Feature-based Word Spotting in Ancient Printed Documents

Khurram Khurshid[1], Claudie Faure[2] and Nicole Vincent[1]

[1]Laboratoire CRIP5 – SIP, Université Paris Descartes, 45 rue des Saints-Pères 75270, Paris Cedex 06, France

[2]UMR CNRS 5141 - GET ENST, 46 rue Barrault, 75634 Paris Cedex 13, France

Abstract. Word spotting/matching in ancient printed documents is an extremely challenging task. The classical methods, like correlation, seem to fail when tested on ancient documents. So for that, we have formulated a multi-step document analysis mechanism which mainly revolves around finding the words and their characters in the text and attributing each character by some multi-dimensional features. Words are matched by comparing these multi-dimensional features of the characters using Dynamic Time warping (DTW). We have tested this approach on ancient document images provided by the Bibliothèque Interuniversitaire de Médecine, Paris. Our Initial experiments exhibit encouraging results having more than 90% precision and recall rates.

1 Introduction

Spotting words in documents written in the Latin alphabet has received considerable attention lately. Although a lot of work has already been done in the field of word spotting, it still remains an inviting and challenging field of research mainly because the results achieved so far are not satisfactory for huge volumes of data; specially if the document base consists of a set of ancient printed documents of relatively degraded quality.

Lot of work has been done in the domain of word spotting and optical character recognition in ancient document images. There are plenty of issues and problems related to ancient printed documents which are discussed in detail in [10] and [12]. These problems provide a big challenge for the researchers working in this domain to achieve better results. In this paper though, we will not be addressing these issues. Rath and Manmatha [1,11] introduced an approach which involves grouping word images into clusters of similar words by using word image matching. Four profile features for the word images are found which are then matched using different methods [1]. [7] has used the corner feature correspondences to rank word images by similarity in historical handwritten manuscripts. Telugu scripts have been characterized by wavelet representations of the words in [5]. But this wavelet representation does not give good results for the Latin letters [5]. Adamek et al. introduced the

Khurshid K., Faure C. and Vincent N. (2008).
Feature-based Word Spotting in Ancient Printed Documents.
In *Proceedings of the 8th International Workshop on Pattern Recognition in Information Systems*, pages 193-198

matching of word contours for holistic word recognition. The closed word contours are extracted & matched using elastic contour matching technique [9].

2 Proposed Method

Our model is based on the extraction of different multi-dimensional features for the character images for the purpose of word matching. As opposed to [1] where features are extracted from the whole word image, we find the characters of the word and the features are extracted from these character images, thus giving more precision in word spotting as later proved by the results.
First of all the document image is binarized using Niblack [3] algorithm. We have modified the original Niblack to gain better results. The text in the document image is separated from graphics and the words in the document are extracted by applying RLSA [2] and finding the connected components in the RLSA image. For each word component, characters are found using its connected component analysis. The connected components are then fixed using a 2-step process to get the characters for each of which, we extract a set of features. The query word is searched in the document by matching the character features using Dynamic Time Warping [6]. Test query is processed in the same way and the features of the query word's characters are matched with the already stored features of the characters in words using DTW. The words for which the matching distance is less than a threshold are the resulting spotted words. We now see in detail the different processing stages:

2.1 Binarization

To get better results for word spotting, binarization has to be very good. For that, we modified the Niblack algorithm [3] to make it more efficient for the ancient documents and gain better results. The Binarization threshold is found out using the following formula:

$$T = m + k\sqrt{\sum (p_i^2 - m^2) / NPT} \quad (1)$$

Where:

k = -0.2, m = mean grey value, p_i = pixel value of grey scale image , NPT = number of pixels

The binarization is tested on different BIUM images [8], having smaller resolution (550 x 913) as well as higher resolution (1536 x 2549). The results obtained are extremely satisfactory for different types of documents.

2.2 Words Extraction using RLSA

We consider the words as connected components in the document image where a horizontal Run Length Smoothing Algorithm (RLSA) has been applied. So after

binarization we apply a horizontal RLSA [2] with threshold = 5, and find the connected components in the resulting image to extract the word components (fig 1).

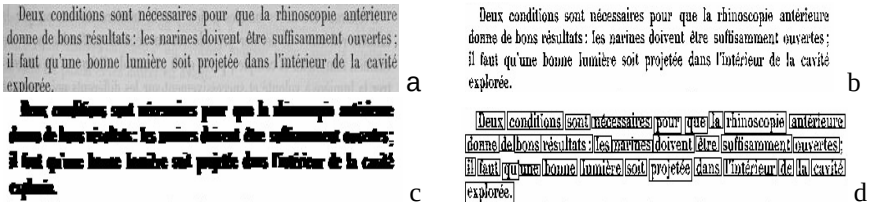

Fig. 1. a) original image b) binary image using modified Niblack c) horizontal RLSA d) Words extracted.

The word components found in the document image also contain the figures/images as large components. So to remove figures, we only have to remove those components which do not satisfy the following criteria:

Component Area < (Mean comp. area x 5) AND Component height < (Mean comp. height x 4)

2.3 Extraction of Characters

As we know that a characters in ideal case should be the connected components in the word, so we carry out a connected component analysis on the extracted word images to get the characters. But a character may be broken into multiple components or multiple characters may form a single component. So once we have the components, we need to fix them so that they correspond to characters. For that, we apply a 2-pass fixing method (figure 2). In the first pass, we fix the characters comprising 2 or more components on top of each other. In the 2nd pass, we fix the characters which are broken into more components. These components overlap with each other at some point. These include characters like r, g etc.

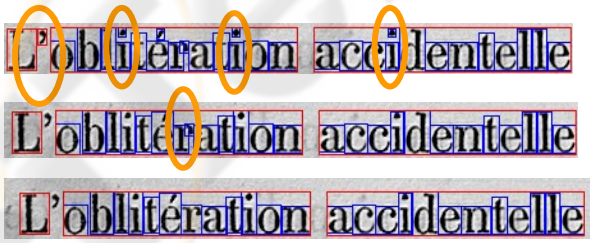

Fig. 2. a) Original components b) After pass-1 c) After pass-2.

2.4 Feature Extraction

We have employed a set of six features for the character images:

Vertical Projection Profile – sum of intensity values in vertical direction; calculated in gray scale character image and normalized to get valued between 0 and 1.

Upper Character Profile – for a binarized character image, for each column we find the distance of the first ink pixel from the top of bounding box.

Lower Character Profile – just like in upper word profile, here we find the distance of the last ink pixel from the top of bounding box. Both upper and lower profiles are normalized between 0 and 1.

Vertical Histogram – number of ink pixels in one column of binarized character image.

Ink/non-ink Transitions – to capture part of 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. Initially we found row transitions for 3 central rows and applied logical OR on them to get mean central row transitional vector. It means that if there is an ink pixel in any of the three central rows, we take it as an ink pixel. We tested using both ways but found that using only the central row gives better results. So we stuck to that.
For each word, we find the features for each of its characters. After processing a document image, we create its index file in which we store the location of the word components, the location of each of its characters and the features of each character.

2.5 Word Matching by DTW

For two words to be considered eligible for matching, we have set bounds on the ratio of their lengths (number of characters). If the ratio does not lie within a specific interval, we do not try matching the two words. Now for matching the words, we match the features of the words' characters using DTW. The advantage of using DTW is that it is able to account for the nonlinear stretch and compression of the words as it finds a common time axis[6]. So two same words which differ in dimension will be matched correctly unlike correlation [4] where the words need to be of the same dimension to be matched.

So for matching 2 characters, we treat the feature vectors of both as two series X = (x_1 ... x_m) and Y = (y_1 .. y_n). To determine the DTW distance/cost between these two time series, we find a matrix D of m x n order which shows the cost/ distance of aligning the 2 subsequences. The entries of the matrix D are found as:

$$D(i,j) = \min\begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_j) \qquad (2)$$

Here for $d(x_i , y_j)$, we have used the Euclidean distance in the feature space:

Once all the values of D are calculated, the warping path is determined by backtracking along the minimum cost path starting from (m , n). The final matching cost is the cost D(m , n) divided by the number of steps of the warping path. Two words are

ranked similar if this final matching cost is less than a character-threshold (which has been found after a set of experimentation). More details of DTW can be found in [6] and [1]. One character of query word is matched with different number of neighboring characters in the test word depending upon the size of the query word. For each query character, we find its best match (if exists) from the inspected test characters and add the character matching cost to the total word cost. After matching the characters of the 2 words, we normalize the total word cost (which is just the sum of the individual character costs divided by the number of characters matched). If for two words, this normalized word-cost is less than our word-threshold, then we say that the 2 words are same.

3 Experimental Results

The absence of a benchmark image database of ancient printed document database for word spotting motivated us not only to apply our method to some set of documents but also to test other approaches on the same set of documents for a rational comparison. The other approaches include correlation as word matching measurement, the word feature representation presented in [1] and word matching using our 6 feature set for word images. We compared the results of all these with our character matching aproach. The experiments were carried out on the document images provided by the Bibliothèque Interuniversitaire de Médecine, Paris [8]. We chose 35 words of different lengths from different document images for query. The evaluation is done by computing recall and precision percentages. Results for word spotting using correlation approach are not very satisfactory with precision just over 70%. By using feature matching of the whole word images as proposed in [1], the precision dropped to 53%. Testing by using our 6 features for the whole word images, it can be noticed from table 1 that our feature set achieves higher recall and precision rates than that of [1]. And by our method of matching words using character features, the results achieved are much better than either correlation or [1] as shown in table1.

Table 1. Comparison of different approaches.

Method	Precision	Recall
Correlation	71%	77%
Matching using 4 features of words[1]	53%	69%
Matching using our 6 word features	74%	85%
Matching using character features	92%	97%

4 Conclusions

We have proposed a new method for word spotting which is based on the matching of features of the characters images by Dynamic Time Warping. The results obtained by using this method are very encouraging. The number of false positives is very less as

compared to the results obtained by cross-correlation as well as [1] 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 our upcoming work is focusing on the word spotting in vertical text lines. This work can also be adapted to handwritten manuscripts.

References

1. Tony M. Rath, R. Manmatha,: Word Spotting for historical documents, IJDAR (2007) 9:139-152
2. K. Y. Wang, R. G. Casey and F. M. Wahl,: Document analysis system, IBM J. Res.Development, Vol. 26, pp. 647-656, (1982).
3. Graham Leedham, Chen Yan, Kalyan Takru, Joie Hadi Nata Tan and Li Mian,: Comparison of Some Thresholding Algorithms for Text/Background Segmentation in Difficult Document Images, 7th International Conference on Document Analysis and Recognition ICDAR, (2003).
4. P. J. Burt, C. Yen, X. Xu,: Local Correlation Measures for Motion Analysis: a Comparative Study, IEEE Conf. Pattern Recognition Image Processing (1982), pp. 269-274.
5. A. K. Pujari, C.D. Naidu, B.C. Jinaga,: An adaptive character recogniser for telugu scripts using multiresolution analysis and associative memory, ICVGIP (2002).
6. Keogh, E. and Pazzani, M.,: Derivative Dynamic Time Warping, First SIAM International Conference on Data Mining, Chicago, (2001).
7. Jamie L. Rothfeder, Shaolei Feng and Toni M. Rath,: Using corner feature correspondences to rank word images by similarity, Conference on Computer Vision and Pattern Recognition Workshop, Madison, USA, (2003), pp. 30-35.
8. Digital Library of BIUM (Bibliothèque Interuniversitaire de Médecine, Paris), http://www.bium.univ-paris5.fr/histmed/medica.htm
9. Adamek, T., O'Connor, N. E. and Smeaton, A. F.,: Word matching using single closed contours for indexing handwritten historical documents, IJDAR (2007), 9, 153 - 16
10. A.Antonacopoulos, Karatzas D., Krawczyk H. and Wiszniewski B.,: The Lifecycle of a Digital Historical Document: Structure and Content, ACM Symposium on Document Engineering, (2004), 147 –154.
11. Tony M. Rath, R. Manmatha,: Features for Word Spotting in Historical Manuscripts, Seventh International Conference on Document Analysis and Recognition ICDAR, (2003).
12. Baird H. S.,: Difficult and urgent open problems in document image analysis for libraries, 1st International workshop on Document Image Analysis for Libraries, (2004).