International Journal of Computational Intelligence Systems, Vol.1, No. 2 (May, 2008), 116–126

LANGUAGE IDENTIFICATION OF KANNADA, HINDI AND ENGLISH TEXT WORDS THROUGH VISUAL DISCRIMINATING FEATURES

M.C. PADMA
Assistant Professor, Dept. of Computer Science & Engineering.
PES College of Engineering, Mandya-571401
Karnataka, India
Email: padmapes@gmail.com

DR. P.A. VIJAYA
Professor, Dept. of Electronics & Communication Engineering.
Malnad College of Engineering
Hassan-573201
Karnataka, India
Email: pavmkv@gmail.com

Received:21-09-2007
Revised:29-10-2008

In a multilingual country like India, a document may contain text words in more than one language. For a multilingual environment, multi lingual Optical Character Recognition (OCR) system is needed to read the multilingual documents. So, it is necessary to identify different language regions of the document before feeding the document to the OCRs of individual language. The objective of this paper is to propose visual clues based procedure to identify Kannada, Hindi and English text portions of the Indian multilingual document.

Keywords: **Document mage Processing, Multi-lingual Document, Language Identification, Horizontal Lines, Vertical Lines, Feature Extraction.**

1. Introduction

Language identification is an important topic in pattern recognition and image processing based automatic document analysis and recognition. The objective of language identification is to translate human identifiable documents to machine identifiable codes [1]. The world we live in, is getting increasingly interconnected, electronic libraries have become more pervasive [2] and at the same time increasingly automated including the task of presenting a text in any language as automatically translated text in any other language. Identification of the language in a document image is of primary importance for selection of a specific OCR system processing multi lingual documents [3]. Language identification may seem to be an elementary and simple issue for humans in the real world, but it is difficult for a machine, primarily because different scripts (a script could be a common medium for different languages) are made up of different shaped patterns to produce different character sets [4].

OCR is of special significance for a multi-lingual country like India, where the text portion of the document usually contains information in more than one language. A document containing text information in more than one language is called a multilingual document. For such type of multilingual documents, it is very essential to identify the text language portion of the document, before the analysis of the contents could be made. Although a great number of OCR techniques have been developed over years [5, 6], almost all existing works on OCR make an important implicit assumption that the language of the document to be processed is known beforehand [2]. Individual OCR tools have been developed to deal best with only one

specific language [7]. In an automated environment such document processing systems relying on OCR would clearly need human intervention to select the appropriate OCR package, which is certainly inefficient, undesirable and impractical [4]. A pre-OCR language identification system would enable the correct OCR system to be selected in order to achieve the best character interpretation of the document [7]. This area has not been very widely researched to date, despite its growing importance to the document image processing community and the progression towards the "paperless office" [7]. Keeping this drawback in mind, in this paper an attempt has been made to solve a more foundation problem of language identification of a text from a multilingual document, before its contents are automatically read.

Language identification is one of the vision application problems. Generally human system identifies the language in a document using some visible characteristic features such as texture, horizontal lines, vertical lines, which are visually perceivable and appeal to visual sensation. This human visual perception capability has been the motivator for the development of the proposed system. With this context, in this paper, an attempt has been made to simulate the human visual system, to identify the type of the language based on visual clues, without reading the contents of the document.

In a multi-lingual country like India (India has 18 regional languages derived from 12 different scripts; a script could be a common medium for different languages [8]), documents like bus reservation forms, passport application forms, examination question papers, bank-challen, language translation books and money-order forms may contain text words in more than one language forms. For such an environment, multi lingual OCR system is needed to read the multilingual documents. 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. In this direction, multi lingual document segmentation has strong direct application potential, especially in a multilingual country like India.

In the context of Indian languages, some amount of research work has been reported [2, 4, 8, 9]. Further there is a growing demand for automatically processing the documents in every state in India including Karnataka. Under the three language formulae [8], adopted by most of the Indian states, the document in a state may be printed in its respective official regional language, the national language Hindi and also in English. Accordingly, a document produced in Karnataka, a state in India, may be printed in its official regional language Kannada, national language Hindi and also in English. For such an environment, multi-lingual OCR system is needed to read the multilingual documents. To make a multilingual-OCR system successful, it is necessary to develop the multilingual-OCR system that would work in two stages: (i) Identification and separation of different language portions of the document and (ii) Feeding of individual language regions to appropriate OCR system. In this paper, we focus on the first stage of the multilingual-OCR system and present procedures for identification and separation of Kannada, Hindi and English text portions of the multilingual document produced at Karnataka, an Indian state. In the present case, it could also be called as script or language identification, since the three languages Kannada, Hindi and English belong to three different scripts.

1.1. *Previous work*

From the literature survey, it has been revealed that some amount of work has been carried out in script/language identification. Peake and Tan [7] have proposed a method for automatic script and language identification from document images using multiple channel (Gabour) filters and gray level co-occurrence matrices for seven languages: Chinese, English, Greek, Korean, Malayalam, Persian and Russian. Tan [2] has developed rotation invariant texture feature extraction method for automatic script identification for six languages: Chinese, Greek, English, Russian, Persian and Malayalam. In the context of Indian languages, some amount of research work on script/language identification has been reported [8,10,11,13]. Pal and Choudhuri [8] have proposed an automatic technique of separating the text lines from 12 Indian scripts (English, Devanagari, Bangla, Gujarati, Kannada, Kashmiri, Malayalam, Oriya, Punjabi, Tamil, Telugu and Urdu) using ten triplets formed by grouping English and Devanagari with any one of the other scripts. Santanu Choudhuri, et al. [3] have proposed a method for identification of Indian languages by combining Gabour filter based technique and direction distance histogram

classifier considering Hindi, English, Malayalam, Bengali, Telugu and Urdu. Basavaraj Patil and Subbareddy [9] have developed a character script class identification system for machine printed bilingual documents in English and Kannada scripts using probabilistic neural network. Pal and Choudhuri [10] have proposed an automatic separation of Bangla, Devanagari and Roman words in multilingual multi-script Indian documents. Nagabhushan et.al. [13] have proposed a fuzzy statistical approach to Kannada vowel recognition based on invariant moments. Pal et. al. [12] have suggested a word-wise script identification model from a document containing English, Devanagari and Telugu text. Chanda and Pal [11] have proposed an automatic technique for word-wise identification of Devanagari, English and Urdu scripts from a single document. Spitz [18] has proposed a technique for distinguishing Han and Latin based scripts on the basis of spatial relationships of features related to the character structures. Pal et al. [19] have developed a script identification technique for Indian languages by employing new features based on water reservoir principle, contour tracing, jump discontinuity, left and right profile. Ramachandra et al. [20] have proposed a method based on rotation- invariant texture features using multichannel Gabor filter for identifying six (Bengali, Kannada, Malayalam, Oriya, Telugu and Marathi) Indian languages. Hochberg et al. [21] have presented a system that automatically identifies the script form using cluster-based templates. Gopal et al. [22] have presented a scheme to identify different Indian scripts through hierarchical classification which uses features extracted from the responses of a multi-channel log-Gabor filter. 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, but hardly few attempts focus on these three languages Kannada, Hindi and English followed in Karnataka, an Indian state.

In one of my earlier works [4], it is assumed that a given document should contain the text lines in one of the three languages Kannada, Hindi and English. In one of my previous papers [14], the results of detailed investigations were presented related to the study of the applicability of horizontal and vertical projections and segmentation methods to identify the language of a document considering specifically the three languages Kannada, Hindi and English. It is reasonably natural that the documents produced at the border regions of Karnataka may also be printed in the regional languages of the neighboring states like Telugu, Tamil, Malayalam and Urdu. The system [4] was unable to identify the text words for such documents having text words in Telugu, Tamil, Malayalam, Urdu languages and hence these text words were misclassified into any one among the three languages, whichever is nearer and similar in its visual appearance. For example, Telugu is misclassified as Kannada and Tamil is misclassified as English. If the document consists of text words in other than the anticipated languages, our previous algorithm fails to identify the type of the language by misclassifying the text words.

Keeping the drawback of the previous method [15] in mind, we have proposed a system that would more accurately identify and separate different language portions of Kannada, Hindi and English documents and also to classify the portions of the document in other than these three languages into a fourth class category - OTHERS, as our intension is to identify only Kannada, Hindi and English. The system identifies the three languages in four stages: in the first stage Hindi is identified, in the second stage Kannada is identified, in the third stage English is identified and in the fourth and the last stage, languages other than Kannada, Hindi and English are grouped into fourth class category OTHERS without identifying the type of that language as our main aim is to focus only on Kannada, Hindi and English languages.

This paper is organized as follows. Section 2 describes some discriminating features in the characters of Kannada, Hindi and English text words. In Section 3, two models proposed for identifying the three languages - Kannada, Hindi and English, have been discussed. The experimental details and the results obtained are presented in section 4. Conclusions are given in section 5.

2. Some Visual Discriminating Features of Kannada, Hindi and English Text Words

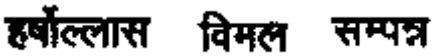
Feature extraction is an integral part of any recognition system. The aim of feature extraction is to describe the pattern by means of minimum number of features or attributes that are effective in discriminating pattern

classes [13]. The new algorithms presented in this paper are inspired by a simple observation that every script/language defines a finite set of text patterns, each having a distinct visual appearance [1]. The character shape descriptors take into account any feature that appears to be distinct for the language [1] and hence every language could be identified based on its visual discriminating features.

Presence and absence of the four discriminating features of Kannada, Hindi and English text words are given in Table-1.

2.1. *Some visual discriminating features of Hindi language*

In Hindi (Devanagari) language, many characters have a horizontal line at the upper part. This line is called sirorekha in Devanagari [8]. However, we shall call it 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 as a visual discriminating feature to recognize Hindi language. We can also observe that most of the Hindi characters have vertical line like structures. It could be seen that since two or more characters are connected together through their head-line portions, the width of the block is much larger than the height of the text line. Some typical Hindi words are given below:



2.2. *Some visual discriminating features of English language*

It has been found that a distinct characteristic of most of the English characters is the existence of vertical line-like structures [8] and uniform sized characters with each characters having only one component (except “i” and “j” in lower-case).

2.3. *Some visual discriminating features of Kannada language*

It could be seen that most of the Kannada characters have horizontal line like structures. Kannada character set has 50 basic characters, out of which the first 14 are vowels and the remaining characters are consonants [11]. A consonant combined with a vowel forms a modified compound character resulting in more than one component and is much larger in size than the corresponding basic character. It could be seen that a document in Kannada language is made up of collection of basic and compound characters resulting in equal and unequal sized characters [11] with some characters having more than one component, which could be expected to support in identifying the text words of Kannada language.

Some typical Kannada words are given below:

Table-1. Presence and absence of discriminating features of Kannada, Hindi and English text words.

(Yes means presence and No means absence of that feature.
F1: Horizontal lines; F2: Vertical lines; F3: Variable sized blocks; F4: Blocks with more than one component)

Discriminating Features. / Text words	F1	F2	F3	F4
Kannada	Yes	No	Yes	Yes
Hindi	Yes	Yes	Yes	No
English	No	Yes	No	No

2.4. *Zonalization of Kannada, Hindi and English Text Lines*

Pal and Choudhuri [8] have proposed that text lines of some Indian languages might be partitioned into three zones. In this paper, we have adopted the zonalization proposed by Pal and Choudhuri [8], which is useful in this method for feature extraction. A sample text line in English, Hindi and Kannada languages, partitioned/zonalized into three zones is shown in Figure-1. Related terminologies used in partitioning the text lines are summarized below:

An imaginary line where the first uppermost black pixels of characters of a text line lies is called an upper line. An imaginary line where the first lowermost black pixels of characters of a text line lies is called a lower line. An imaginary line, where the maximum number of uppermost black pixels of characters of a text line lies, is called a mean line. An imaginary line, where

maximum number of lowermost black pixels of characters of a text line lies, is called a base line. The upper zone covers the portion between the upper line and the mean line of a text line. The middle zone covers the portion between the mean line and the base line of a text line. The lower zone is the portion between the base line and the lower line. The height of the text line is defined as the normal distance between the upper line and the lower line. The X-height is defined as the distance between the mean line and the base line.

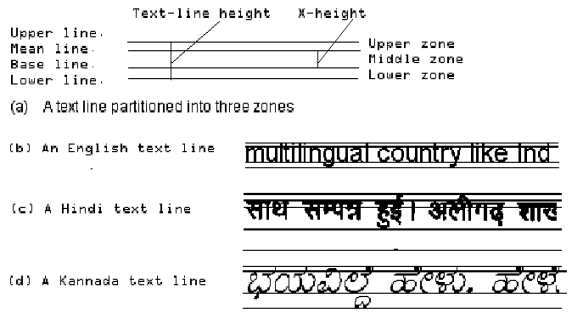


Figure 1. Partitioned text lines of English, Hindi and Kannada.

3. Proposed Models

The new model is developed based on the four visual features such as (i) horizontal lines (ii) vertical lines (iii) variable sized blocks and (iv) the number of components present in each block.

Two assumptions are made in our proposed model: 1. Input document is a machine printed document with standard font for Kannada, Hindi and English text lines. 2. Every text line must have at least four words and each text line may have different font sizes but the font and font size within a text line is same.

3.1. *Supportive Knowledge Base*

Knowledge base plays an important role in recognition of any pattern and knowledge base is a repository of derived information [14]. A supportive knowledge base is constructed for each specific class of patterns, which further helps during decision making to arrive at a conclusion. In the present method, the percentage of the presence of the four features for each text of the three languages Kannada, Hindi and English, are practically computed using sufficient data set. Based on the experimental results, a supportive knowledge base is constructed to store the percentage of the presence of the four visual features. The technique of obtaining the four visual features from the input image through experimentation is explained below:

Feature 1-Horizontal lines: In the binary image of each text line, if there are continuous one's in a row greater than the horizontal threshold value (Horizontal threshold value is calculated for each text line. Horizontal threshold value = 75% of the X-height of that text line), then such continuous one's are retained resulting in a horizontal line and if there are no continuous one's greater than the horizontal threshold value, then such one's are changed to zeroes. A component has a horizontal line-like structure, if a black run length (sequence of continuous one's) of that component is greater than the horizontal threshold value of that text line.

Feature 2- Vertical lines: In the binary image of each text line, if there are continuous one's in a column greater than the vertical threshold value (Vertical threshold value is computed for each text line. Vertical threshold value = X-height of that text line) then such continuous one's are retained resulting in a vertical line and if there are no continuous one's greater than the vertical threshold value, then such one's are changed to zeroes. A component has a vertical line-like structure, if a black run length (sequence of continuous one's) of that component is greater than vertical threshold of the text line.

Feature 3- Variable sized Blocks: The input binary image is segmented into several text lines and then each text line is segmented into several text words. Every text word is partitioned into three zones - upper zone, middle zone and lower zone as explained in section 2 to get the upper line and the lower line as two boundary lines for every text word. Then every text word is scanned vertically from upper line to reach the lower line of the respective text word without touching any black pixel, which results in a stream of variable sized blocks.

Feature 4- Blocks with more than one component: The number of components (a component is defined as an unconnected segment of the character) present within each block is computed using 8-neighbour connectivity [17].

The percentage of the spatial occurrence of all the four visual features for each of the three languages are practically computed through extensive experimentation

and stored in the knowledge base as given in Table-2 for later use during decision-making.

Table-2. Percentage of the presence of discriminating features of Kannada, Hindi and English text words. (F1: Horizontal lines; F2: Vertical lines; F3: Variable sized blocks; F4: Blocks with more than one component)

Discriminating Features / Text words	F1	F2	F3	F4
Kannada	65%	3%	60%	30%
Hindi	90%	80%	40%	5%
English	2%	80%	5%	5%

3.2. *Line-wise Identification Model*

In this section, we have proposed a line level identification model that would accurately identify and separate different language portions of Kannada, Hindi and English text lines of the input document and also group the portions of the document in other than these three languages into a separate class called OTHERS, without identifying the type of the language, as our intention is to identify texts in Kannada, Hindi and English languages only.

The proposed model is developed based on the discriminating features viz., horizontal lines, vertical lines, variable sized blocks (equal/unequal-sized blocks) and blocks with more than one component. These discriminating features are extracted from the processed document image and compared with the values that are stored in the knowledge base, to arrive at a decision regarding the type of the text language.
The different steps involved in the proposed model are as follows:

Input: 256x256 BMP file containing text lines in Kannada / Hindi / English / Telugu /Tamil / Urdu / Malayalam languages.
Output: Text line of Kannada, Hindi and English languages.
Step.1: The input document is preprocessed i.e., noise removed, smoothing done, skew compensated and binarized.
Step-2: Line segmentation: To segment the document image into several text lines, we use the valleys of the horizontal projection [14] 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 [14].
Step-3: Zonalization: Each text line is partitioned into three zones - upper zone, middle zone and lower zone as explained in Section 2.4.
Step-4: Block segmentation: From the zonalized text line, upper line and lower line is used as two boundary lines for every text line. Then every text line is scanned vertically from its upper line to reach its lower line without touching any black pixels to get a boundary line. Such characters enclosed within each boundary lines lead to a stream of blocks.
Step-5: Feature extraction:
Feature (i): Horizontal line detection: From the input image, the horizontal lines are obtained as explained in Section 3.1. Then the percentage of the presence of these horizontal lines for each text line is computed and compared with the stored values in the knowledge base.
Feature (ii): Vertical line detection: From the input image, the vertical lines are obtained as explained in Section 3.1. Then the percentage of the presence of these vertical lines for each text line is computed and compared with the stored values in the knowledge base.
Feature (iii): Variable Sized blocks: The size of the blocks of each text line is calculated by taking the ratio of width to height of each block. Then the percentage of equal and unequal sized blocks of each text line is calculated.
Feature (iv): Blocks with more than one component: The percentage of the number of components present in each block of every text line is computed.
Step-6: Decision making:
(i) Condition-1: If 90% of the horizontal lines on the mean line is greater than two times the X-height of the corresponding text line; if there are 80% of vertical lines in the middle zone and also if 70% of the blocks have width greater than two times the X-height, then such portion of the document is recognized as Hindi language.
(ii) Condition-2: If 65% of the horizontal lines on the mean line is greater than half of the X-height of the corresponding text line and if there are 40% of unequal sized blocks in a text line, then such portion of the document is recognized as Kannada language.

(iii) Condition-3: If there are 80% of vertical lines in the middle zone greater than half of the text line height and if 80% of the blocks are equal in size, then such portion of the document is recognized as English language.

(iv) Condition-4: If the output image does not belong to any of the above three classes, then such portion of the document is grouped into a separate class called OTHERS.

OTHERS class: If the referenced text lines do not contain the above mentioned discriminating features of the three languages Kannada, Hindi and English, then such text lines could be grouped into a separate category called OTHERS, without identifying the type of the language as our main aim is to identify and select only Kannada, Hindi and English text lines.

3.2.1 Limitations

One of the limitations of this algorithm is that if a text line contains words in more than one language, then such a text line is classified as the language type of the majority of words in that text line. Another limitation of this method is that if a text line contains some numerical figures in addition to the text words, then the entire text line is grouped into the language type of the remaining words, without identifying and grouping numerical figures into a separate category.

3.3. *Word-wise Identification Model*

In the previous section, we have suggested text language identification at the line level, with the assumption that the input document contains text lines in one and only one language. In this section, we suggest another model of language identification at word level, to overcome the limitations of the previous method. In a more practical scenario as found in majority of business applications, most of the documents such as language translation books, electric bills, application forms, marks cards and reservation forms, are comprised of several text lines, in which a text line itself could contain words in two or more languages or Hindu Arabic numerals representing some numerical information like date, year, count, age, height, weight, marks, roll-number, percentage, page-number, amount and so on. For such circumstances, we have suggested a modified method to identify the type of the language at word level using some visible discriminating features, which are relatively strong enough to identify the text language, without reading its contents. In this method, an attempt is made to develop a system that should accurately identify and separate text words of Kannada, Hindi and English language portions of the document; to identify and separate the portions of the text line containing Hindu Arabic numerals and also to group the portions of the document in other than these types into a separate category called OTHERS, avoiding misclassification.

The different stages involved in the word-level identification model are as follows:

Input: 256x256 BMP file containing text words in Kannada / Hindi / English / Hindu Arabic Numerals / Telugu /Tamil / Urdu / Malayalam languages.

Output: Text language type of portions of the input document in different languages.

Stage 1: Preprocessing: It is same as explained in Section 3.2.

Stage 2: Line segmentation: It is same as explained in Section 3.2.

Stage 3: Word segmentation: Every text line is segmented into words by finding the valleys of the vertical projection [14]. If the width of the valley is greater than the threshold value (Threshold value = two times the inter character gap), then a vertical line is drawn at the middle of the columns with zero values (Inter word gap). Using these vertical lines, words within a text line are segmented.

Stage 4: Word Partitioning: Each text word is partitioned into three zones - upper zone, middle zone and lower zone as explained in Section 2 to get upper line and lower line as two boundary lines for every text word.

Stage 5: Block segmentation: From the partitioned text word, upper and lower lines are used as two boundary lines for every text word. Then, every text word is scanned vertically from upper line to lower line without touching any black pixels to get a stream of blocks. Thus a block is defined as a rectangular section of the text words that has one or more characters with more than one component.

Stage 6: Block size evaluation: The size of each block of every text word is calculated by taking the ratio of width to height of each text word. Then the percentage of equal and unequal sized blocks of each text word is calculated.

Stage 7: Blocks having more than one component: The number of components (a connected component is one in which the pixels are aggregated by an 8-connected

points analysis) present within each block is calculated using 8-neighbour connectivity [17]. Then the percentage of the occurrence of blocks having more than one component is calculated.

Stage 8: Feature Extraction:

For each text word, the four features (i) Horizontal lines, (ii) Vertical lines, (iii) Variable Sized blocks and (iv) Blocks with more than one component are obtained as explained in step 5 of Section 3.2.

Stage 9: Decision making:

Level-1: If the length of the horizontal line on the mean line is greater than two times the X-height of the corresponding text word, if there are vertical lines in the middle zone, if the block has width greater than two times the X-height and also if the word/block contains only one component, then that text word is identified as a text word in Hindi language.

Level-2: If the length of the horizontal line on the mean line is equal to the x-height of the corresponding text word; if there are 70% of unequal sized blocks in the output image and also if 30% of the blocks contain more than one component, then that text word is recognized as a text word in Kannada language.

Level-3: If there are vertical lines in the middle zone greater than half of the text word height; if a text word contains 70% equal blocks in size and also if a text word contains 90% of the blocks having only one component, then that text word is identified as a text word in English language.

Level-4: From the segmented document image, the words occupying only upper zone and middle zone, having all the characters equal in height and enclosed between base line and upper line are extracted and such words are classified as Hindu Arabic numerals.

Level-5: If a text word does not belong to any of the above four levels, then such a text word does not belong to the above categories and hence it is grouped into a separate class called OTHERS.

It is suggested that the recognition be accomplished in five levels: in the first three levels Hindi, Kannada and English languages are identified respectively, in the fourth level, Hindu Arabic numerals are identified and finally, languages other than Kannada, Hindi and English are grouped into fourth-class category called OTHERS.

3.3.1 Limitations

One of the limitations of this method is that, if a text word contains alphanumeric values, then such text word is misclassified into any one of the five groups depending on the majority features present in that word.

4. Experimental Results and Discussion

The document images used were scanned by a flat-bed scanner at a resolution of 300 dpi. The size of the sample image considered was 256x256 pixels. The input documents used for experimentation are clipped from language translation books / newspapers / magazines / manuals / bank passbooks / money order forms / examination question papers.

4.1. *Experimental Results of Line-wise Identification Model*

We have tested our algorithm on 1000 text lines, out of which Kannada, Hindi and English text lines were 250, 200 and 500 respectively. We have also tested our algorithm on 50 text lines with Telugu, Tamil, Malayalam and Urdu languages. We have also considered the documents having each text line in different font sizes. The misclassification error of an input document is due to the presence of text lines with one or two words with different font sizes. Samples of Kannada, Hindi and English documents partitioned, horizontal and vertical lines detected are given in Figures 2, 3 and 4 respectively. A sample document in all the three languages Kannada, Hindi and English is given in Figure 5. Details of results obtained through extensive experimentation are depicted in Table 3.

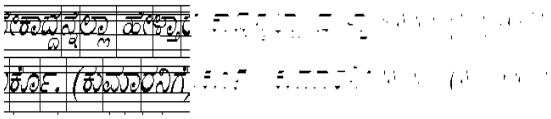


Figure 2. Sample output image of Kannada language.

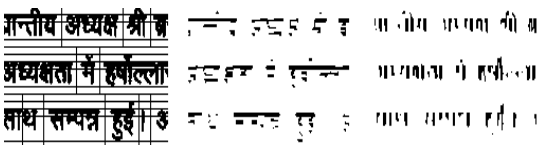


Figure 3. Sample output image of Hindi language.

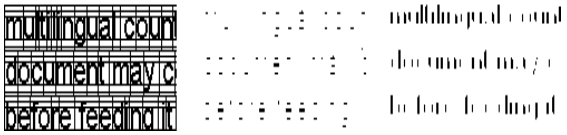


Figure 4. Sample output image of English language.

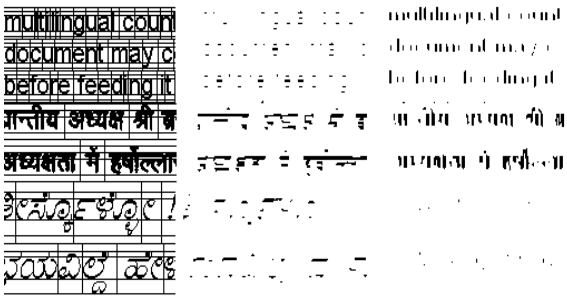


Figure 5. Sample output image containing all the three languages Kannada, Hindi and English.

Table 3. Percentage of experimental results of line-wise language identification model.

Output text line / Input Text line. (no. of text lines)	Kannada	Hindi	English	OTHERS
Kannada (250)	98.8%	0%	0.7%	0.5%
Hindi (200)	0%	99.4%	0.2%	0.4%
English.: Bookman Old Style (100)	0.4%	0%	99.2%	0.4%
English: OCR A Extended (100)	0.5%	0%	99.2%	0.3%
English: Times New Roman (100)	0.8%	0%	98.7%	0.5%
English: Arial (100)	1%	0%	98.5%	0.5%
English : Upper Case only (100)	0%	0%	100%	0%
OTHERS (50)	0.6%	0.2%	0.4%	98.8%

4.2. *Experimental Result of Word-wise Identification Model*

We have tested our algorithm on sample document images containing 1450 words. Each document image was a mixture of Kannada, Hindi and English text words with Hindu Arabic Numerals also. Out of these words, the number of Kannada, Hindi, English text words and Hindu Arabic numerals were 250, 200, 450 and 250 respectively. We have also experimentally tested the algorithm for 300 text words in Telugu, Tamil, Malayalam and Urdu languages. Clipped portion of a machine printed multilingual input document containing Kannada, Hindi and English text words and Hindu Arabic numerals is given in Figure 6. Sample of output images of English, Hindi and Kannada text words are shown in Figure 7. Figure 8 depict the sample output images of English, Hindi, Kannada text words and Hindu Arabic numerals showing horizontal lines, vertical lines and recognized text language type of each text word respectively. Details of results obtained through extensive experimentation are given in Table 4. We have noted that the wrongly recognized words are mostly of smaller length having one or two characters. Some special characters and other punctuation symbols are discarded.

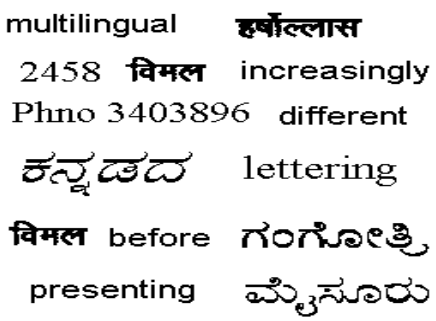


Figure 6. Sample input document containing Kannada, Hindi, English text words and Hindu Arabic numerals

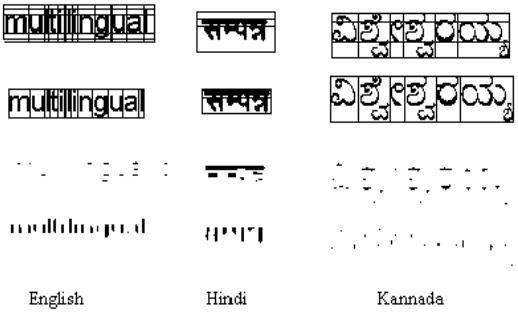
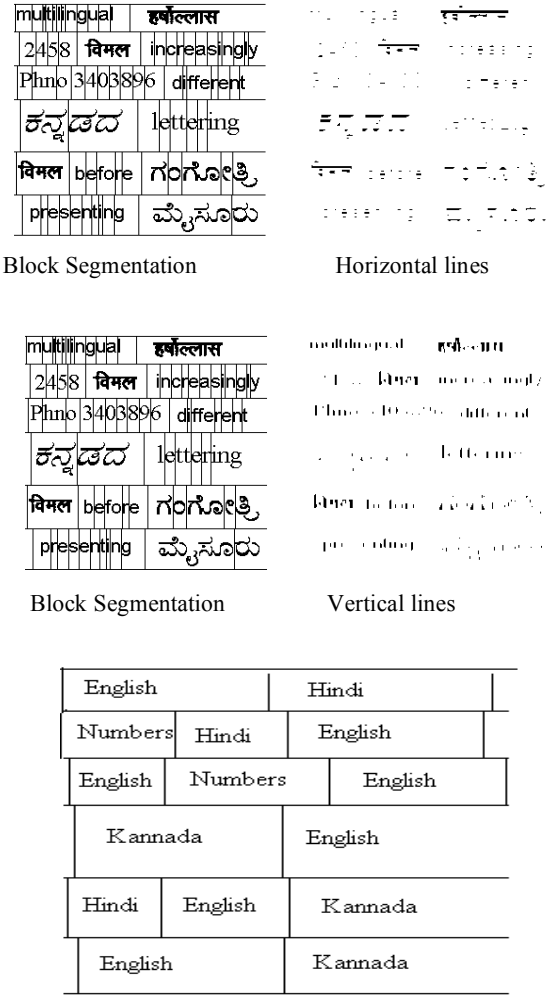


Figure 7. Sample output images of English, Hindi and Kannada text words.

Table 4. Percentage of experimental results of word-wise language identification.

Output text word / Input Text word		Kannada	Hindi	English	Hindu Arabic Numerals	OTHERS
Kannada		97.2%	0%	1.4%	0%	1.4%
Hindi		0%	98.5%	1.2%	0%	0.3%
English	Bookman Old Style	1%	0%	97.8%	0%	1.2%
	OCR A Extended	0.9%	0%	97.8%	0%	1.3%
	Times New Roman	1.2%	0%	97.2%	0%	1.6%
	Arial	1.1%	0%	97.3%	0%	1.6%
English Text: Upper Case		0%	0%	98.2%	1.8%	0%
Hindu Arabic Numerals		0%	0%	1.6%	98.4%	0%
OTHERS		1.8%	0%	1.0%	0%	97.2%



Recognized text language type of each text word.

Figure 8. Sample output images of English, Hindi, Kannada text words and Hindu Arabic numerals.

5. Conclusion

In this paper, we have presented 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. The methods help to accurately identify and separate different language portions of Kannada,

Hindi, English and Hindu Arabic numerals and also to group the portion of the document in other than these three languages into a separate class called OTHERS. The experimental results show that the two methods are effective and good enough to identify and separate the three language portions of the document, which further helps to feed individual language regions to specific OCR system. Our future work is to develop a system that can identify other Indian languages.

References:

1. P.Naghabhushan, Radhika M Pai, "Modified Region Decomposition Method and Optimal Depth Decision Tree in the Recognition of non-uniform sized characters – An Experimentation with Kannada Characters", *Journal of Pattern Recognition Letters,* 20, 1467-1475, (1999).
2. T.N.Tan, "Rotation Invariant Texture Features and their use in Automatic Script Identification", *IEEE Trans. Pattern Analysis and Machine Intelligence,* 20(7), 751-756, (1998).
3. Santanu Choudhury, Gaurav Harit, Shekar Madnani, R.B. Shet, "Identification of Scripts of Indian Languages by Combining Trainable Classifiers", ICVGIP 2000, Dec., 20-22, Bangalore, India.
4. M.C.Padma, P.Nagabhushan, "Horizontal and Vertical linear edge features as useful clues in the discrimination of multiligual (Kannada, Hindi and English) machine printed documents", *Proc. National Workshop on Computer Vision, Graphics and Image Processing (WVGIP),* Madhurai, 204-209, (2002).
5. U.Pal, B.B.Choudhuri, "OCR in Bangla:an Indo-Bangladeshi language", *IEEE, no.2,* 1051-4651, (1994).
6. U.Pal, B.B.Choudhuri, "An OCR system to read two Indian language scripts:Bangla and Devanagari(Hindi)", *Proc. 4th ICDAR,* Uhn, 18-20, (1997).
7. G.S. Peake, T.N.Tan, "Script and Language Identification from Document Images", *Proc. Eighth British Mach. Vision Conference.,* 2, 230-233, (1997).
8. U.Pal, B.B.Choudhuri, "Script Line Separation From Indian Multi-Script Documents", *Proc. 5th International Conference on Document Analysis and Recognition(IEEE Comput. Soc. Press),* 406-409, (1999).
9. S.Basvaraj Patil, N.V.Subba Reddy, "Character script class identification system using probabilistic neural network for multi-script multi lingual document processing", *Proc. National Conference on Document Analysis and Recognition*, Mandya, Karnataka, 1-8, (2001).
10. U.Pal B.B.Choudhuri, "Automatic Separation of Words in Multi Lingual multi Script Indian Documents", Proc. 4th *International Conference on Document Analysis and Recognition*, 576-579, (1997).
11. S.Chanda, U.Pal, "English, Devanagari and Urdu Text Identification", *Proc. International Conference on Document Analysis and Recognition,* 538-545, (2005).
12. U.Pal, S.Sinha, B.B.Choudhuri, "Word-wise script identification from a document containing English, Devanagari and Telugu text", *Proc. 2nd National Conference on Document Analysis and Recognition,* Karnataka, India, 213-220, (2003).
13. P.Nagabhushan, S.A.Angadi, B.S.Anami, "A Fuzzy Statistical Approachto Kannada Vowel Recognition based on Invariant Moments", *proc. 2nd National Conference, NCDAR*, Mandya, 275-285, (2003).
14. M.C.Padma, P.Nagabhushan, "Study of the Applicability of Horizontal and Vertical Projections and Segmentation in Language Identification of Kannada, Hindi and English Documents", *Proc. National Conference NCCIT*, Kilakarai, Tamilnadu, 93-102, (2001).
15. M.C.Padma, P.Nagabhushan, "Identification and separation of text words of Kannada, Hindi and English languages through discriminating features", *Proc. 2nd National Conference on Document Analysis and Recognition,* Mandya, Karnataka, 252-260, (2003).
16. U.Pal, B.B.Choudhuri, "Automatic Identification of English, Chinese, Arabic, Devanagari and Bangla Script Line", *Proc. 6th International Conference on Document Analysis and Recognition*, 790-794, (2001).
17. R.C.Gonzalez, R.E.Woods, *Digital Image Processing* Pearson Education Publications, India, 2002.
18. A.L.Spitz, "Determination of the Script and language Content of Document Images", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, 235-245, 1997.
19. U.Pal, S.Sinha, B.B.Choudhuri, "Multi-Script Line Identification from Indian Documents", *Proc. 7th International Conference on Document Analysis and Recognition (ICDAR 2003)* vol. 2, 880-884, 2003.
20. Ramachandra Manthalkar and P.K. Biswas, "An Automatic Script Identification Scheme for Indian Languages", NCC, 2002.
21. J.Hochberg, P.Kelly, T.Thomas, L.Kerns, "Automatic Script Identification from Document Images using Cluster –based Templates", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 176-181, 1997.
    Gopal Datt Joshi, Saurabh Garg, Jayanthi Sivaswamy, "Script Identification from Indian Documents", *DAS 2006, LNCS 3872*, 255-267, 2006.