INFORMATION RETRIEVAL FROM HISTORICAL DOCUMENT IMAGE BASE

Khurram Khurshid
CSE department, Institute of Space Technology,44000, Islamabad, Pakistan

Imran Siddiqi
College of Telecommunication Engineering, MCS-NUST, 46000, Rawalpindi, Pakistan

Claudie Faure
UMR CNRS 5141 - GET ENST, 46 rue Barrault, 75634 Paris, France

Nicole Vincent
Laboratoire CRIP5 – SIP, Université Paris Descartes, 45, rue des Saints-Pères, 75006, Paris, France

Keywords: Information retrieval, Word spotting, Dynamic Time Warping (DTW), Ancient document images, Edit distance.

Abstract: This communication presents an effective method for information retrieval from historical document image base. Proposed approach is based on word and character extraction in the text and attributing certain feature vectors to each of the character images. Words are matched by comparing their characters through a multi-stage Dynamic Time warping (DTW) stage on the extracted feature set. The approach exhibits extremely promising results reading more than 96% retrieval/recognition rate.

1 INTRODUCTION

The importance of digital libraries for information retrieval cannot be denied. Historical collections are of interest to a number of people, like historians, students and scholars, who need to study the historical originals. These documents contain invaluable knowledge but it is very time consuming for the researchers to search the required information in these books. Unfortunately, digitization alone is not enough to render historical document collections useful for such research purposes. Having the information available in an electronic image format makes it possible to share it with many people across large distances via the internet, DVDs or other digital media. The ability to search in ancient historical documents enhances the importance of digital libraries manifolds. However, the size of a collection is often substantial and the content is generally unstructured, which makes it hard to quickly find particular documents or passages of interest. Professional OCR software, designed for different languages especially Latin alphabets, give excellent recognition results on scanned images of contemporary documents in good quality. However, when used with ancient documents suffering from different degradations, discussed in detail in (Baird 2004), the recognition results drop significantly. The factors that affect the recognition rates mainly include the physical issues like document quality, strains and marks of liquids, dust and ink etc. on the pages; and also the semantic issues like labeling of entities on foreground etc.

Our work in this domain aims to facilitate information search in the invaluable content of ancient digitized books by spotting the instances of a given query word and presenting the user with all retrieved document images relevant to the given query. Since the content cannot be recognized, the query word is searched by creating the

Khurshid K., Siddiqi I., Faure C. and Vincent N..
INFORMATION RETRIEVAL FROM HISTORICAL DOCUMENT IMAGE BASE.
DOI: 10.5220/0003087401880193
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2010), pages 188-193
ISBN: 978-989-8425-28-7

corresponding image template and using pattern classification techniques to find all of its instances in the digitized image corpus; a technique commonly known as '*word spotting*' in the document analysis community. This ability to search in ancient historical documents will further enhance the importance and utility of such digital libraries providing users with instant access to the required information.

Word spotting, especially on the Latin alphabet, has been the focus of research in the last few years and a variety of techniques have been proposed to improve recognition rates. The field however, remains challenging and inviting, especially if the document base comprises low quality images, which in fact will be the subject of our research. Rath et al. (Rath 2007) introduced an approach for information retrieval and image indexing which involves grouping word images into clusters of similar words using word image matching. Four profile features for the word images are found which are then matched using different methods. Khurshid et al. (Khurshid 2008) have argued that features at character level give better results for word spotting as compared to word-level features. In (Rothfeder 2003), correspondences between the corner features have been used to rank word images by similarity. Harris corner detector is used to detect corners in the word images. Correspondences between these points are established by comparing local context window using the Sum of Square Differences (SSD) measure. Euclidean distance is then found between the word correspondences to give a similarity measure. Telugu scripts, a native Indian language, have been characterized by wavelet representations of the words in (Pujari 2002). Variations in image at different scales are studied using the wavelets. Wavelet representation exploits the inherent characteristics of the Telugu character. This wavelet representation does not give good results for the Latin letters. Adamek et al. introduced word contour matching in holistic word recognition for information extraction. The closed word contours are extracted and matched using an elastic contour matching technique (Adamek 2007).

In this paper, we propose an effective method for information retrieval from historical printed documents using word spotting. The proposed approach is based on matching the features of character images using multistage dynamic time warping (DTW). We first present an overview of the proposed scheme followed by a detailed discussion on indexing and retrieval phases. We then discuss the results obtained with the proposed methodology and a comparison with existing methods. Finally we present our concluding remarks and some interesting future research directions.

2 PROPOSED METHOD

The basis of our model for word spotting is the extraction of a feature set from character images. As opposed to (Rath 2007) where features are extracted at word-level, we first find all the characters in words and then extract features from these character images, thus giving more precision in word spotting than (Rath 2007) as later proved by the results. To begin, document image is first binarized and then a horizontal Run Length Smoothing Algorithm (RLSA) is applied to separate text in the image from graphics and extract the words. For each word, its characters are segmented using connected component analysis and applying a set of heuristics on the extracted components to find the actual characters. Once the characters are segmented, a set of six features is extracted for each character. Features of query word character images are compared with candidate word characters using multistage DTW.

Document processing is carried out offline, and an index file is created for each document image. The coordinates of each word, number and position of characters in the word, and also the features for each character image are stored in the index files. One advantage of having the index files beforehand is the capability it gives to crisply select the query word by just clicking on the word in the graphical interface of our document processing system. Similar processing is done on this query word to extract its character features. The features corresponding to the characters of query word are matched with the already stored features of character images in the corpus. The words for which the matching distance is less than a pre-defined threshold are considered to be the resulting spotted instances of the query word, representing the retrieval of required information.

3 INDEXING

The first step towards the extraction of features from a document image is separation of text and background – the binarization. Binarization of historical documents pose additional problems since the quality of these collections is generally quite

poor owing to storage over a long period of time, strains of liquids, dust, ink marks etc. Binarization methods that are based on computing a global threshold for the entire image are therefore not likely to work on these degraded images. We thus employ a local binarization method NICK (Khurshid 2009) that is adapted to low quality documents and where a binarization threshold is calculated for each pixel in the image as a function of its neighboring pixels. Figure 1 shows some binarization results using this method while the algorithmic details of threshold calculation can be found in (Khurshid 2009).

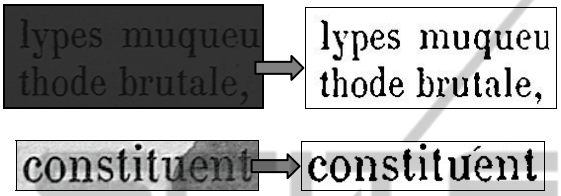

Figure 1: Binarization using NICK algorithm.

Once the document image is binarized, we seek to extract the words in document image. Since the inter-word distance (in pixels) is known to be greater than intra-word distance, applying a horizontal Run Length Smoothing Algorithm (RLSA) (Wong 1982) would merge all characters within a word into a single connected component. These connected components are then extracted to find the words in the image as illustrated in Figure 2. The threshold in RLSA is set as a function of the average distance between the raw connected components in the image which in fact provides an approximate idea of the intra-word distances.

Figure 2: a) Original image, b) binary image using NICK, c) horizontal RLSA, d) Extracted words (Khurshid, 2009).

Connected components extracted from RLSA applied image may also contain figures and images. Naturally they are larger in size in comparison to the components that correspond to words and hence are filtered out by comparing the area of each connected component with the average area of connected components in an image.

3.1 Extraction of Characters

A word comprises a set of characters and these Characters in an ideal case should be the connected components in a word. However, a connected component does not necessarily correspond to a character. A component may be composed of multiple characters or multiple components may form a single character. Therefore once we have the connected components of a word, we need to fix them so that they correspond to characters. This is achieved by applying a *three-pass* fixation method which is based on a set of heuristics. In the first pass, we fix the characters comprising two or more components on top of each other (figure 3a). Examples include characters like '*i*', '*j*' etc. and all such components are merged into one single character. In the second pass, we fix the characters which are broken into more than one component owing to poor quality of the document. These components overlap with each other at some point (figure 3b). Generally, these include characters like '*r*', '*g*' etc. The errors persisting after the first two passes are handled in the third pass where we remove the unnecessary punctuation marks (like '*,*' or '*.*') which are incorrectly found as a part (character) of the word (figure 4). All such marks are removed by eliminating the characters having an area significantly smaller than the average character area in the word.

$$A_i^j < \alpha . \mu^j$$

Where A_i^j represent the area of character i in word j, μ^j is the average character area of word j while the parameter α is an empirically chosen constant.

Figure 3: a) Original comps b) After Pass1 c) After Pass 2 d) Pass 3 demonstration, bounding box of the punctuation mark is removed (Khurshid, 2008).

The punctuation marks and all the small noisy characters are marked in this pass and hence they no longer constitute a part of the word.

3.2 Feature Extraction

Once the characters are segmented, we represent

them by a set of six feature vectors. Unlike (Rath 2007), where there are only four profile features and those too for characterizing the word image as a whole, our approach of having a set of six feature sequences for each character image provides a better representation as will be shown in the results section. The length of each of the six feature sequences is equal to the pixel width of the character image. These features are:

Vertical Projection Profile – Sum of intensity values in vertical direction; calculated in gray scale character image and normalized to get values between 0 and 1.
Upper Character Profile – For binarized character image, in each column we find the distance of the first ink pixel from the top of bounding box.
Lower Character Profile – The distance of the last ink pixel from the top of bounding box. Both the profiles are normalized to interval [0 1].
Vertical Histogram – Number of ink pixels in one column of the binarized character image.
Ink/non-ink Transitions – To capture the inner structure of a character, we find the number of non-ink to ink transitions in each binarized character image column.
Middle Row Transition– For the central row of the character image we find the transitional vector for ink/non-ink transitions. We place a 1 for every transition and 0 for all the non-transitions in the row.

For each word, we find these feature vectors for each of its characters. An index file is then created for each document where we store the location of the word components, the location of each of its characters and the features of each character.

4 INFORMATION RETRIEVAL

For information retrieval/key-word spotting, a two-step retrieval system is proposed. The first step finds out all eligible candidate words by applying a length-ratio filter. A bound has been set on the ratio of lengths of the two words for them to be considered eligible for matching. If the ratio of query word and some other word does not lie within a pre-specific interval, the word is excluded from the list of candidates.
The second step which is the main matching stage, query and candidate words are matched using a multi-stage DTW method. The first stage implements the classic DTW Edit distance algorithm (Wagner 1974) for matching the characters of the two words while elastic DTW coupled with Euclidean distance is used in the second stage for matching the features of the two characters.

Consider two words A and B to be compared, A having *m* characters while B comprising *n* characters. Both words are treated as series of characters, A = (a_1 ... a_m) and B = (b_1 ... b_n). Edit distance between the two character-series is found by defining a matrix W that calculates the cost of aligning the two subsequences. The entries of the matrix W are calculated as:

$$W(i,j) = \min\begin{Bmatrix} W(i-1,j-1)+\gamma(a_i \rightarrow b_j) \\ W(i-1,j)+\gamma(a_i \rightarrow \Lambda) \\ W(i,j-1)+\gamma(\Lambda \rightarrow b_j) \end{Bmatrix}$$

for all $i, j,\ 1 \le i \le m; 1 \le j \le n$

Λ is an empty character for which we have set all the values in its feature vectors to zero. The cost of replacing a_i with b_j is given by $\gamma(a_i \rightarrow b_j)$ while the costs of deleting a_i or inserting b_j are given by $\gamma(a_i \rightarrow \Lambda)$ and $\gamma(\Lambda \rightarrow b_j)$ respectively. These character matching costs are calculated in the second stage of DTW where we match the feature vectors of two characters. The advantage of using elastic DTW in this stage is its ability to cater for the nonlinear compression and stretch of characters. Hence two same characters will be matched correctly even if they differ in dimension.

To match two characters X and Y of widths *m* and *n* respectively, we treat the feature vectors of both as two series $X = (x_1 \ldots x_m)$ and $Y = (y_1 \ldots y_n)$. The distance between the two feature series is found by calculating an *mxn* matrix D representing the cost of aligning the two subsequences. The entries of D matrix are found using the following equation:

$$D(i,j) = \min\begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_j)$$

Where $d(x_i,y_j)$ is a distance metric which in our case is the Euclidean distance in the feature space. Figure 4 visually depicts the proposed multi-stage DTW.

Once the entries of D are calculated, the warping path is determined by backtracking from *(m, n) towards (0,0)* along minimum cost path. The final distance between the two characters is the cost *D(m, n)* divided by the number of steps of the warping path. If this final distance is less than a pre-defined character-threshold, the two characters are ranked similar. The same is done for the matrix W to get the cost of matching the two words. If this cost *W(m,n)*

is less than a word-threshold, A and B are considered to be the instances of the same word.

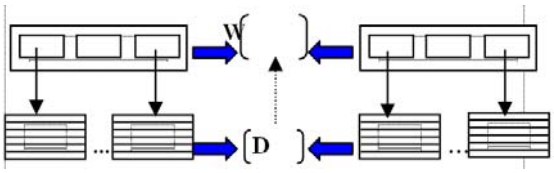

Figure 4: Multistage DTW: Two words are compared using Edit matrix W whose each entry represents the cost of matching two characters/substrings; while D gives the cost of matching two characters & is calculated by matching the feature sequences of the two characters.

5 EXPERIMENTAL RESULTS

Owing to the absence of a benchmark of ancient printed documents, we not only applied our method to a set of documents but also evaluated other approaches on the same data set for a rational comparison. In our previous work (Khurshid 2008), we compared a character-matching based word spotting method (employing simple DTW) with other approaches such as correlation-based word matching, the word matching based on features presented in (Rath 2007) and word matching using our six features computed from words (instead of characters). The results showed that our character-matching system achieved the best results (Figure 5) (Khurshid 2008).

In the present study we compare the results of (Khurshid 2008) with our multistage DTW to analyze if introducing multi-stage DTW would produce any improvements in results. Evaluations were carried out on the document images provided by the Bibliothèque Interuniversitaire de Médecine, Paris (BIUM).

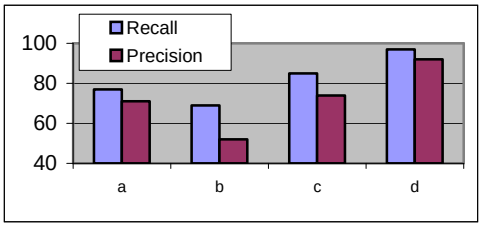

Figure 5: Comparison of information retrieval (via precision and recall rates) using different methods a) Correlation b) Matching using four word features (Rath 2007) c) Using six word features d) Matching using character features (Khurshid 2008).

For the experimental dataset, we chose, from 10 different books, 40 document images having a total of 17,010 words in all. For evaluation, 50 different query lexiques of varied lengths and styles having 400 instances in total were selected. The results of word spotting are summarized in Table 1.

Table 1: Results summary.

	(Khurshid 2008)	Multi-stage DTW
#query word instances	400	400
#words detected perfectly	381	386
#False positives (FP)	51	16
Precision %	88.19%	96.20%
Recall %	95.25%	96.50%

It can be seen from the above table that the new multi-stage DTW based method has higher precision with better recall rate, thus exhibiting its robustness and usability in information retrieval for commercial applications. Considering optimum threshold values for different word lengths, a precision of 96.2% is achieved by our system with a recall rate of 96.5%.

6 CONCLUSIONS

We have proposed a new system of information retrieval in ancient document images using word spotting. The approach is based on the matching of character features by employing a multi-stage Dynamic Time Warping. Results obtained by the proposed methodology are very encouraging. The number of false positives is very less as compared to the previous results, which shows the prospects of taking this method even further by improving different stages and adding more features to achieve even higher percentages. Currently, the word spotting is done only on the horizontal text but we also intend to focus on spotting words in vertical text lines in our future research. A web based user-friendly interface can be built using the proposed algorithm at the back end which will give a global accessibility to the system. The system can be enhanced in future for information retrieval in Oriental scripts as well.

REFERENCES

Adamek, T., O'Connor, N. E., Smeaton, A. F., 2007. Word

matching using single closed contours for indexing handwritten historical documents, *IJDAR*, 9, 153 – 165

Baird H. S., 2004. Difficult and urgent open problems in document image analysis for libraries, *1st International workshop on Document Image Analysis for Libraries*

Digital Library of BIUM (Bibliothèque Interuniversitaire de Médecine, Paris), http://www.bium.univ-paris5.fr/histmed/medica.htm

Keogh, E. and Pazzani, M., 2001. Derivative Dynamic Time Warping, *First SIAM International Conference on Data Mining,* Chicago, IL.

Rath, T. M, Manmatha, R., 2007. Word Spotting for historical documents, *IJDAR, 9, 139-152*

Khurshid, K., Faure, C., Vincent, N., 2008. Feature based word spotting in ancient printed documents, *8th International workshop on pattern recognition in information systems*, Spain

Khurshid, K., Faure, C., Vincent, N., 2009. A novel approach for word spotting using Merge-Split Edit distance, *CAIP'09*, Germany

Khurshid, K., Siddiqi, I., Faure, C., Vincent, N., 2009. Comparison of Niblack inspired binarization techniques for ancient document images, 16th internation conference of document recognition and retrieval (*DRR), USA.*

Leedham, G., Yan, C., Takru, K., Hadi, J., Tan, N., Mian, L., 2003. Comparison of Some Thresholding Algorithms for Text/Background Segmentation in Difficult Document Images, *7th International Conference on Document Analysis and Recognition, USA*

Pujari, A. K., Naidu, C. D., Jinaga, B. C. 2002. An adaptive character recogniser for telugu scripts using multiresolution analysis and associative memory, *ICVGIP'02*

Rothfeder, J. L., Feng, S., Rath, T. M., 2003. Using corner feature correspondences to rank word images by similarity, *Conference on Computer Vision and Pattern Recognition Workshop*, Madison, USA

Wagner, R. A., Fischer, M. J., 1974. The string-to-string correction Problem, *Journal of ACM*, v21, pp 168-173

Wang, K. Y., Casey, R. G., Wahl, F. M., 1982. Document analysis system, *IBM J. Res.Development*, Vol. 26, pp. 647-656