2011 International Conference on Document Analysis and Recognition

Browsing Heterogeneous Document Collections by a Segmentation-free Word Spotting Method

Marçal Rusiñol, David Aldavert, Ricardo Toledo and Josep Lladós
Computer Vision Center, Dept. Ciències de la Computació
Edifici O, Universitat Autònoma de Barcelona
08193 Bellaterra (Barcelona), Spain
{*marcal,aldavert,ricard,josep*}*@cvc.uab.cat*

***Abstract*—In this paper, we present a segmentation-free word spotting method that is able to deal with heterogeneous document image collections. We propose a patch-based framework where patches are represented by a bag-of-visual-words model powered by SIFT descriptors. A later refinement of the feature vectors is performed by applying the latent semantic indexing technique. The proposed method performs well on both handwritten and typewritten historical document images. We have also tested our method on documents written in non-Latin scripts.**

***Keywords*-Word Spotting, Heterogeneous Document Collections, Dense SIFT Features, Latent Semantic Indexing.**

I. Introduction

In the field of document image analysis, the problem of word spotting can be defined as the task of identifying the zones from a document image which are likely to contain an instance of a queried word without explicitly recognizing it. Word spotting techniques are particularly interesting to facilitate the browsing of large image collections in which the performance of OCR engines is still poor. Some examples might be either historical or handwritten documents, or even collections of natural scene images containing textual information.

The first attempts to tackle the word spotting problem relied on an initial layout analysis step devoted to perform word segmentation. These words were then encoded with shape signatures to be treated as one-dimensional signals. By using some DTW-based distance, the query word image was finally matched against the whole word corpus. A classical example of this framework is the work presented by Rath and Manmatha in [1].

Nowadays, the late trends in word spotting research are focused on trying to propose methods that do not require a perfect word segmentation step. In [2], a handwritten word spotting method that just needs the text lines to be segmented is presented. Given a text line, it is first normalized and a trained Hidden Markov Model (HMM) is able to perform the word spotting and the segmentation of the word within a line in a single step. In a similar fashion, Frinken et al. present in [3] a method that uses a Neural Network (NN) to perform the spotting at line level. In both cases, another advantage of the presented methods is that they are writer independent and they can perform the spotting independently of the different writing styles. However, the drawback of both methods is that they still rely on a good line segmentation and line normalization steps in order to be able to process lines as one-dimensional signals. In addition, the use of HMM or NN usually require an important volume of training data.

In [4], a word spotting methodology based on local keypoints is presented. In this work, word segmentation is not required. For a given query image, interest points are extracted and encoded by a simple descriptor based on gradient information. The word spotting is then performed by trying to locate zones of the document images with similar interest points. This retrieved zones are then filtered and only the ones sharing the same spatial configuration than the query model are returned.

Finally, another segmentation-free word spotting method is presented in [5]. In that case, the authors propose to use a sliding-window approach with a patch descriptor that encodes pixel densities. The hypothetic locations where the queried word is likely to appear are found by a template matching strategy.

Besides the segmentation problem, another interesting issue when defining a word spotting approach is the choice of the word descriptors. Projection profiles or background/ink transitions are usually selected when dealing with cursive text (as in [1], [2]) whereas features extracted from connected components are usually preferred when the documents are typewritten (as in [6], [7]). However, these features are unlikely to perform well if we change the document scenario. One of the remaining challenges in the word spotting domain is to propose methods that are able to manage heterogeneous document collections. That is, that the word features should not be ad-hoc defined for a specific kind of documents but general enough to perform well in an heterogeneous collection of documents including multiple fonts, scripts, writers, etc.

In this paper, we present a word spotting method that on the one hand does not need any word nor line segmentation step. This is achieved by using a patch-based framework. On the other hand, the proposed method is able to tackle

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.22

with heterogeneous collections. By the combined use of a bag-of-visual-words model based on SIFT descriptors and the later refinement of the patch descriptors by applying the latent semantic indexing (LSI) technique, our method performs well on both handwritten and typewritten manuscripts. Although most of the experiments have been carried out on Latin documents, we have also tested our method on other scripts such as Persian.

The remainder of this paper is organized as follows: We detail in Section II the process of feature extraction in the proposed patch-based framework. Section III is focused on the use of the latent semantic indexing technique for refining the patch description. In Section IV, we detail the retrieval step to be done when a query word is provided by the user. Section V presents the experimental setup by using a handwritten and typewritten document collection. Finally, the conclusions and a short discussion can be found in Section VI.

II. Off-line Feature Extraction

We address the word spotting problem by dividing the original document images into a set of densely sampled local patches. Each patch is then represented with a feature vector descriptor. Then, the similarity between the query image and a patch is computed as the distance between both descriptors. Taking into account these similarities, the document's zones with a higher likelihood of containing the query instance are retrieved. With such a procedure, we avoid both image preprocessing steps (i.e. binarization, slant correction, etc.) and word segmentation algorithms.

In this Section, we present the bag-of-visual-words (BoVW) model used to create the description of document patches. In other Computer Vision scenarios such as object categorization or object detection, the BoVW model has obtained a good performance compared to more complex approaches albeit its simplicity. The use of the BoVW model gives robustness to occlusions or image deformations, while the use of local descriptors adds invariance to changes of illumination or image noise. Besides, the descriptors obtained by the BoVW model can be compared using standard distances and subsequently any statistical pattern recognition technique can be applied.

A. Document Level: Visual Words Description

For each document page, we densely calculate the SIFT descriptors over a regular grid of 5 pixels by using the method presented in [8]. Three different scales using squared regions of 10, 15 and 20 pixels size are considered. These parameters are related to the font size, and in our case have been experimentally set. The selected scales guarantee that a descriptor either covers part of a character, a complete character or a character and its surroundings. As shown in [9], the larger amount of descriptors we extract from an image, the better the performance of the BoVW model is. Therefore, a dense sampling strategy has a clear advantage over approaches defining their regions by using interest points. Since the descriptors are densely sampled, some SIFT descriptors calculated in low textured regions are unreliable. Therefore, descriptors having a low gradient magnitude before normalization are directly discarded.

Once the SIFT descriptors are calculated, they are quantized into visual words by using a codebook. This codebook is obtained by clustering the descriptor feature space into k clusters. Then, the visual word associated to a descriptor corresponds to the index of the cluster that the descriptor belongs to. For the learning stage, a single page from the corpus is considered. The SIFT descriptors are extracted from this sample page, and the codebook is generated by clustering those descriptors using the k-means algorithm. In the experiments carried out in this paper, we use a codebook with dimensionality of 1500 visual words.

B. Patch Level: Feature Vector Descriptor

Once we have calculated the visual words of the document image, we split the document image into a set of overlapping local patches. These local patches have a fixed geometry of 300×75 pixels and are densely sampled at each 25 pixels. Again, these parameters are related to the document resolution, and have been experimentally set. This patch size ensures that almost all the words from the corpus will fit in a single patch. The patch displacement guarantees enough overlapping so that all the words in a document page are represented by at least one patch. Although a salient patch detection strategy will effectively reduce the amount of patches to be processed, by densely sampling them we do not make any assumption of which regions of the documents are important to the final user. The descriptor of a given patch is formed by the frequencies of the different visual words which lie within the patch. Therefore, the descriptor has the same dimensionality than the visual words codebook.

The main drawback of bag-of-words-based models is that they do not take into account the spatial distribution of the features. In order to add spatial information to the orderless BoVW model, Lazebnik et al. proposed in [10] the Spatial Pyramid Matching (SPM) method. This method roughly takes into account the visual word distribution over the patch by creating a pyramid of spatial bins.

This pyramid is recursively constructed by splitting the patch into $P_x \times P_y$ spatial bins, where P_x and P_y correspond to the number of partitions following the X and Y directions, respectively.

At each spatial bin, a different BoVW histogram is extracted. The resulting descriptor is obtained by concatenating all the BoVW histograms. Therefore, the final dimensionality M of the descriptor is

$$M = k \sum_{l=0}^{L-1} P_x^l P_y^l, \tag{1}$$

where L is the number of levels of the pyramid. Since the amount of visual words assigned to each bin is lower at higher levels of the pyramid, due to the fact that the spatial bins are smaller, the visual words contribution is weighted by $w_l = P_x^l P_y^l$, where l is the corresponding pyramid level. In our experiments, we have used a two levels SPM, with $P_x = 2$ and $P_y = 1$, resulting in 3 spatial bins and therefore a descriptor of 4500 dimensions for each patch. Using this configuration, the descriptors encode separately the left and right side of the words. This spatial information increases the whole performance of the method.

Summarizing, for each document page D_i we obtain a number of overlapping local patches p_j^i with $j \in \{0, ..., N\}$. Each patch p_j^i is characterized by a BoVW model over densely extracted SIFT descriptors quantized with a k-dimensional codebook and a SPM of two levels. Each p_j^i is then described by an M-dimensional descriptor $\mathbf{f}_j^i$.

III. Latent Semantic Indexing

Each p_j^i in our collection of patches is represented by the descriptor $\mathbf{f}_j^i$ obtained by the BoVW model presented in the previous Section. Hence, the patch corpus is represented by a feature-by-patch matrix $\mathbf{A}^i \in \mathbb{R}^{M \times N}$, where M is the descriptor dimensionality and N is the number of patches. The matrix $\mathbf{A}^i$ is then weighted by applying the *tf-idf* model. This normalization emphasizes the features that are frequent in a particular patch and infrequent in the complete patch corpus. After this normalization, we apply the LSI technique first introduced by Deerwester et al. in [11]. The motivation of using LSI is that this technique is able, given a text retrieval framework, to return results that are conceptually similar in meaning to the query even if the results do not share an important set of words with the query. In our particular methodology, the use of LSI allows us to retrieve relevant patches even if they do not contain the same exact features than the query sub-image.

The LSI model assumes that there exists some underlying semantic structure in the descriptor space. This semantic structure is defined by assigning to each patch descriptor a set of topics, which can be estimated in an unsupervised way using standard statistical techniques. The goal is to obtain a transformed space where patches having similar topics but with different descriptors will lie close. This transformed space is obtained by decomposing the feature-by-patch matrix in three matrices by a truncated Singular Value Decomposition (SVD). In order to reduce the descriptor space to K topics we proceed as follows:

$$\mathbf{A}^i \simeq \hat{\mathbf{A}}^i = \mathbf{U}_K^i \mathbf{S}_K^i \left(\mathbf{V}_K^i\right)^\top, \tag{2}$$

where $\mathbf{U}_K^i \in \mathbb{R}^{M \times K}$, $\mathbf{S}_K^i \in \mathbb{R}^{K \times K}$ and $\mathbf{V}_K^i \in \mathbb{R}^{N \times K}$. The super-index i indicates that a different LSI transformation is generated for each document separately. In this way, each of our topics model the words in a document page, allowing to work with heterogeneous document sets. In our experimental setup, we use a value of $K = 200$ topics. Note that LSI does not result in a reduction of the number of features, but a transformation from the patch descriptor space to a topic space.

IV. Retrieval Stage

At the retrieval stage, the user provides an example of the word he wants to find. This sub-image is taken as if it corresponded to a single patch within a document. Dense SIFT descriptors are thus extracted from the query image and quantized by using the codebook. Then, applying the same SPM configuration than in the document corpus, the final query descriptor $\mathbf{f}_q$ is obtained.

The first step of the retrieval is to obtain a list of patches sorted by the similarity to the query for each document page in the collection. This is accomplished by first projecting the descriptor $\mathbf{f}_q$ to each document topic space by

$$\hat{\mathbf{f}}_q^i = \mathbf{f}_q^\top \mathbf{U}_K^i \left(\mathbf{S}_K^i\right)^{-1}. \tag{3}$$

Then, we obtain the similarity list by using the cosine distance between $\hat{\mathbf{A}}^i$ and $\hat{\mathbf{f}}_q^i$ for each document in the corpus. By just considering the 200 topmost patches, we build a voting space in order to find the zones of the image having more accumulation of evidences that the queried word is likely to be found. The final retrieved zones are determined by searching for local maxima in the voting space.

V. Experimental Results

A. Datasets and Performance Evaluation

To perform the experiments, we worked with three datasets of different nature. On the one hand the George Washington (GW) dataset described in [1]. This dataset consists of 20 handwritten pages with a total of 4860 words. On the other hand, the Lord Byron (LB) dataset consists on 20 typewritten pages from a 1825 book[1] with a total of 4988 words. The ground-truth for both collections contains the word transcriptions and their bounding-boxes. Finally, the Persian (PE) dataset consists of 20 typewritten pages from a 1848 book written in Persian. Unfortunately, we do not have any ground-truth for this dataset, and will only be used as a proof-of-concept that the method is able to work with non-Latin scripts by showing qualitative results.

Concerning the performance evaluation of the method, we will show the precision and recall curves and the mean average precision indicator. In order to compute these measures, we need to define a notion of relevance from the returned results. For a given query, a returned zone will be labeled as relevant if it overlaps more than a 50% of one of the bounding-boxes in the ground-truth containing the same queried word.

[1]A binary and cleaned version of the book can be downloaded from http://books.google.com/books?id=u6poWVzCIWsC

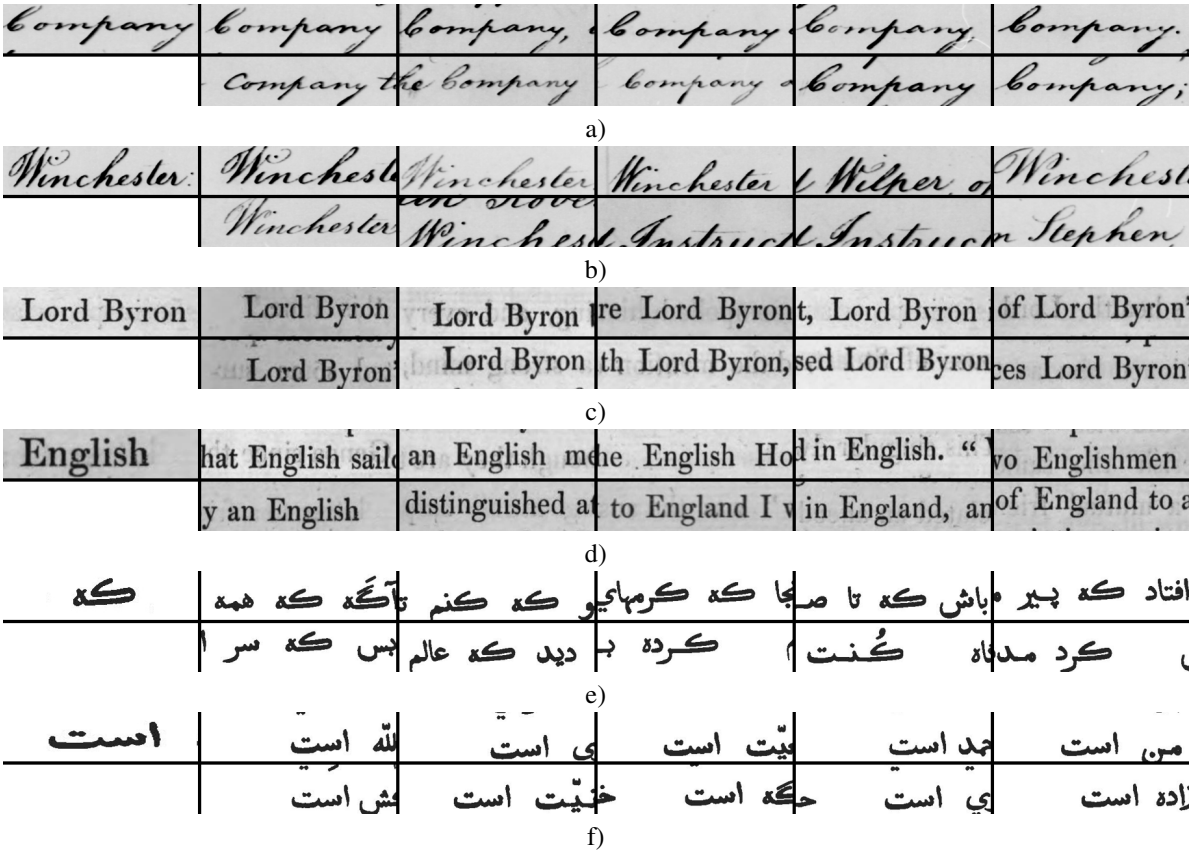

Figure 1. Qualitative results. Query and top ten retrieved words. a) and b) GW dataset. c) and d) LB dataset. e) and f) PE dataset.

B. Results

We present in Figure 1 some qualitative results of the method in the three different datasets for a couple of queries. Note that the proposed method is able to work with queries formed by multiple words. Note also that the false positives are still visually similar to the queries.

Concerning the quantitative evaluation, we used all the words in GW and LB datasets as queries. For the GW dataset the mean average precision was 30.42% and the mean recall was 71.1% whereas for the LB dataset the mean average precision was 42.83% and the mean recall was 85.86%. For the proposed spotting method, we can see that most of the relevant words are retrieved, even if the ranking is not that good due to false positives. Analyzing the results, we noticed that the performance of the system was highly related to the length of the queried words. We can see in Figure 2 the precision and recall curves for different word lengths. The mean average precision and mean recall results depending on the query length are shown in Figure 3. As we can appreciate, if we do not consider small words as queries (usually corresponding to stop-words) the system performance presents an important increase. For example, considering words larger than 5 characters, for the GW dataset the mean average precision is increased until reaching a 53.76% and the mean recall results in a 93.39% whereas for the LB dataset the mean average precision results in a 70.23% and the mean recall is increased until reaching a 98.32%. Typewritten documents perform a little bit better than the handwritten ones basically due to the variability of word shapes in the handwritten context.

Regarding the time complexity, the process of querying a word against a single page (that is, indexing more than 9200 patches) takes in average 340ms. in our prototype implementation using Matlab and Python.

VI. Conclusions and Future Work

In this paper we have presented a word spotting method that does not rely on any previous segmentation step. We have shown that the proposed method can be used in heterogeneous collections since it yields good results in both handwritten and typewritten documents. Our method can also be used with non-Latin scripts and does not require any preprocessing step of noise removal or normalization. The presented method combines the use of a bag-of-visual-words model based on SIFT descriptors and the later refinement of the descriptors by using the latent semantic indexing technique. A final voting scheme aims to locate the zones within document images where the queried word is likely to

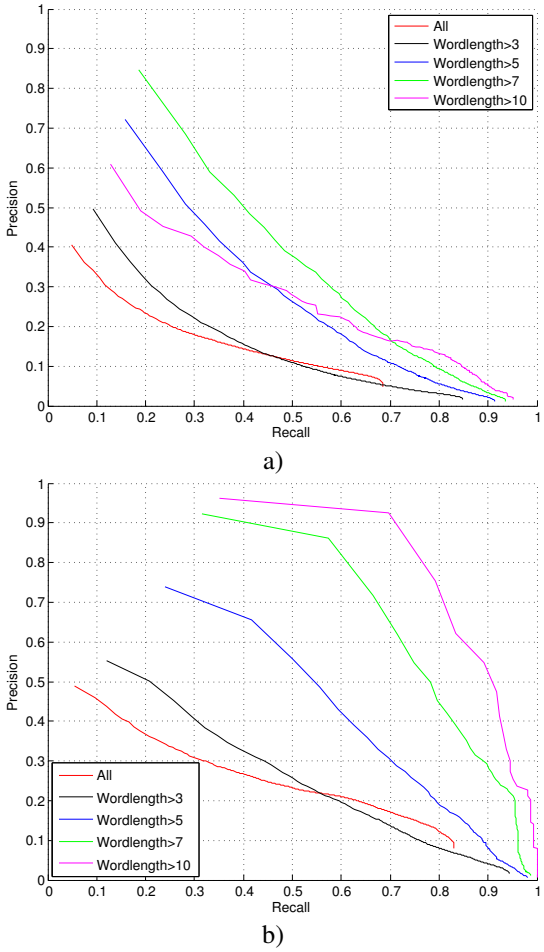

Figure 2. Precision and recall curves for different word lengths. a) GW dataset. b) LB dataset.

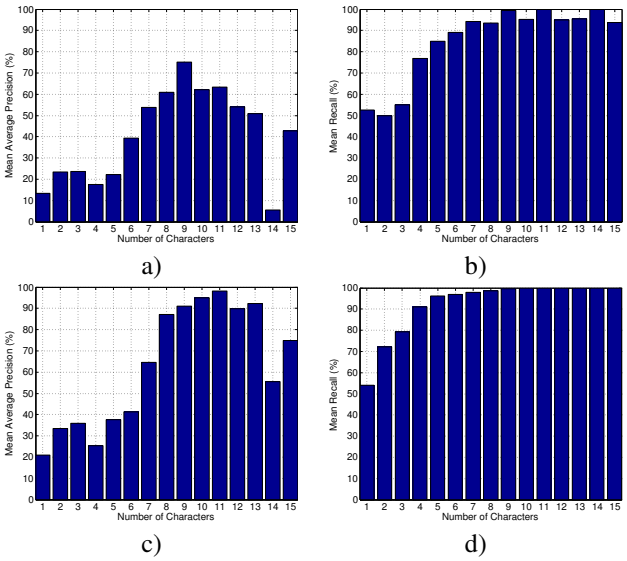

Figure 3. Quantitative results for different word lengths. a) and b) mean average precision and mean recall for GW dataset respectively. c) and d) mean average precision and mean recall for LB dataset respectively.

appear. As future research lines we would like to combine the use of the proposed method with some approximate nearest neighbor technique in order to avoid the one-to-one distance computation.

Acknowledgments

This work has been partially supported by the Spanish Ministry of Education and Science under projects TIN2008-04998, TIN2009-14633-C03-03, TRA2010-21371-C03-01, Consolider Ingenio 2010: MIPRCV (CSD200700018) and the grant 2009-SGR-1434 of the Generalitat de Catalunya. Special thanks go to M. Rouhani for his assistance with the Persian documents.

References

[1] T. Rath and R. Manmatha, "Word spotting for historical documents," *Int. J. Doc. Anal. Recogn.*, vol. 9, no. 2–4, pp. 139–152, 2007.

[2] A. Fischer, A. Keller, V. Frinken, and H. Bunke, "HMM-based word spotting in handwritten documents using subword models," in *Proc. of the Int. Conf. on Pattern Recognition*, 2010, pp. 3416–3419.

[3] V. Frinken, A. Fischer, and H. Bunke, "A novel word spotting algorithm using bidirectional long short-term memory neural networks," in *Artifitial Neural Networks in Pattern Recognition*, ser. LNCS, 2010, vol. 5998, pp. 185–196.

[4] Y. Leydier, A. Ouji, F. LeBourgeois, and H. Emptoz, "Towards an omnilingual word retrieval system for ancient manuscripts," *Pattern Recognit.*, vol. 42, no. 9, pp. 2089–2105, 2009.

[5] B. Gatos and I. Pratikakis, "Segmentation-free word spotting in historical printed documents," in *Proc. of the Int. Conf. on Document Analysis and Recognition*, 2009, pp. 271–275.

[6] S. Bai, L. Li, and C. Tan, "Keyword spotting in document images through word shape coding," in *Proc. of the Int. Conf. on Document Analysis and Recognition*, 2009, pp. 331–335.

[7] E. Şaykol, A. Sinop, U. Güdükbay, O. Ulusoy, and A. Çetin, "Content-based retrieval of historical ottoman documents stored as textual images," *IEEE Trans. on Im. Proc.*, vol. 13, no. 3, pp. 314–325, 2004.

[8] B. Fulkerson, A. Vedaldi, and S. Soatto, "Localizing objects with smart dictionaries," in *Computer Vision - ECCV*, ser. LNCS, 2008, vol. 5302, pp. 179–192.

[9] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int. J. Comput. Vision*, vol. 73, no. 2, pp. 213–238, 2007.

[10] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. of the Conf. on Computer Vision and Pattern Recognition*, 2006, pp. 2169–2178.

[11] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *J. Am. Soc. Inform. Sci.*, vol. 41, no. 6, pp. 391–407, 1990.