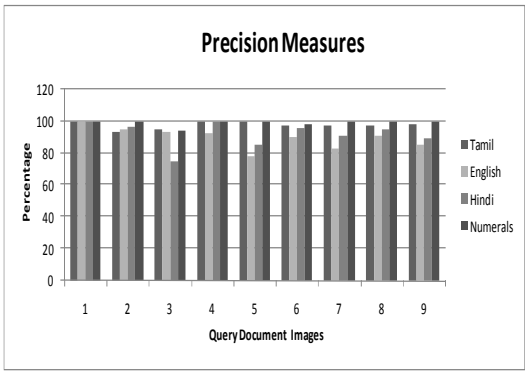

Figure 20 Precision of Multi-Scripts

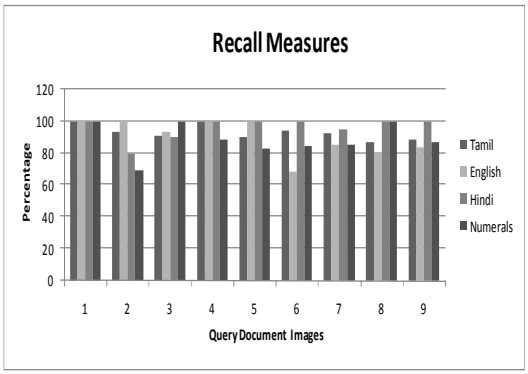

Figure 21. Recall of Multi-Scripts

Table 4 shows the confusion matrix of script recognizer over Tamil, English, Hindi words and numerals of different document images before noise removal and Table 5 tabulates the results after noise removal.

Table 4 Accuracy before Noise Removal

Script	No. of words	Tamil	English	Hindi	Numeral
Tamil	125	108	6	11	0
English	50	4	46	0	0
Hindi	23	9	0	16	0
Numerals	17	0	5	0	12

Table 5. Accuracy after Noise Removal

Script	No. of words	Tamil	English	Hindi	Numeral
Tamil	125	118	1	6	0
English	50	1	49	0	0
Hindi	23	2	0	21	0
Numerals	17	0	2	0	15

The proposed work produces classification accuracy of 97.5% for Tamil , 97% for English, 96% for Hindi and 94% for numerals. It misclassifies some of Tamil words as English. For example the letter l of Tamil and L of English has same structure. The density distribution of these two letters is same in all zones and hence the vectors of these two letters are also same. Similarly it misclassifies some of English words as Tamil or Numeral and some numeral as English because of same number of horizontal transitions.

6. CONCLUSION AND FUTURE WORK

This paper reports an identification method that detects scripts and numerals at word level by using the first character of a word. This method identifies the script through Zone digitization technique. Document images are binarized, words and characters are segmented from the images. Features of characters such as its shape, density and frequency are extracted through zone segmentation and the extracted ones get transformed into digital values. Script is then determined according to the classification imported by the formation of the rules for digital values. Results from experiments, simulations, and human vision encounter that the proposed technique is promising and easy to extend for other languages. Since this system can act as a plug-in, this can be embedded with OCR prior to the recognition stage.

References

[1] Anoop M. and Anil K.J., (2004) " Online Handwritten Script Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.26, No.1, pp.124-130.

[2] Busch A., Boles W.W. and Sridharan S. (2005), 'Texture for Script Identification', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.27, No.11, pp.1720-1732.

[3] Cheng. J, Ping. X, Zhou G and Yang. Y(2006), " Script Identification of Document Image Analysis", Proceedings of the First International Conference on Innovative Computing, Information and Control.

[4] Dhandra B.V. and Mallikarjun. H (2007), 'Global and Local Features Based Handwritten Text Words and Numerals Script Identification', Proceedings of International Conference on Computational Intelligence and Multimedia Applications, pp. 471-475.

[5] Dhandra B.V., Mallikarjun H., Ravindra H. and Malemath.V.S. (2007), 'Word Level Script Identification in Bilingual Documents through Discriminating Features', Proceedings of International Conference on Signal processing, Communications and Networking, pp. 630-635.

[6] Dhandra B.V., Mallikarjun H., Ravindra H. and Malemath V.S. (2006b), 'Word-wise Script Identification based on Morphological Reconstruction in Printed Bilingual Documents,' Proceedings of IET International conference on Vision Information Engineering, pp 389-393.

[7] Dhandra B.V., Mallikarjun H., Ravindra H. and Malemath V.S. (2006c), 'Word-wise Script Identification from Bilingual Documents based on Morphological Reconstruction,' Proceedings of First IEEE International Conference on Digital Information Management, pp. 389-394.

[8] Dhandra B.V., Nagabhushan P., Mallikarjun H., Ravindra H. and Malemath V.S. (2006a), 'Script Identification Based On Morphological Reconstruction in Document Images', Proceedings of the Eighteenth International Conference on Pattern Recognition, pp. 950-953.

[9] Dhanya D., Ramakrishnan A.G. and Peeta Basa P. (2002), 'Script Identification In Printe Bilingual Documents,' Sadhana, Vol. 27, Part-1, pp. 73-82.

[10] Dunning. T, "Statistical Identification of Language," Technical report, Computing Research Laboratory, New Mexico State University, 1994.

[11] He.Z , You. X, Tang Y.Y and Xue. Y (2006), "Texture Image Retrieval Using Novel Non Separable Filter Banks Based on Centrally Symmetric Matrices", The 18th International Conference on Pattern Recognition (ICPR'06).

[12] Hochberg J., Kerns L., Kelly P. and Thomas T. (1995), 'Automatic Script Identification from Images Using Cluster-Based Templates', Proceedings of International Conference on Document Analysis and Recognition , pp. 378-381.

[13] Hochberg J., Kelly P., Thomas T. and Kerns L. (1997), '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.

[14] Joshi G., Saurabh G. and Jayanthi S. (2006), 'Script Identification from Indian Documents', Proceedings of the Seventh IAPR workshop on Document Analysis Systems, LNCS 3872, pp. 255-267.

[15] Liu Y.H., Lin C. and Chang F. (2005), 'Language Identification of Character Images using Machine Learning techniques', Proceedings of the Eighth International conference on Document Analysis and Recognition, pp. 630-634.

[16] Lu.S and Tan.C.L., (2006) " Script and Language Identification in Degraded and Distorted Document Images", Twenty-First National Conference on Artificial Intelligence.

[17] Lu S. and Tan C.L. (2008), 'Script and Language Identification in Noisy and Degraded Document Images', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, No. 1, pp. 14-24.

[18] Ma H. and Doermann D. (2003),'Gabor Filter Based Multi-class Classifier for Scanned Document Images', Proceedings of the Seventh International Conference on Document Analysis and Recognition, pp. 968-972.

[19] Padma. M.C and Vijaya. P.A. (2008), " Language Identification of Kannada, Hindi and English Text Words through Visual Discriminating features", International Journal of Computational intelligence systems, Vol. 1, No. 2, pp.116 – 126.

[20] Pal U. and Chaudhuri B.B. (1999), 'Script Line Separation from Indian Multi-Script Documents,' Proceedings of the International Conference on Document Analysis and Recognition, pp.406- 409.

[21] Pal U. and Chaudhuri B.B. (2001), 'Automatic identification of English, Chinese, Arabic, Devnagari and Bangla script line', Proceedings of the International Conference on Document Analysis and Recognition, pp. 790-794.

[22] Pal U. and Chaudhuri B.B. (1997), 'Automatic separation of words in multi-lingual multiscript Indian documents', Proceedings of the International Conference on Document Analysis and Recognition, pp. 576-579.

[23] Pal U., Sinha S. and Chaudhuri B.B. (2003), 'Word-Wise Script Identification From A Document Containing English, Devnagari And Telugu Text', Proceedings of the Document Analysis and Recognition, pp. 213-220.

[24] Pati P.B., Sabari Raju S., Pati N. and Ramakrishnan A.G. (2004), 'Gabor filters for document analysis in Indian Bilingual Documents', Proceedings of the International Conference on Intelligent Sensing and Information Processing, pp.123-126.

[25] Peake G.S. and Tan T.N. (1997), 'Script and Language Identification from Document Images', Proceedings of the Eighth British Machine Vision Conference, Vol. 2, pp. 230-233.

[26] Qiao Y.L., Lu .Z.M and Sun S.H., (2006), "Gabor Filter based Text Extraction from Digital Document Images", Proceedings of the International Conference.

[27] Spitz A.L. (1994), 'Script and Language Determination from Document Images', Proceedings of the Third Annual Symposium of Document Analysis and Information Retrieval, pp. 229-235.

[28] Spitz A.L. (1997), 'Determination of script, language content of document images', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.3, pp. 235-245.

[29] Tan C.L., Leong P.Y. and He S. (1999), 'Language Identification in Multilingual documents', Proceedings of the International Symposium on Intelligent Multimedia and Distance Education

[30] Tan T.N. (1998), 'Rotation Invariant Texture features and their use in Automatic Script Identification', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.20, pp. 751-756.

[31] Wood. X. Yao. K.Krishnamurthi and Dang "Language Identification For Printed Text Independent Of Segmentation," Proc. Of Int'l. Conf. On Image Processing, Pp. 428-431, 1995.