Shape-based Word Spotting in Handwritten Document Images

Angelos P. Giotis*†, Giorgos Sfikas†, Christophoros Nikou* and Basilis Gatos†
*Department of Computer Science and Engineering, University of Ioannina, Greece
{agiotis, cnikou}@cs.uoi.gr
†Computational Intelligence Laboratory, Institute of Informatics and Telecommunications,
National Center for Scientific Research "Demokritos", GR-15310 Athens, Greece
{bgat, sfikas}@iit.demokritos.gr

***Abstract*—In this paper, we address the problem of word spotting using a shape-based matching scheme between segmented word images represented by local contour features. As in a typical query-by-example (QBE) paradigm, a user selects an instance of the query word from the collection of interest and a ranked list of images is returned, based on their similarity with the query. This is accomplished in two steps. The query image is firstly aligned with the test image according to a similarity measure defined on their descriptors and then the aligned images are matched through a deformable non-rigid point matching algorithm. Experiments are carried out on historical handwritten text, written in Greek and English, respectively. Moreover, comparisons with other QBE methods show the efficiency of our system as well as its flexibility in adapting to different scripts.**

***Keywords*-word spotting; non-rigid point matching; local contour features; handwritten text;**

I. Introduction

Digitized information contained in large databases of documents renders their indexing essential for information retrieval purposes. In this context, word spotting is an alternative solution to optical character recognition (OCR) approaches, which are rather inefficient for recognizing text of degraded quality.

Perhaps the most common distinction of word spotting approaches depends on how the input is specified. In *query-by-example* (QBE) methods, an actual instance of the query word is provided to trigger the search for similar instances in terms of appearance. In most QBE approaches, no prior semantic knowledge or transcription is available and thus, another way to categorize word spotting techniques lies on whether they are *learning-based* or not. Following the *learning-free* scenario, local features containing appearance and texture information are combined in a fixed-length vector in [1]. This representation renders the spotting problem as a nearest neighbor search, thereby allowing for fast comparison between two words. Variable-length representations are also widely used in the QBE approach. Leydier et al. [2] propose an elastic matching method to compare different pixel-wise gradient matchings. In addition, the most common local approach to compute the distance between sequences of features is dynamic time warping (DTW), which is thoroughly employed in [3] and compared with similar matching techniques of word profiles. A QBE, though learning-based, method is presented in [4], for spotting out-of-vocabulary (OOV) words, using a semi-continuous hidden Markov model (HMM). The model's parameters are estimated on a pool of unsupervised samples which allow the model to adapt online to the query image.

In *query-by-string* (QBS) approaches on the other hand [5]–[7], a query word representation is accrued from character or sub-word level training samples. Almazan et al. [5] use a fixed-length representation computed over SIFT descriptors, in an attribute-based framework. These attributes encode information which is shared between similar words. Either an example from the document collection or an ASCII text query can be used as input. A recognition-oriented system based on recurrent neural networks is used in [6] to spot arbitrary textual queries using models learnt from character class probabilities. Fisher et al. [7] incorporate character language models into their HMM-based system to improve the spotting performance. These methods can deal with the inherent handwriting variability on the ground that an adequate subset from the collection of interest is transcribed beforehand. However, their adaptation capability to different languages is uncertain.

Some methods require the document images to be segmented at word [3]–[5] or line [6], [7] level, while others [1], [2] are applied directly to the document page. Therefore, we can distinguish two more categories of word spotting approaches, corresponding to the *segmentation-based* and the *segmentation-free* track. With respect to the QBE paradigm, in the segmentation-based track, the query image itself is usually discarded from the evaluation task, whereas in the segmentation-free track, the query image is also considered to be a true positive, since no ground-truth bounding box is available and thus, it could be missing from the retrieved areas.

Relying on an object detection system for real images [8], we propose a technique for matching contour shapes, which is built upon our previous work [9] for spotting handwritten words in multi-writer conditions. The proposed system differs significantly from our previous one, as it does not involve training from multiple instances of the query. It

978-1-4799-1805-8/15/$31.00 ©2015 IEEE

is rather applied on a single word image selected as a query, without the need for building an average shape to represent a word-class. Assuming that document images have already been binarized and segmented at word level, the first step of the proposed approach is to extract the contour from segmented word images using a thinning morphological operation. Subsequently, scale invariant contour features, initially proposed by Ferrari et al. [8], are extracted from thinned word images and stored offline.

Our main contribution lies on the direct use of these features for retrieving the location and scale of the center of the query's bounding box inside the test image. This acts as an initialization of the non-rigid point matching algorithm, which deforms the query word in order to capture the shape of the word of interest. The outcome of this matching process is a detection at point level (boundary) which is scored by a weighted sum of four terms [8]. As a second contribution, we extend this weighted sum with an extra term to account for false detections obtained from partial matches of the query inside the test image. Finally, we evaluate the Mean Average Precision (MAP) of the proposed system in heterogeneous handwritten scripts and compare, among others, to the DTW method [10], thereby implying the potential of the system to extend to different languages.

The rest of our work is structured as follows. In Section II, we present the local contour features used to represent a word image. Section III describes the word image matching algorithm. Experimental results on the George Washington dataset (GW20) [11], as well as on a dataset containing handwritten historical Greek (GRPOLY-DB) [12] are discussed in Section IV and finally, conclusions are drawn in Section V.

II. Word Representation

In our recent work [9], it was shown that to achieve a matching of high accuracy in documents which present variability in writing style, it is essential to detect a query word at boundary level. Such a detection requires a contour-shape, formed by continuous connected curves, to describe each word image. This representation allows for determining the candidate location and scale of the query inside the test image which is then used as input to the subsequent non-rigid point matching scheme (Section III).

A. Preprocessing

To create this contour shape, we first extract the skeleton of a word by applying a thinning morphological operation to the binarized word images. This procedure erodes away the boundaries of foreground shapes as much as possible, but does not affect pixels at the ends of lines. Edge pixels (edgels) comprising the skeleton are initially chained into edgel-chains, which are then linked at their discontinuities and approximately straight segments are fit to them, using the technique described in [13]. Segments are fit over individual edgel-chains and bridged across their links.

(a) (b)

Figure 1. (a) The word "$M\acute{\eta}\tau\eta\rho$" ("Mother" in English) from the GRPOLY-DB dataset written in historical Greek. (b) The word "Orders" from the GW20 dataset. Extracted PAS features from each thinned image are shown on the right (the figure is better seen in color).

B. Word description

The next step is to detect the pairs of adjacent segments (PAS) conceived by Ferrari et al. [8] and use them to represent each word. A PAS feature, $\mathbf{P} = (x, y, s, d)$ has a location (x, y) which consists of the mean over the two segment centers, a scale s which is the distance between the segment centers and a descriptor $\mathbf{d} = (\theta_1, \theta_2, l_1, l_2, r)$, invariant to translation and scale changes. Example binary instances of the words "$M\acute{\eta}\tau\eta\rho$" ("Mother" in English) written in historical Greek and the word "Orders" from the GW20 benchmark [11], along with their respective skeletons and a subset of PAS features are illustrated in Fig. 1. Each color on the right of the figure corresponds to a PAS whereas the numbers correspond to its segment IDs.

C. Descriptor similarities

Connecting segments over edge discontinuities renders PAS features robust to interruptions along the word contour and to short missing parts. These may be due to segmentation errors, faded ink or poorly pressed thin strokes. It is interesting to notice that PAS may overlap, meaning that they can share segments and thus cover pure portions of a word's boundary. Consequently, they can be easily detected across instances of the same word-class, in terms of finding a common structure among similar instances.

To this end, we make use of the similarity measure between two word images proposed in our previous work [9]. This PAS dissimilarity $D(\mathbf{P}, \mathbf{K})$ between the descriptors $\mathbf{d}^p$, $\mathbf{d}^k$ of two PAS $\mathbf{P}, \mathbf{K}$, is defined by:

$$D(\mathbf{d}^p, \mathbf{d}^k) = w_r \|\mathbf{r}^p - \mathbf{r}^k\| + w_\theta \sum_{i=1}^{2} D_\theta(\theta_i^p, \theta_i^k) + \sum_{i=1}^{2} |\log(\frac{l_i^p}{l_i^k})| \quad (1)$$

The first term is the difference in the relative locations of the two PAS, the second term contains the difference between their segment orientations and the last term accounts for the difference in their segment lengths. The relative location

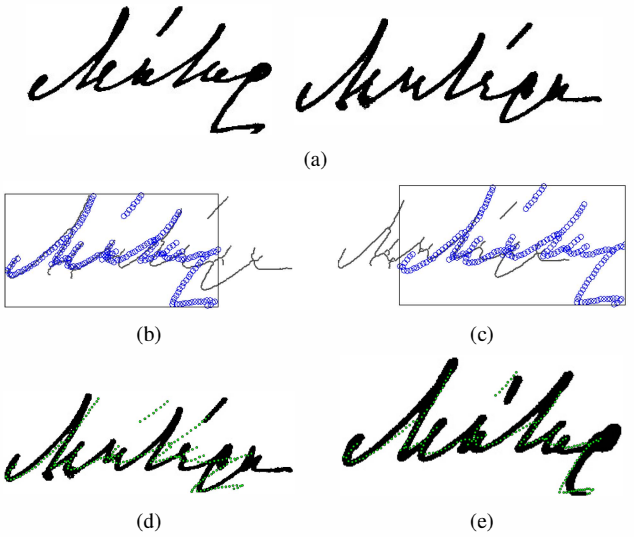


Figure 2. Query detection. (a) Query image on the left, test image on the right. (b)-(c) Initializations of TPS-RPM by centering the query to the word's center. (d) The output shape (false positive) is superimposed in green on the test image. (e) Superimposed output shape in green upon an actual instance (the figure is better seen in color).

of each PAS feature, as well as its segment lengths are normalized by dividing with its scale s. As segment lengths are often inaccurate, higher weight is given to the two other terms of the dissimilarity measure.

III. Word Image Matching

The first step to detect occurrences of the query inside the test images is to determine their possible location and scale using the predefined dissimilarity measure (1). More specifically, each PAS $\mathbf{P}$ inside the query is matched with every PAS $\mathbf{K}$ from the test image according to $D(\mathbf{P}, \mathbf{K})$. If the dissimilarity is lower than a specific threshold γ then this match votes for a candidate location and scale of the query's center inside the test image. Each vote is weighted by $(1 - D(\mathbf{P}, \mathbf{K})/\gamma)$.

For instance, Fig. 2(a) depicts the query "$Μήτηρ$" and the test word "$Μητέρα$", which is rather relevant, though not an actual occurrence. Local maxima inside the $3D$ voting spaces (location, scale) yield approximate positions and scales of the query's center inside the test image. These act as different initializations (Fig. 2(b), 2(c)) to the subsequent non-rigid point matcher which deforms the query to capture the shape of the unknown word, as it is shown in Fig. 2(d) for the initialization of Fig. 2(b).

Regarding the first stage of the matching process, the success of this alignment of the query inside the test image is attributed to adopting PAS as basic shape elements. Unlike other local features, such as individual edgels, the shape of the PAS and its size, are more distinctive than the orientation of an edgel. Hence, it is very unlikely for a set of PAS not belonging to a common shape structure of the query-class, to accidentally have similar locations, sizes and shapes across instances. In other words, a subset of the query's PAS is common among its instances.

As for the second step, we apply the thin plate spline robust point matching (TPS-RPM) algorithm [14], which matches two point sets $\mathbf{V} = \{\boldsymbol{v}_i\}_{i=1,\ldots,N}$ and $\mathbf{X} = \{\mathbf{x}_i\}_{i=1,\ldots,M}$, by applying a non-rigid TPS mapping parameterized by $\{\mathbf{c}, \mathbf{w}\}$ to $\mathbf{V}$. TPSs are chosen because they can be decomposed into affine and non-affine subspaces as it is shown by the following vector valued function:

$$\mathbf{f}(\boldsymbol{v_i}) = \boldsymbol{v}_i \cdot \mathbf{c} + \boldsymbol{\phi}(\boldsymbol{v}_i) \cdot \mathbf{w} \tag{2}$$

where $\mathbf{c}$ is the affine component and $\mathbf{w}$ is a non-affine warping coefficient, which is combined with the TPS vector valued kernel $\boldsymbol{\phi}(\boldsymbol{v}_i)$ to form the non-rigid warp. TPSs minimize an energy function by iteratively alternating between updating a correspondence matrix, while keeping the transformation $\{\mathbf{c}, \mathbf{w}\}$ fixed and vice versa. Moreover, it rejects points for which no correspondence exists.

In line with [8], a detection at point level is scored by a weighted sum of four terms which is explained as follows:

1) The amount of matched query points to the points of the test image with a high confidence measure. These are all points $\boldsymbol{v}_i$ with $\max_{j=1,\ldots,N}(\mathbf{m}_{ij}) > 1/N$, where $\mathbf{m}$ is the correspondence matrix.
2) The sum of square distances between the matched query points and the corresponding image points, which is made scale-invariant by normalizing them by the squared range r^2 of the image point coordinates (width or height, whichever is larger).
3) The deviation $\sum_{i,j\in\{1,2\}} (\mathbf{I}(i,j) - \mathbf{c}(i,j)/\sqrt{|\mathbf{c}|})^2$ of the affine component $\mathbf{c}$ of the TPS from the identity $\mathbf{I}$. The normalization by the determinant of $\mathbf{c}$ factors out deviations due to scale changes.
4) The amount of the non-rigid warp $\mathbf{w}$ of the TPS $\text{trace}(\mathbf{w}^T \boldsymbol{\Phi} \mathbf{w})/r^2$, where $\boldsymbol{\Phi}$ is a $N \times N$ matrix formed by the kernels $\boldsymbol{\phi}(\boldsymbol{v}_i)$.

This scoring integrates the information provided by a matched shape. Its value is high when TPS fits many points well (terms 1 and 2), without having to distort much (terms 3 and 4). It is also interesting to note that different initializations from the previous stage result into separate detections from which we retain the one with the highest score. The second step of the proposed matching scheme is crucial for obtaining a more accurate detection. While the query alignment stage handles invariance in terms of translation and scale, the non-rigid registration algorithm deals with the case of skewed words or slanted characters, which are rather frequent in handwritten documents.

Finally, we add a term to tackle false detections of partial matches, such as that of Fig. 2(d). Assuming that B_{test} expresses the image boundary points and that B_{query} consists of the matched output points to the test image, we propose an accuracy term as the average value between two measures:

1) *Coverage* is the percentage of points from B_{test} closer than a threshold t from any point of B_{query}.
2) *Precision* is the percentage of points from B_{query} closer than t from any point of B_{test}.

The measures are complementary and t is set to be 4% of the diagonal of the bounding-box of B_{test}. In our implementation, the relative weights between these five terms have been selected manually and kept fixed in all experiments. The impact of this extra term on the scoring function is that it renders scores between correct and false detections even more discriminative. In fact, the output shape of Fig. 2(d) achieves a matching score with value 25.61% whereas the true positive score of the output shape in Fig. 2(e) is 82.66%.

IV. Experimental Evaluation

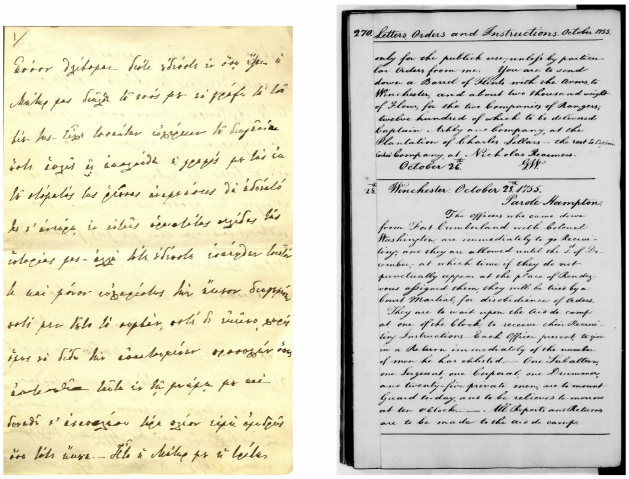
In this section, we present the datasets used to evaluate the proposed word spotting approach as well as the criteria applied for selecting appropriate queries. Then we briefly refer to the state-of-the-art QBE systems upon which comparisons are made for each dataset.

A. Datasets and Protocol

Experiments are carried out on two challenging datasets. The first dataset is written in historical Greek by Sophia Trikoupi, during the 19^{th} century. There are 46 pages of handwritten polytonic text containing 4939 words, which derive from the archives of the Hellenic Parliament library. A sample page from the GRPOLY-DB [1] dataset [12] is illustrated in Fig. 3(a). Text is rather cursive accompanied by intra-writer variability among instances of the same word. In order to evaluate our method we selected words whose occurrences appear more than five times and their length is greater than 6 characters. The query list provided by this criterion includes 21 distinct words along with their instances, yielding a total number of 141 queries. All pages are binarized using the technique described in [15] and manually segmented at word level. Each word is manually annotated and we only deem an exact match of the query inside the test image as a hit.

The second dataset is the English manuscript GW20 from the George Washington collection [11], containing 20 pages of historical handwritten cursive text which include 4860 words. A sample page from this collection is shown in Fig. 3(b). Similarly to Leydier et al. [2], we selected the same 15 words to evaluate our method. These are the most significant words in terms of occurrence frequency and semantics. We consider all instances of each of the 15 words, comprising a total number of 306 queries. In line with the GRPOLY-DB benchmark, close hits such as the words "Fort" and "fort" are deemed as false positives in the evaluation task.

Finally, one important but not restrictive aspect of our approach is the parameter estimation of our system. All



(a) (b)

Figure 3. Sample pages from (a) the GRPOLY-DB dataset and (b) the GW20 benchmark [11], respectively.

parameters concerning the proposed system are estimated once using a small subset of handwritten word images from the IAM dataset and kept fixed in all experiments. Neither query nor dataset specific tuning is applied. As a means to improve the speed of the proposed matching scheme we introduce a pruning criterion which discards unlikely similar matches. This is based on the difference in the size of the descriptors between two words as well as the difference in their respective number of PAS. Such a pruning decision step, before comparing two words, seems to not only avoid at least half of the total matches to be processed per query, but also improve the average precision of our system, with low risk of reducing its recall.

Considering the above, we evaluated the performance of the proposed approach using the Mean Average Precision (MAP). This metric is calculated using the trec_eval software as it is implemented by the National Institute of Standards and Technology (NIST) [2]. Concisely, it is the average value of the area under the Precision-Recall curve over all queries.

B. Word spotting results

Before presenting the results we briefly discuss the reference systems used to compare the performance of our approach. The first system is the work of Gatos et al. [16]. Therein, a combination of word image normalization and feature extraction methods is presented for cursive handwritten word recognition. The second approach, which is described in [17], introduced the idea of adaptive zoning features for word recognition in historical, machine-printed documents. These features are extracted after adjusting the position of every zone based on local pattern information. The adjustment is performed by moving every zone towards

[1] http://www.iit.demokritos.gr/~nstam/GRPOLY-DB

[2] The trec_eval software is available at http://trec.nist.gov/trec_eval

Table I
MEAN AVERAGE PRECISION FOR VARIOUS METHODS

Method	**GRPOLY-DB** (141 queries)	**GW20** (306 queries)
Efficient Recognition [16]	39.44%	21.93%
Adaptive Zoning [17]	40.38%	22.50%
DTW [10]	56.18%	22.08%
Proposed	60.04%	37.86%

the pattern body according to the maximization of the local pixel density around each zone. The final approach is the DTW method, based on the word profiles of Rath et al. [10] for handwritten historical documents.

Following the configuration defined in Section IV-A, we compare our system with these reference systems and illustrate the results for both datasets in Table I. With respect to the first two reference systems [16], [17], we should note that they were originally created for different datasets. The method of Gatos et. al [16] was tested on the IAM benchmark, containing text written by multiple authors, while [17] was applied on historical machine printed text. The results shown in Table I indicate that their adaptation flexibility to different scripts is not trivial. As for the DTW method, it is only almost 4% worse than the proposed system in the GRPOLY-DB dataset, whereas in the GW20 benchmark, it's MAP is by far lower than that of our approach. This confirms our expectation that our system would be able to perform well in different scripts, as it treats word images as $2D$ shapes, independently of the underlying language.

V. CONCLUSION

In this paper, we propose a shape matching technique for spotting handwritten words in the presence of intra-class variability. The approach was tested in two challenging datasets and outperformed a number of QBE techniques, thereby assuring its stability across different scripts. There is, however, a tradeoff between the accuracy and computational cost of the shape matching procedure. This means that we could re-estimate the parameters of the whole system in order to increase the speed at the cost of precision.

Several aspects regarding the performance in speed remain unexplored. For instance, a faster and more accurate non-rigid point registration algorithm can be proposed, while keeping the first step of initialization as it is. Furthermore, possible extensions of the proposed approach lie on the image matching step. In a segmentation-free concept, the system is able to spot query instances in a document page, based solely on the Hough-style voting process. Its overall speed can be drastically improved by discarding the registration algorithm and selecting a bounding-box overlap percentage criterion to measure the accuracy of the detection.

ACKNOWLEDGMENT

This work has been supported by the *OldDocPro* project (ID 4717) funded by the GSRT.

REFERENCES

[1] A. Kovalchuk, L. Wolf, and N. Dershowitz, "A simple and fast word spotting method," in *proc. ICFHR*, Sep 2014, pp. 3–8.

[2] Y. Leydier, F. Lebourgeois, and H. Emptoz, "Text search for medieval manuscript images," *Pattern Recognition*, vol. 40, no. 12, pp. 3552– 3567, 2007.

[3] T. M. Rath and R. Manmatha, "Word spotting for historical documents," *International Journal on Document Analysis and Recognition*, vol. 9, no. 2-4, pp. 139–152, 2007.

[4] J. A. Rodríguez-Serrano and F. Perronnin, "A model-based sequence similarity with application to handwritten word spotting," *IEEE Trans. PAMI*, vol. 34, no. 11, pp. 2108–2120, 2012.

[5] J. Almazan, A. Gordo, A. Fornes, and E. Valveny, "Word spotting and recognition with embedded attributes," *IEEE Trans. PAMI*, vol. 36, no. 12, pp. 2552–2566, Dec 2014.

[6] V. Frinken, A. Fischer, R. Manmatha, and H. Bunke, "A novel word spotting method based on recurrent neural networks," *IEEE Trans. PAMI*, vol. 34, no. 2, pp. 211–224, Feb 2012.

[7] A. Fischer, V. Frinken, H. Bunke, and C. Suen, "Improving hmm-based keyword spotting with character language models," in *proc. ICDAR*, Aug 2013, pp. 506–510.

[8] V. Ferrari, F. Jurie, and C. Schmid, "From images to shape models for object detection," *International Journal of Computer Vision*, vol. 87, no. 3, pp. 284–303, May 2010.

[9] A. P. Giotis, D. P. Gerogiannis, and C. Nikou, "Word spotting in handwritten text using contour-based models," in *proc. ICFHR*, Sep 2014, pp. 399–404.

[10] T. M. Rath and R. Manmatha, "Word image matching using dynamic time warping," in *proc. CVPR*, vol. 2, Jun 2003, pp. 521–527.

[11] V. Lavrenko, T. M. Rath, and R. Manmatha, "Holistic word recognition for handwritten historical documents," in *proc. International Workshop on Document Image Analysis for Libraries*, Jan 2004, pp. 278–287.

[12] B. Gatos, N. Stamatopoulos, G. Louloudis, G. Sfikas, G. Retsinas, V. Papavassiliou, F. Simistira, and V. Katsouros, "GRPOLY-DB: An old Greek polytonic document image database," in *proc. ICDAR*, Aug 2015, Accepted.

[13] V. Ferrari, T. Tuytelaars, and L. V. Gool, "Object detection by contour segment networks," in *proc. ECCV*, 2006, pp. 14–28.

[14] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, Feb. 2003.

[15] B. Gatos, I. Pratikakis, and S. Perantonis, "Adaptive degraded document image binarization," *Pattern Recognition*, vol. 39, no. 3, pp. 317 – 327, 2006.

[16] B. Gatos, I. Pratikakis, K. A.L., and S. Perantonis, "Efficient off-line cursive handwritten word recognition," in *proc. IWFHR*, Oct 2006, pp. 121–125.

[17] B. Gatos, A. Kesidis, and A. Papandreou, "Adaptive Zoning Features for Character and Word Recognition," in *proc. ICDAR*, Sept 2011, pp. 1160–1164.