International Journal of Soft Computing 1(4): 303-307, 2006

Enabling Efficient Information Retrieval from Tamil Document Images

[1]Abirami, S. and [2]D. Manjula
Department of Computer Science and Engineering, College of Engineering,
Anna University, Chennai, India

Abstract: This study performs an efficient Information Retrieval from scanned Tamil Document Images. Keywords are valuable indexing tools and if they can be identified at the image level, extensive computation during recognition will be avoided. Printed documents can be scanned to produce document images. Instead of converting entire document images into text equivalent, a Feature String algorithm has been designed to generate feature strings for the word images in Tamil documents based on the features extracted from it. During retrieval, the same features could be extracted from the user specified word and can be matched with the word images in the document .This yields a faster result even in a quality degraded document. This kind of Information Retrieval (Keyword Based Study) can be adapted in Digital Libraries which employs digitized documents instead of text processing

Key words: Tamil document image, feature extraction, keyword matching

INTRODUCTION

With the rapid development of computer technology, digital documents have become popular for storage and transmission, instead of the traditional paper documents. The most widespread format for these digital documents is the text in which the characters of the documents are represented by the machine-readable codes (e.g. ASCII codes).

Modern Technology has made it possible to produce process and store and transmit document images efficiently. In an attempt to move towards paperless office, large volumes of printed documents are digitized and stored as images in databases. Optical Character Recognition deals with the machine recognition of characters present in an input image obtained using scanning operation. It refers to the process by which scanned images are electronically processed and converted to an editable text. The need for Information Retrieval arises in the context of digitizing Tamil documents from the ancient and old era to the latest, which helps in sharing the data through the Internet.

Tamil language: Tamil is a South Indian language spoken widely in TamilNadu in India. Tamil has the longest unbroken literary tradition amongst the Dravidian languages. Tamil is inherited from Brahmi script. Tamil has 12 vowels and 18 consonants. These are combined with each other to yield 216 composite characters and 1 special character (aayutha ezhuthu) counting to a total of (12 + 18 + 216 + 1) 247 characters.

Vowels: Vowels in Tamil are otherwise called UyirEzhuthu and are of two types short (Kuril) and long (Nedil).

Consonants: Consonants are classified into three classes with 6 in each class and are called Vallinam, Idaiyinam and Mellinam.

Related work: Document Image Processing (DIP) technology is utilized to automatically convert the digital images of documents to a machine-readable text format using Optical Character Recognition (OCR). But it is not a cost effective and practical way to process huge number of paper documents because of inherent weakness in its recognition ability with document images of poor quality. Generally speaking, manual correction/proofing of the OCR results is usually unavoidable, which is typically not cost effective for transferring a huge amount of study documents to their text format.

In Recent years, there has been much interest in the area of Document Image Retrieval[1,2]. DIR is relevant to Document Image Processing (DIP), but there are some essential differences between them. In recent years, a number of attempts have been made by restudyers to avoid the use of character recognition for various document image retrieval applications. For example, Chen and Bloomberg[3,4] described a method for automatically

Corresponding Author: Abirami, S., Department of Computer Science and Engineering, College of Engineering, Anna University, Chennai, India

selecting sentences and key phrases to create a summary from an imaged document without any need for recognition of the characters in each word.

Sptiz described character shape codes for duplicate document detection[5], Information retrieval[6], word recognition[7] and document reconstruction[8], without resorting to character recognition. Yu and Tan[9,10] proposed a method for text retrieval from document images without the use of OCR. In their method, documents are segmented into character objects, whose image features are utilized to generate document vectors. Some approaches have been reported in the past years for studying keywords in handwritten and printed[11] documents. In word studying system[12], a weighted Hausdorff distance is used to measure the dissimilarity between word images.

In short, DIR and DIP[13] address different needs and have for directly retrieving information from document images, could achieve a relatively higher performance in terms of recall, precision and processing speed.

Certain OCR engines[14] are available for Tamil document images with relative constraints such as good quality input (minimum 300 dpi), post processing of documents (Manual corrections) etc. Enabling Keyword study on those OCR engines may not be effective.

To enable an efficient information retrieval (Keyword Based Study) from Tamil document images even of degraded quality, word level feature extraction has been done. In addition to that suitable keyword matching algorithm has been devised in this study.

MATERIALS AND METHODS

The block diagram to perform Information Retrieval in Tamil Document images is given in Fig. 1. They are Scanning the document, Preprocessing, Segmentation, Feature Extraction, Feature String Generation, User Query Processing, Keyword Matching from the document image.

Scanning the document: A printed document is chosen for scanning. It is placed over the scanner. Scanner software is invoked which scans the document. The document is sent to a program that saves it in preferably TIF, JPG or GIF format. The size of the input image is inherently restricted by the scope of the vision and by the scanner software length. Document Image Processing is not limited to a particular font size unlike other recognition systems.

Preprocessing: As a preprocessing stage, Image Binarization has been done. Binarization is a technique by which the gray scale images are converted to binary

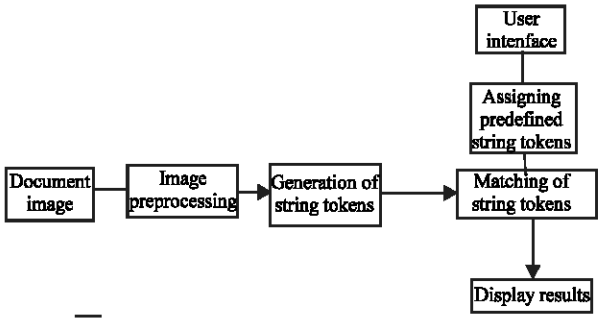

Fig. 1: Overall functional diagram

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. We used Otsu's threshold algorithm to binarize our gray scale image. In our convention, black represented a character (or noise) and white represented the foreground. Otsu's threshold makes an assumption that the histogram of the gray scale image has a bimodal distribution.

Word segmentation: After pre-processing, the noise free image is passed to the segmentation phase, where the image is decomposed into words. Algorithm for segmentation:

- The binarized image is checked for inter line spaces.
- If inter spaces are detected then the image is segmented into sets of paragraphs across the interline gap.
- Each line in the paragraph is scanned horizontally.

Line bounding: The line detection is responsible for finding the baseline of every text line. Lines of characters are detected by looking for interline spaces. These are characterized by a large number of non-black pixels in a row. The stroke lines for a word are drawn in such a way by identifying the black pixels, top boundary, bottom boundary, x-line and base line.

Word bounding: Once the line bounding has been over, with the help of vertical projection, word has been identified. Once the word has been identified, then a rectangular box bound it and its top left and bottom right coordinates are accounted for processing.

Feature extraction: Before extracting the features of the bounded word, an analysis of the Tamil characters have been made.

Analysis: Unique Tamil Characters in Tamil Language have been shown in Fig. 2. For our analysis we scanned the image at 150 dpi (dots per inch). We have reduced these above 125 characters into 73-character set as shown in Fig. 3 by categorizing according to their similarities.

Character which lies above the x-line are Ascenders and which lies below the bottom line are Descenders.

Characters are assigned into 4 sub categories:

- Letters which lies between Ascender zone and middle zone.
- Letters which lie in the Middle Zone.
- Letters which lies between Middle Zone and Descender Zone.
- Letters which covers all the Ascender, Middle and Descender Zone.

Figure 4 shows an example consists of letters characterized by Ascender-Middle-Descender zones.

In addition to the zone analysis of Tamil characters, we also analyzed the primitives of each character lying in that zone based on the occurrence of the following:

- vertical lines
- horizontal lines
- slopes
- circles.

Extraction: Once the word has been bounded, traversal starts from left end of the word to right end. A word is explicitly segmented, from the leftmost to the rightmost, into discrete entities. It traverses each and every pixel of the word. Each entity is denoted as a primitive here. It traverses the primitives sequentially.

Feature string generation: Each primitive is represented as a sequence of three tuples (A, L, C). Once the primitives gets segmented and features get extracted, its corresponding Ascender-Descender zone value has to be assigned to A. Number of horizontal and vertical lines gets assigned in L. Circles gets assigned to the value of C. As a result, feature string for the word image gets generated as a sequence of P's where

$$P = (p1, p2 \ldots pn) = (A1, L1, C1)\,(A2, L2, C2) \ldots (An, Ln, Cn)$$

where
A-ADA attribute
L-value returned by line detection algorithm
C-corresponds to dots and circles.

Feature String for the word Amma would be (0,1,1) (1,1)(0,1)(1,1,0)(2,1)(0,1)(0,1,0)(2,1)(0,1)(0,1,0)(2,1)(0,0).

Figure 5 illustrates the primitive extraction from the word "amma" and its corresponding feature string gets

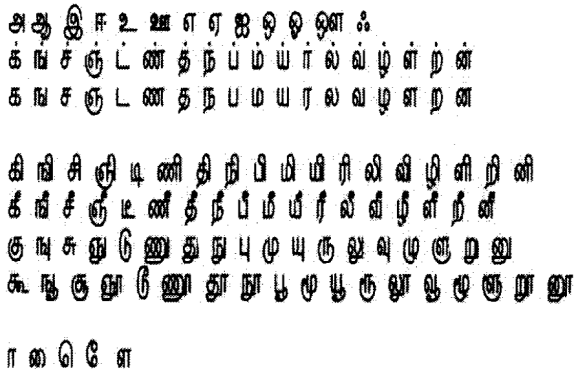

Fig. 2: Unique tamil characters

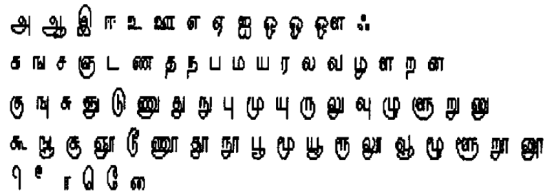

Fig. 3: Analyzed character set

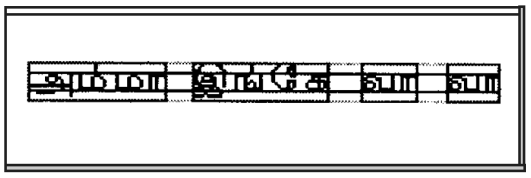

Fig. 4: Zone classification

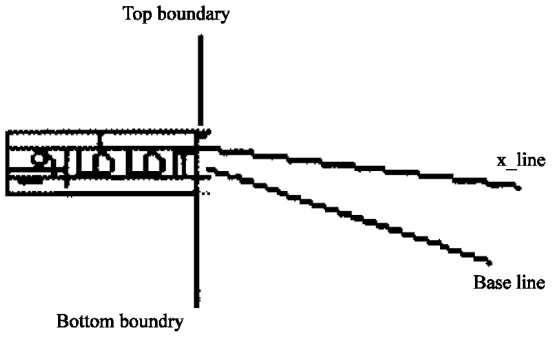

Fig. 5: Primitive extraction of the word amma

generated for the same example. Likewise feature strings can be generated for the keywords in the Tamil document image.

Likewise, the feature string gets generated for the entire keywords in Tamil Document Images, when it is presented as an input to the system.

User query matching: A Table with predefined strings for every character has been constructed initially. The

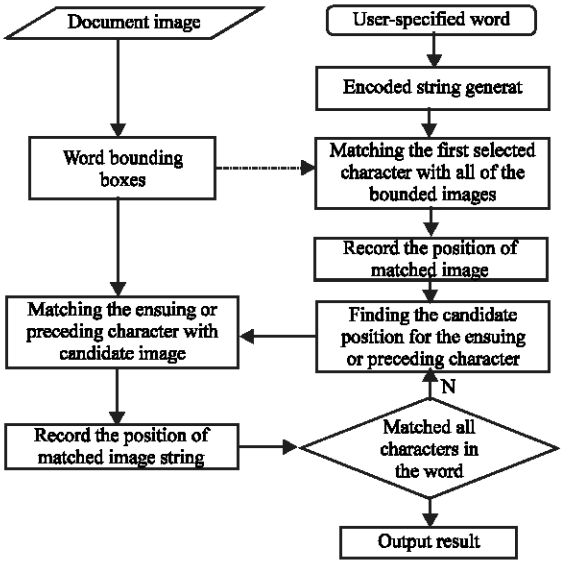

Fig. 6: Diagram for word studying

predefined feature strings are assigned to the characters in the user's input query. The codes for the character strings are appended in the text box and compared with the string tokens obtained from the image. The feature string tokens are assigned in such a way that it is produced from the images with the help of traversal features.

Retrieving keywords (word studying): Based on the processing described above, each word image is described by a primitive string. The word-studying problem can then be stated as finding a particular sequence in the primitive string of a word. The procedure of matching word images then becomes a measurement of the similarity between the strings that is the string representing the features of the query word and the string representing the features of a word image extracted from a document.

When the first code in the feature string of bounded image in the document gets matched with the encoded string of the user query word, the remaining codes are tested in succession to see if they can match the corresponding characters in the user specified word, subject to the constraint of size similarity and text line alignment. Once if they get matched, then all the keywords relevant to the user query would be produced as output result.

RESULTS AND DISCUSSIONS

When a document image is presented to the system, it goes through preprocessing, as in many document image-processing systems. Word objects are bounded based on a merger operation on the connected components. As a result, the left, top, right and bottom coordinates of each word bitmap are obtained. Meanwhile, the baseline and x-line locations in each word are also available for subsequent processing. Extracted word bitmaps with baseline and x-line information are the basic units for the downstream process of word matching and are represented with the use of primitive strings as described in this study. When a user keys in a query word, the system generates its corresponding feature string by aggregating the characters' primitive string tokens according to the character sequence of the word.

For example, the Feature String of the word "Amma" is: (0,1,1)(1,1)(0,1)(1,1,0)(2,1)(0,1)(0,1,0)(2,1)(0,1)(0,1,0)(2,1)(0,0).

The procedure of matching word images then becomes a measurement of the similarity between the strings that is the string representing the features of the query word and the string representing the features of a word image extracted from a document.

This system has been implemented using VB.NET . A set of 150 images whose dots per inch lesser than 300 have been tested in this system and the keyword study is relatively faster.

This yields a faster result in keyword based study with relative precision and recall measures. Moreover this kind of information retrieval can be performed even in low quality degraded documents where the dpi is less than 300.

CONCLUSION

Keywords are valuable indexing tools and if they can be identified at the image level, extensive computation during recognition will be avoided. Instead of converting entire document images into text equivalent, a Feature String algorithm has been devised in this study to generate feature strings for the word images in Tamil documents based on the features extracted from it. During retrieval, the same features could be extracted from the user specified word and can be matched with the word image. This would yield better results even in quality degraded documents.

REFERENCES

1. Doermann, D., 1998. The Indexing and Retrieval of Document Images: A Survey, Computer Vision and Image Understanding, pp: 287-298.
2. Mitra, M. and B.B. Chaudhuri, 2000. Information Retrieval from Documents: A Survey, Information Retrieval, pp: 141-163.

3. Chen, F.R. and D.S. Bloomberg, 1998. Summarization of Imaged Documents without OCR, Computer Vision and Image Understanding, pp: 307-319.
4. Bloomberg, D.S. and F.R. Chen, 1996. Document Image Summarization without OCR, Proc. Intl. Conf. Image Processing, pp: 229-232. A.L. Spitz, Duplicate Document Detection, Proc. SPIE, Document Recognition IV, 1997, pp: 88-94.
5. Smeaton, A.F. and A.L. Spitz, 1997. Using Character Shape Coding for Information Retrieval, Proc. Fourth Intl. Conf. Document Analysis and Recognition, pp: 974-978.
6. Spitz, A.L., 1999. Shape-Based Word Recognition, Intl. J. Document Analysis and Recognition, pp: 178-190.
7. Spitz, A.L., 2002. Progress in Document Reconstruction, Proc. 16th Intl. Conf. Pattern Recognition, pp: 464-467.
8. Yu, Z. and C.L. Tan, 2000. Image-Based Document Vectors for Text Retrieval, Proc. 15th Intl. Conf. Pattern Recognition, pp: 393-396.
9. Tan, C.L., W. Huang, Z. Yu and Y. Xu, 2002. Imaged Document Text Retrieval without OCR, IEEE Trans. Pattern Analysis and Machine Intelligence, pp: 838-844.
10. Syeda-Mahmood, T., 1997. Indexing of Handwritten Document Images, Proc. Workshop Document Image Analysis, pp: 66-73.
11. DeCurtins, J. and E. Chen, 1995. Keyword Spotting via Word Shape Recognition, Proc. SPIE, Document Recognition II, pp: 270-277.
12. Lu, Y., C.L. Tan, W. Huang and L. Fan, 2001. An Approach to Word Image Matching Based on Weighted Hausdorff Distance, Proc. Sixth Intl. Conf. Document Analysis and Recognition, pp: 921-925.
13. Lu, Y. and C.L. Tan, 2004. Information Retrieval in Document Image Databases, IEEE Transactions on knowledge and data engineering.
14. Seethalakshmi, R., 2005. OCR for Tamil Document Images using Unicode zheizang Journal.