Boosting the Handwritten Word Spotting Experience by Including the User in the Loop

Marçal Rusiñol [*], Josep Lladós

Computer Vision Center, Dept. Ciències de la Computació
Edifici O, Univ. Autònoma de Barcelona, 08193 Bellaterra (Barcelona), Spain

Abstract

In this paper, we study the effect of taking the user into account in a query-by-example handwritten word spotting framework. Several off-the-shelf query fusion and relevance feedback strategies have been tested in the handwritten word spotting context. The increase in terms of precision when the user is included in the loop is assessed using two datasets of historical handwritten documents and two baseline word spotting approaches both based on the bag-of-visual-words model. We finally present two alternative ways of presenting the results to the user that might be more attractive and suitable to the user's needs than the classic ranked list.

Key words: Handwritten word spotting, Query by example, Relevance feedback, Query fusion, Multidimensional scaling

1 Introduction

Handwritten word spotting can be defined as the task of retrieving a set of locations from document images where a given word is likely to appear without explicitly transcribing all the handwritten words. Within the field of document image analysis, handwritten word spotting has received a lot of attention and is today a quite mature research topic. The kickoff word spotting approaches applied to handwritten document images were presented in the mid 90's [27,36]. Research in this topic has been mainly motivated by the huge amounts of cultural heritage assets that are still nowadays confined

[*] Corresponding author. Tel.: +34-93-581-40-90; fax:+34-93-581-16-70.
Email addresses: marcal@cvc.uab.es (Marçal Rusiñol), josep@cvc.uab.es (Josep Lladós).

in digital libraries without any effective framework providing accessibility to those contents.

We can broadly define a taxonomy of handwritten word spotting methods that distinguishes two main families. The first group consists of the word spotting methods that are aimed at detecting just a closed set of predefined words. These methods usually entail a training step in which a model for each of the possible words that the user wants to spot is built. Usually, these methods are preferred in multi-writer scenarios, where the user wants to assess whether a document contains one of the predefined keywords or not. Some examples of this family are the works proposed by Rodríguez-Serrano and Perronnin in [32], by Fischer et al.[11], by Choisy [6], by Edwards et al. in [9] or Chan et al. in [5] in which Hidden Markov Models (HMM) are used to model handwritten words, or the work proposed by Frinken et al. in [13] and in [14] in which Neural Networks (NN) are used to build the models. Such methods are usually known as *learning-based* methods since they entail the use of machine learning techniques.

One the other hand, there is another set of word spotting methods which are more retrieval-oriented. In that case, given a document collection which has been indexed off-line, the user casts a word query and he wants to retrieve from the image collection similar instances of that word. In that case there is no training stage involved and the user can query whatever word he wants. Most of the early-days works on handwritten word spotting followed this paradigm, as the seminal publication of Manmatha et al. in [27] or the work of Syeda-Mahmood [36]. Such paradigm is often known as *query-by-example* methods, and they are based on matching the word provided by the user with the rest of words in the collection. Many recent handwritten word spotting methods that follow this paradigm have been proposed such as the works by Fornés et al. in [12], Lladós et al. in [25], Zhang et al. in [39], Terasawa and Tanaka in [37] or Rusiñol et al. in [34]. We target our work in the query-by-example handwritten word spotting methods.

Query-by-example handwritten word spotting methods can be understood as a particular case of Content-Based Image Retrieval (CBIR), in which given an image collection (of handwritten words in our case) and a query image we want to retrieve the most similar image in terms of contents (in our case the actual textual contents). Although these word spotting methods are a particular application of the information retrieval (IR) field, very few works have taken advantage of common strategies that have been used within the IR community for long time. A clear example is the lack of word spotting methods that include the user in the loop. Just some works like the method by Bhardwaj et al. [3] or the one by Cao et al. [4] propose to include a relevance feedback step. They both use the Rocchio's [31] well-known relevance feedback method and they both show significant improvements when including this feedback from the user. Similar conclusions were drawn in the case of typewritten word spotting in the work presented by Konidaris et al. [21] and Kesidis et al. [19].

We present in this paper a study on the effect of taking the user into account in a handwritten word spotting framework. We test in this paper two different approaches, namely, query fusion and relevance feedback. The former consists of asking to the user to cast several queries instead of a single one and somehow combine the results. The latter consists of retrieving the similar words from the dataset and asking to the user to provide some feedback about which results were correct and which were incorrect. This relevance feedback allows to provide an enhanced result list in a subsequent iteration. Several off-the-shelf IR methods are applied in the word spotting context. The increase in terms of precision is assessed using two datasets of historical handwritten documents and two baseline word spotting approaches both based on a bag-of-visual-words model. This paper is an extension of a previous conference version [35]. We have substantially extended its contents by proposing a new baseline method, adding four additional score normalization strategies and by finally introducing two different alternative ways of visualizing the spotting results.

The remainder of this paper is organized as follows. We overview in Section 2 the baseline handwritten word spotting methods. Section 3 is focused on the query fusion experiments whereas Section 4 deals with relevance feedback. In Section 5 we present the document image datasets and the evaluation measures. We then provide in Section 6 the experimental results. In Section 7 we propose the two alternative results visualization options. We conclude and present some discussion on Section 8.

2 Baseline Bag-of-Visual-Words Methods

In this section, we give the details of our word spotting baseline methods. Here, we assume that the words in the document pages have been previously segmented by a layout analysis step. Both the queries and the items in the database are thus segmented word snippets. The way we describe those word images is based on the bag-of-visual-words (BoVW) model powered by either SIFT [26] or Shape Context [2] descriptors. We start with a clustering of the descriptors to build a codebook. Once we have the codebook, word images are encoded by the BoVW model. In a last step, in order to produce more robust word descriptors, we add some coarse spatial information to the orderless BoVW model. Let us first detail the baseline system using SIFT features and subsequently the one using the shape context descriptor.

2.1 SIFT features

The first baseline consisting on a BoVW model powered by SIFT features was proposed in [34], and the exact parametrization we use here has been compared against a number of alternate handwritten word representations in [24]. We refer the interested reader to [24] for an exhaustive description of the representation method.

For each word image in the reference set, we densely calculate the SIFT descriptors over a regular grid by using the method presented by Fulkerson et al. in [15]. Three different SIFT descriptor scales are considered. The grid and scale parameters are dependent on the word sizes, and in our case have been experimentally set. We can see in Figure 1 an example of dense SIFT features extracted from a word image. Because 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.

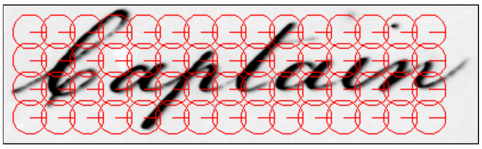


Fig. 1. Dense SIFT features extracted from a word image.

Once the SIFT descriptors are calculated, by clustering the descriptor feature space into k clusters we obtain the codebook that quantizes SIFT feature vectors into visual words. We use the k-means algorithm to perform the clustering of the feature vectors. In this work, we use a codebook with dimensionality of $k = 20.000$ visual words.

For each of the word images, we extract the SIFT descriptors, and we quantize them into visual words with the codebook. Then, the visual word associated to a descriptor corresponds to the index of the cluster that each descriptor belongs to. The BoVW feature vector for a given word snippet is then computed by counting the occurrences of each of the visual words in the image.

However, one of the main limitations of the 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. [23] proposed the Spatial Pyramid Matching (SPM) method. This method roughly takes into account the word distribution over the image by creating a pyramid of spatial bins.

This pyramid is recursively constructed by splitting the images in spatial bins following the vertical and horizontal axis. 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 of the descriptor is determined by the number of levels used to build the pyramid.

In our experiments, we have adapted the idea of SPM to be used in the context of handwritten word representation. We use the SPM configuration presented in Figure 2 where two different levels are used. The first level is the whole word image and in the second level we divide it in its right and left part and its upper, central and lower parts. With this configuration we aim to capture information about the ascenders and descenders of the words as well as information about the right and left parts of the words. Since we used a two levels SPM with 7 spatial bins, we therefore obtain a final a descriptor of 140.000 dimensions for each word image.

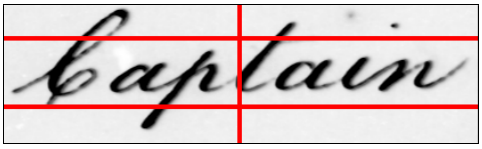


Fig. 2. Second level of the proposed SPM configuration. Ascenders and descenders information and right and left parts of the words is captured.

2.2 Shape Context Features

As a second baseline system we propose to build the BoVW model in terms of shape context features [2]. The idea of aggregate shape context descriptors into a bag-of-words representation was originally proposed by Mori et al. in [28]. Shape context descriptors have also been proven to yield good results to represent words [25].

In order to extract the shape context descriptors from handwritten words we have applied some preprocessing steps. First word snippets are binarized by using the Otsu's method and the edges from the binary image are extracted. These edge points are then equally-spaced sampled and we end up with n points that roughly describe the word's shape. We can see an example of those steps in Figure 3.

Following the original proposal of the shape context descriptor a log-polar binning is centered at each of the considered points and a histogram accumulates the amount of points that fall within each bin. In our experimental setup we used 12 angular and 5 distance bins delivering a 60-dimensional descriptor for each of the sampled points. We can see an example of the shape context descriptor centered in one point in Figure 4.

The codebook quantizing the shape context space is obtained by clustering

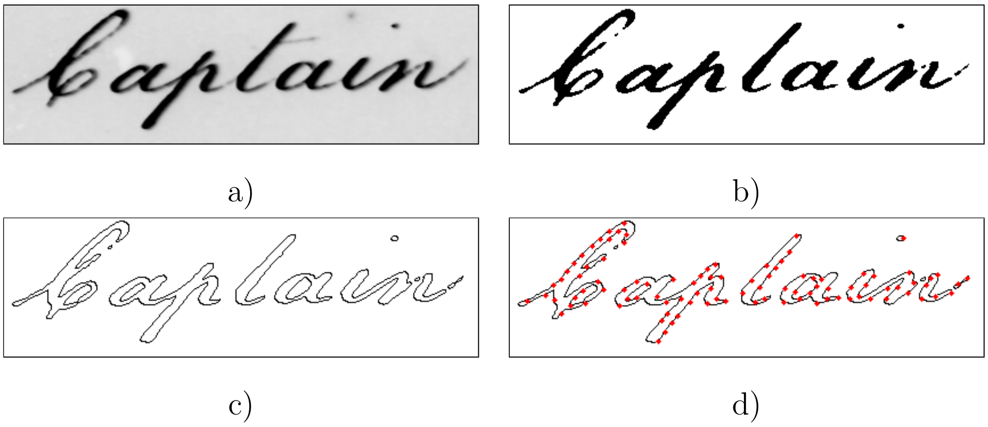
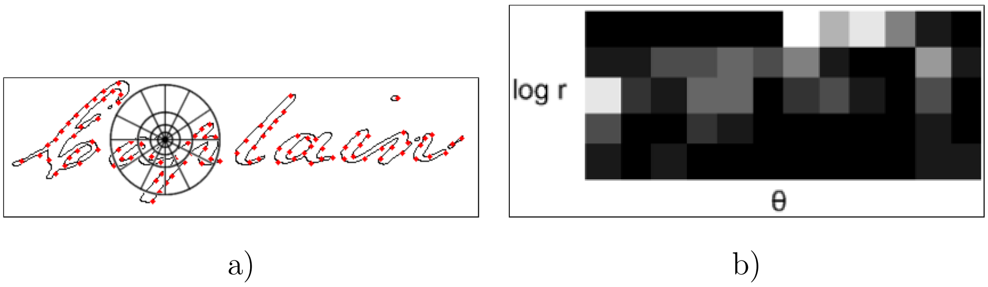

a) b) c) d)

Fig. 3. Preprocessing steps to compute the shape context descriptor. a) Original image, b) binarized version by using Otsu's method, c) its edge and d) equally spaced sampled points from the word edges.



a) b)

Fig. 4. Example of the shape context descriptor. a) the log-polar bins centered at a given sampled point of the shape, b) its corresponding histogram values.

the descriptor feature space into k clusters by the k-means algorithm. For this approach, we used a much smaller codebook than the previous baseline, having dimensionality of $k = 2^{13}$ visual words. Since the shape context descriptor already encodes the spatial distribution of the points within a word, we have not observed significant improvements when using spatial pyramids, so the original BoVW model is kept.

2.3 Normalization and Word Retrieval by Similarity

Finally, all the word descriptors are normalized by using the L_2-norm. In order to assess whether a query word image is similar or not to a given word image in the collection, we use the cosine distance between its feature vectors $\mathbf{f}_q$ and $\mathbf{f}_c$ respectively.

$$d(\mathbf{f}_q, \mathbf{f}_c) = 1 - \frac{\mathbf{f}_q \cdot \mathbf{f}_c}{\parallel \mathbf{f}_q \parallel \parallel \mathbf{f}_c \parallel} \tag{1}$$

3 Query Fusion

One of the classic ways to enhance the retrieval results in an IR scenario is to cast several queries instead of a single one and somehow combine the results. This is particularly interesting when the queries come from different modalities. In the case of word spotting, asking the user to provide several instances of the sought word might be advantageous in order to overcome the variability of handwritten words.

We have tested three different fusion strategies. One *early fusion* strategy where the queries feature vectors are combined before performing the retrieval step and two *late fusion* strategies where we perform as many retrieval rounds as different queries the user provides and the ranked lists are finally combined somehow. Let us detail these three fusion methods.

- **Early fusion** is achieved by simply averaging the query image descriptors and then normalizing again by the L_2-norm.
- **CombMAX** is a late fusion method that assigns to the words in the collection its maximum score across the different casted queries. The final resulting list is then re-sorted in terms of these maximum obtained scores.
- **Borda Count** is also a late fusion method in which the topmost image on each ranked list gets n votes, where n is the dataset size. Each subsequent rank gets one vote less than the previous. The final ranked list is obtained by adding all the votes per image and re-sorting.

By these three different strategies we believe we cover a wide range of fusion families. On the one hand the early fusion method combines the queries in the descriptor space. On the other hand the late fusion strategies combine the resulting lists from different retrieval rounds. In that case, the CombMAX method takes into account the obtained scores across queries whereas the Borda Count ignores the scores and focuses on the absolute ranking of the collection words.

However, in the related literature it has been noted that the late fusion strategies that combine scores might benefit from a score normalziation step [20,29]. Here we have used four different score normalization strategies used in combination with the CombMAX that given a set of matching scores S_k with $k = 1, 2, \ldots, n$ compute their normalized scores S'_k.

- **Minmax**: applies a scaling factor and transforms the scores in a common range $[0, 1]$. Being S_{min} and S_{max} the minimum and maximum of the scores respectively,

  $$S'_k = \frac{S_k - S_{min}}{S_{max} - S_{min}}.$$

  As noted in [18], such method is highly sensitive to outliers in the data used for estimation.

- **Z-score**: is the most common normalization score technique. It is computed using the arithmetic mean μ and standard deviation σ of the given data.

$$S'_k = \frac{S_k - \mu}{\sigma}.$$

By using the arithmetic mean and standard deviation, the method is also sensitive to outliers.

- **Tanh**: is a more robust and efficient score normalization technique that also takes into account the mean and standard deviation.

$$S'_k = \frac{1}{2}\left\{\tanh\left[0.01 \cdot \left(\frac{S_k - \mu}{\sigma}\right)\right] + 1\right\}.$$

- **MAD**: the median and median absolute deviation are insensitive to outliers and the points in the extreme tails of the distribution [18].

$$S'_k = \frac{S_k - median}{MAD},$$

where $MAD = median(|S_k - median|)$.

We will test the influence of such score normalization techniques used together with the CombMAX late fusion strategy.

4 Relevance Feedback

The most natural way to take into account the user in an IR application is by means of relevance feedback [1]. After an initial retrieval step, the user is asked to provide some feedback about which results were correct and which were incorrect. This feedback about relevance allows to provide an enhanced result list in the subsequent iterations.

Here, we have tested three different relevance feedback methods from two different families. The Rocchio and the Ide methods, are relevance feedback algorithms that follow the idea of query reformulation whereas the relevance score method is a re-ranking method. Relevance feedback methods that follow the idea of query reformulation try to find, given the relevance assessments, a new query point in the vector domain that is closer to the positive samples and farther to the negative ones than the original query point. On the other hand, re-ranking methods, such as the relevance score method, try to reorganize the original resulting list in terms of the relevance assessments without casting any new query. Let us detail these three relevance feedback methods.

4.1 Rocchio's Algorithm

The Rocchio's algorithm [31] is one of the most widely used relevance feedback strategies in the IR field. To our best knowledge it is the only relevance feedback strategy that has been used in the context of word spotting [3,4,19,21]. At each relevance feedback iteration, the Rocchio's algorithm computes a new query point in the descriptor space aiming to incorporate relevance feedback information into the vector space model. The modified query vector $\mathbf{f}_q^m$ is computed as

$$\mathbf{f}_q^m = \alpha \mathbf{f}_q^o + \frac{\beta}{|D_r|} \sum_{\mathbf{d}_j \in D_r} \mathbf{d}_j - \frac{\gamma}{|D_n|} \sum_{\mathbf{d}_j \in D_n} \mathbf{d}_j, \tag{2}$$

where $\mathbf{f}_q^o$ is the original query vector, and D_r and D_n the sets of relevant and non-relevant handwritten word images that the user has marked respectively. α, β and γ are the associated weights that shape the modified query vector with respect to the original query, the relevant and non-relevant items. In our experimental setup we have experimentally set the following values $\alpha = 1$, $\beta = 0.75$ and $\gamma = 0.25$.

4.2 Ide Dec-hi Method

The Ide dec-hi method [17] is a variant of the Rocchio's algorithm usually known to perform slightly better in most of the IR scenarios. Instead of considering all the non-relevant items, it just takes into account the topmost ranked non-relevant item d_{non} in order to compute the modified query vector as

$$\mathbf{f}_q^m = \alpha \mathbf{f}_q^o + \beta \sum_{\mathbf{d}_j \in D_r} \mathbf{d}_j - \gamma \mathbf{d}_{non}. \tag{3}$$

In our setup we experimentally set the weighting values to $\alpha = \beta = \gamma = 1$.

4.3 Relevance Score

Finally, the relevance score algorithm presented in [16] by Giacinto and Roli is a re-ranking method. The idea behind the algorithm is that for each word image in the resulting list we assign the ratio between the nearest relevant and the nearest non-relevant word images as the new score for this particular image. The relevance score RS is computed as follows:

$$RS(\mathbf{x}, (D_r, D_n)) = \left(1 + \frac{\min_{\mathbf{d}_j \in D_r} d(\mathbf{x}, \mathbf{d}_j)}{\min_{\mathbf{d}_j \in D_n} d(\mathbf{x}, \mathbf{d}_j)}\right)^{-1}, \tag{4}$$

where $\mathbf{x}$ is the feature vector of any image in the dataset and $d(\mathbf{x}, \mathbf{d}_j)$ is the cosine distance between two handwritten word descriptors previously described in Equation 1. The new resulting list is obtained by re-ranking the word list in terms of their relevance scores.

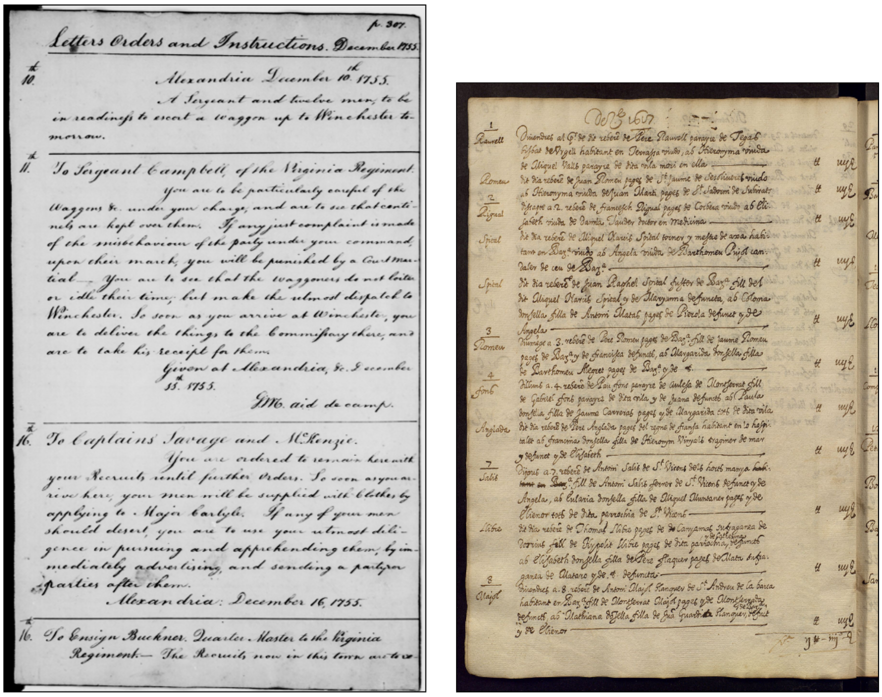
5 Datasets and Evaluation Measures

To perform the experiments, we used two datasets of handwritten documents that are accurately segmented and transcribed. All the words having at least three characters and appearing at least ten times in the collections were selected as queries. The first image corpus (GW dataset) consists of a set of 20 pages from a collection of letters by George Washington [30]. It has a total of 4860 segmented words with 1124 different transcriptions. That is 1847 word snippets that are taken as queries, and that correspond to 68 different words. The second evaluation corpus (BCN dataset) contains 27 pages from a collection of marriage registers from the Barcelona Cathedral [10] having 6544 word snippets with 1751 different transcriptions. In that collection we use 514 queries from 32 different words. We can see an example of both datasets in Figure 5.

In order to evaluate the performance of the different user interaction methods in a word spotting framework we have chosen to report the mean average precision mAP measure [38]. Given the retrieved and relevant sets to a query, ret and rel respectively, the mean average precision is computed using each precision value after truncating at each relevant item in the ranked list. For a given query, let $\mathrm{r}(n)$ be a binary function on the relevance of the n-th item in the returned ranked list and $P@n$ the precision considering only the n topmost results returned by the system. The mean average precision obtained for the Q queries is then defined as follows:

$$mAP = \frac{\sum_{q=1}^{Q} \frac{\sum_{n=1}^{|ret|} (P@n \times \mathrm{r}(n))}{|rel|}}{Q}. \tag{5}$$

In order to assess the statistical significance of the improvement when using either query fusion or relevance feedback against the baseline systems, we have computed a paired-sample t-test at the 0.01 significance level.



p. 307.

Letters Orders and Instructions. December 1755.

10th. Alexandria December 10th 1755.

A Sergeant and twelve men, to be in readiness to escort a waggon up to Winchester tomorrow.

11th. To Sergeant Campbell, of the Virginia Regiment.

You are to be particularly careful of the Waggons &c. under your charge, and are to see that centinels are kept over them. If any just complaint is made of the misbehaviour of the party under your command, upon their march, you will be punished by a Court Martial — You are to see that the waggoners do not loiter or idle their time, but make the utmost dispatch to Winchester. So soon as you arrive at Winchester, you are to deliver the things to the Commissary there, and are to take his receipt for them.

Given at Alexandria, &c. December 11th 1755.

GW. aid de camp.

16th. To Captains Savage and McKenzie.

You are ordered to remain here with your Recruits until further Orders. So soon as you arrive here, your men will be supplied with Clothes by applying to Major Carlyle. If any of your men should desert, you are to use your utmost diligence in pursuing and apprehending them, by immediately advertising, and sending a party or parties after them.

Alexandria: December 16, 1755.

16th. To Ensign Buckner. Quarter Master to the Virginia Regiment.— The Recruits now in this town are to

Dese 1617

1 Rauvell Divendres al Q.t de dit reberé de Pere Rauvell parayre de Tegas bisbat de Vrgell habitant en Terrassa viudo, ab Hieronyma viuda de Miquel Valls parayre de dita vila mori en ella — tt iiij ℓ

Romeu dit dia reberé de Juan Romeu pages de St. Jaume de Sesoliveres viudo ab Hieronyma viuda de Juan Marti pages de St. Sadorni de Subirats — tt iiij ℓ

2 Rigual Dissapte a 2. reberé de Francesch Rigual pages de Corbera viudo ab Elisabeth viuda de Damia Sauder doctor en Medicina — tt iiij ℓ

Spital dit dia reberé de Miquel Marcis Spital torner y mestre de arxa habitant en Barq.a viudo ab Angela viuda de Barthomeu Pujol candaler de ceu de Barq.a — tt iiij ℓ

Spital dit dia reberé de Juan Raphael Spital fuster de Barq.a fill del dit Miquel Marcis Spital y de Maryanna defuncta, ab Coloma donsella filla de Antoni Matas pages de Pierola defunct y de Angela — tt iiij ℓ

3 Romeu Diumenge a 3. reberé de Pere Romeu pages de Barq.a fill de Jaume Romeu pages de Barq.a y de Francisca defuncts, ab Margarida donsella filla de Barthomeu Alegret pages de Barq.a y de ‡. — tt iiij ℓ

4 Fons Dilluns a 4. reberé de Pau Fons parayre de Anlesa de Montserrat fill de Gabriel Fons parayre de dita vila y de Juana defuncts ab Paula donsella filla de Jaume Carreras pages y de Margarida tots de dita vila — tt iiij ℓ

Anglada dit dia reberé de Pere Anglada pages del regne de Fransa habitant en lo hospital ab Francina donsella filla de Hieronym Viryals traginer de mar y defunct y de Elisabeth — tt iiij ℓ

7 Salis Dijous a 7. reberé de Antoni Salis de St. Vicens dels horts manya ~~habitant en Barq.a~~ fill de Antoni Salis ferrer de St. Vicens defunct y de Angela, ab Eularia donsella filla de Miquel Muntaner pages y de Elienor tots de dita parrochia de St. Vicens — tt iiij ℓ

Llibre dit dia reberé de Thomas Llibre pages de ~~de~~ Canyamas sufraganea de Dosrius fill de Hypolit Llibre pages de dita parrochia, y de Catherina defuncts ab Elisabeth donsella filla de Pere Flaquer pages de Mata sufraganea de Mataro y de ‡ defuncta — tt iiij ℓ

8 Majol Divendres a 8. reberé de Antoni Majol flanquer de St. Andreu de la barca habitant en Barq.a fill de Montserrat Majol pages y de Montserrada de Barq.a defuncts, ab Mathiana donsella filla de Jua Guardiola flanquer, defunct y de Elienor — tt iiij ℓ

a) b)

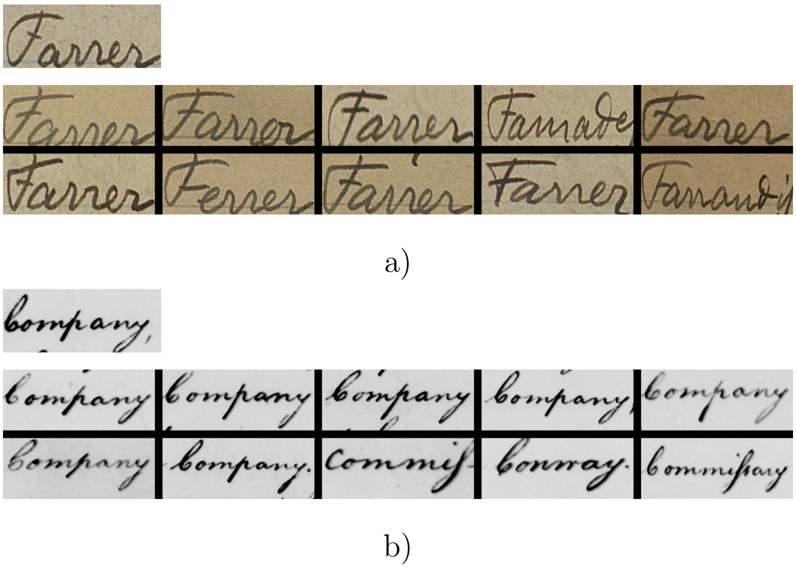
Fig. 5. Example of pages from the a) George Washington and b) Barcelona Cathedral collections.

6 Experimental Results

First, we can see some qualitative results for both collections in Figure 6. Although some false positives appear in the first ten responses, it is interesting to notice that this false positive words are in most of the cases similar to the query in terms of shape. In Figure 6, when asking for the word ***Farrer*** in the Barcelona Cathedral collection, we obtain similar results such as ***Ferrer***, ***Famades*** or ***Farrandis***. In the case of the George Washington collection, the behavior is similar. When asking for the word Company we obtain as false alarms similar words as ***Conway*** or ***Commissary***.

6.1 Query Fusion

In order to test the fusion methods we ask the user to cast three simultaneous queries to the system. For each collection all the possible combinations of three queries for all the word classes are tested and the mAP averaged. We can observe the obtained results in Table 1. We can see that all the fusion methods outperform the baseline methods in both collections except the Borda Count method in the BCN collection when using the shape context descriptor, which might indicate that with such descriptors, the sole use of the



a)

b)

Fig. 6. Queries and qualitative results for the a) BCN collection and b) GW collection when using the baseline powered by SIFT descriptors

Table 1
mAP for various query fusion strategies. Displayed in bold the best fusion strategy. Triangles denote either improvement or deterioration and statistical significance versus the baseline when filled.

		Baseline	Early Fusion	CombMAX	Borda
SIFT	GW	0.4219	**0.50409**▲	0.46813▲	0.44749▲
	BCN	0.3004	**0.43471**▲	0.38803▲	0.39929▲
Shape Context	GW	0.40289	**0.47064**▲	0.46148▲	0.4141▲
	BCN	0.28821	**0.31975**▲	0.30596△	0.26599▼

ranking is not enough to provide a robust fusion of different queries, whereas when using the scores a performance increase can be observed. In addition, early fusion performs better than the two late fusion strategies for both collections and both baselines as well. There are no significant differences between the two late fusion strategies in the baseline using SIFT features whereas in the baseline using shape contexts the combMAX strategy outperforms Borda Count, indicating that the scores associated to the words convey a pertinent information in that case.

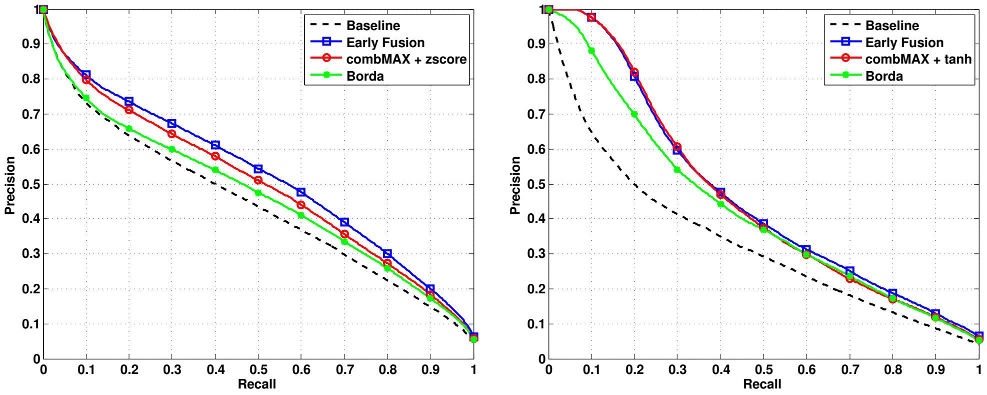
We have used the raw scores in order to apply the late fusion strategy CombMAX in Table 1. We present in Table 2, the obtained results when applying a score normalization function before applying the CombMAX method. Besides the Minmax method, that in some scenarios perform worse than the raw CombMAX strategy, the rest of the score normalization functions outperform the sole use of CombMAX, although there is no clear advantage in using one or the other, since the best performances depend on the scenario. However, the tanh normalization score provides a statistical significant improvement in all the experiments. We provide in Figure 7 the precision and recall curves when

Table 2

mAP for various score normalization strategies when used with the combMAX fusion approach. Displayed in bold the best score normalization function. Triangles denote either improvement or deterioration and statistical significance versus the raw use of CombMax when filled.

		Baseline	CombMAX	Minmax	Zscore	Tanh	MAD
SIFT	GW	0.4219	0.46813	0.4672$^{\triangledown}$	**0.4797**$^{\blacktriangle}$	0.47869$^{\blacktriangle}$	0.47757$^{\blacktriangle}$
	BCN	0.3004	0.38803	0.41723$^{\blacktriangle}$	0.39394$^{\triangle}$	**0.42786**$^{\blacktriangle}$	0.38855$^{\triangle}$
Shape Context	GW	0.40289	0.46148	0.46232$^{\triangle}$	0.47147$^{\blacktriangle}$	0.47051$^{\blacktriangle}$	**0.47303**$^{\blacktriangle}$
	BCN	0.28821	0.30596	0.31257$^{\triangle}$	**0.33609**$^{\blacktriangle}$	0.32324$^{\blacktriangle}$	0.33444$^{\blacktriangle}$

using SIFT features for both datasets, and using the best score normalization function in each scenario.



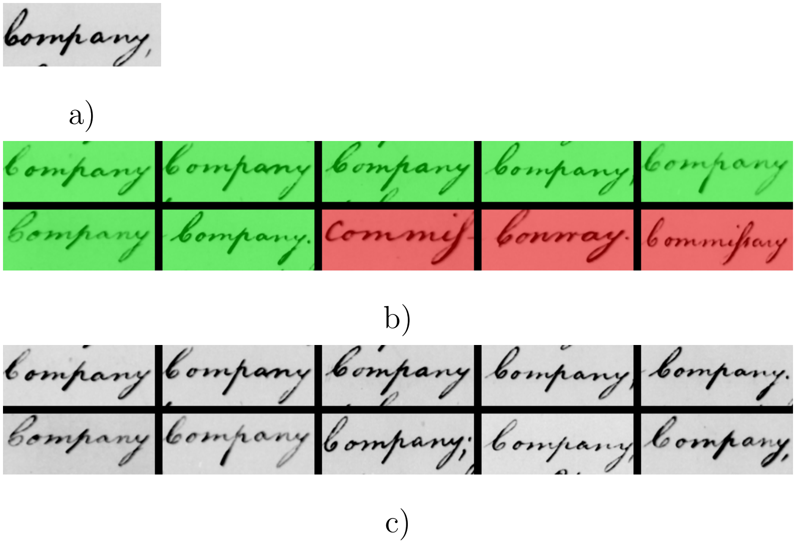
a) b)

Fig. 7. Precision Recall curves for the a) GW and b) BCN collections when using SIFT features for the different fusion strategies.

6.2 Relevance Feedback

We can see in Figure 8 an example of the relevance feedback strategy behavior. Given a query the system returns a ranked list. The user is then asked to mark a certain amount of returned words as being relevant or non-relevant (in green and red respectively in the figure). For the query reformulation strategies a new query is created and casted again in order to obtain an improved ranked list whereas in the re-ranking strategy we just re-rank the already obtained list. In order to test the three relevance feedback methods, we ask the user to give relevance on the first ten retrieved images. We guarantee that at least one positive and one negative sample are provided by taking the topmost ranked from each category.

We can see in Table 3 the quantitative obtained results. We can observe that when using any of the relevance feedback strategies, the results clearly outperform the baselines handwritten word spotting systems for both collections.



a)

b)

c)

Fig. 8. Qualitative example of applying the Rocchio relevance feedback method. a) The query word, b) the original retrieval with relevance assessments by the user (in green relevant words in red non-relevant words), c) returned results after a relevance feedback round.

Table 3
mAP for various relevance feedback methods. Displayed in bold the best feedback strategy. Triangles denote either improvement or deterioration and statistical significance versus the baseline when filled.

		Baseline	Rocchio	Ide	RS
SIFT	GW	0.4219	0.48215▲	**0.60345**▲	0.56977▲
	BCN	0.3004	0.41532▲	**0.47197**▲	0.39062▲
Shape Context	GW	0.40289	0.42236▲	0.53441▲	**0.56858**▲
	BCN	0.28821	0.36651▲	**0.42347**▲	0.35955▲

We provide in Figure 9 the precision and recall curves when using SIFT features for both datasets. Concerning the SIFT baseline, in both cases the best method is the Ide Dec-hi method which clearly performs better than the rest. In the case of the shape context baseline Ide Dec-hi performs better in the Barcelona Cathedral dataset whereas in the George Washington collection the relevance score method slightly outperforms the rest. In all the studied scenarios, between the two query reformulation strategies, Rocchio and Ide, Ide always steadily improves the results obtained by Rocchio.

In Figure 10 we show the evolution of the *mAP* measure depending on how many retrieved images the user has provided feedback. Obviously, the more images the user is asked to mark, the best the final performance is. Although in Table 3 the performance between Rocchio's method and relevance score varied depending on the dataset, we can see from Figure 10, that when asking for more relevance assessments, we have the same behavior in both datasets, where the Ide and relevance score methods outperform Rocchio's algorithm. The same exact behavior can be observed for the two different baseline systems.

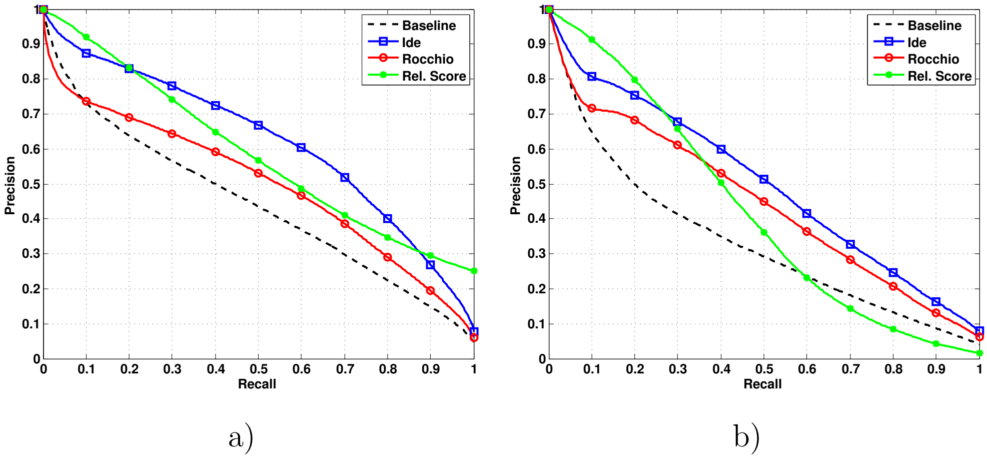


Fig. 9. Precision Recall curves for the a) GW and b) BCN collections when using SIFT features for the different relevance feedback strategies.

Table 4
Average time per query for all the query fusion and relevance feedback methods

	Baseline	Early F.	combMAX	Borda	Rocchio	Ide	RS
Average time (secs.)	0.3429	0.3559	1.0567	1.0331	0.3672	0.3677	0.0968

Of course, depending on the application, asking for a manual labeling of so much images would not be feasible and a trade-off between manual effort and system's performance has to be achieved.

6.3 Time Complexity

Finally, we report in Table 4 the average times taken for each of the methods. Regarding the query fusion methods, the early fusion strategy is as costly as the baseline, since in both scenarios just one query is casted, on the other hand, the late fusion methods are more computationally expensive since we cast three queries instead of one. Regarding the relevance feedback experiments, the reported times in Table 4 correspond to the time to compute the second result list. In that case, both Rocchio and Ide methods are like casting a new query to the system whereas the relevance score method is much more faster since it only has to re-rank the first obtained list. On the other hand, the relevance score method needs to have precomputed all the distances among words in the collection.

7 Visualization

The most common way in which image retrieval applications present the results to the user is by means of a ranked list that the user can navigate [8].

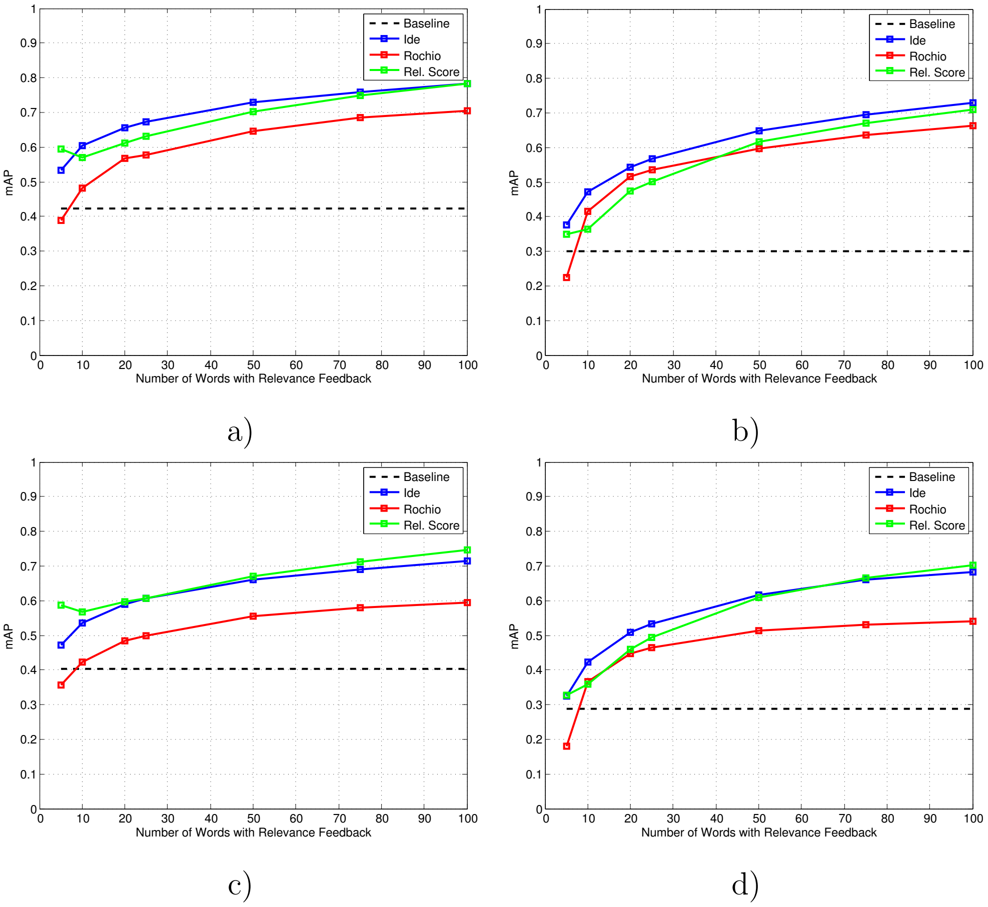


Fig. 10. Evolution of the mAP depending on the amount of words with feedback from the user. a) GW collection with SIFT features, b) BCN collection with SIFT features, c) GW collection with shape context features and d) BCN collection with shape context features.

However this might no be neither the most attractive nor informative way of presenting the retrieval results to the user. A common approach proposed by Rubner et al. in [33] is to use multidimensional scaling (MDS) in order to plot in a 2D (or even 3D) space the different similarities of the returned results instead of delivering to the user a single dimensional signal. Such visualization frameworks have been scarcely used in document image retrieval applications, and the only related work to our best knowledge is the one presented by Cloppet et al. in [7]. While a simple ranked list might be enough for the large set of plain users, specialized users such as historians or paleographers might benefit from more complex views. We propose to use a spatial view that reflects the similarities or dissimilarities observed among the retrieved words.

Given a query, the handwritten word spotting system delivers a ranked list of the topmost n similar elements in the collection. This ranked list can be browsed from more similar to less similar by the user, but the similarity notion is always with respect to the casted query. In many applications, it would be however nice to also have some indicator on the similarity among all the returned elements. In order to obtain this, we can compute the similarity matrix δ in which at position (i, j) we have stored the similarity measure $d(\mathbf{f}_i, \mathbf{f}_j)$ between the elements i and j. Now the idea is to find n points in a 2-dimensional space so that the Euclidean distance among them matches the distances of the matrix δ. Kruskal [22] then defined a closeness function to minimize that determines the positions in the 2D space where the returned images have to be located. We can see in Figures 11 and 12 an example of visualization with multidimensional scaling for the George Washington and Barcelona Cathedral collections respectively.

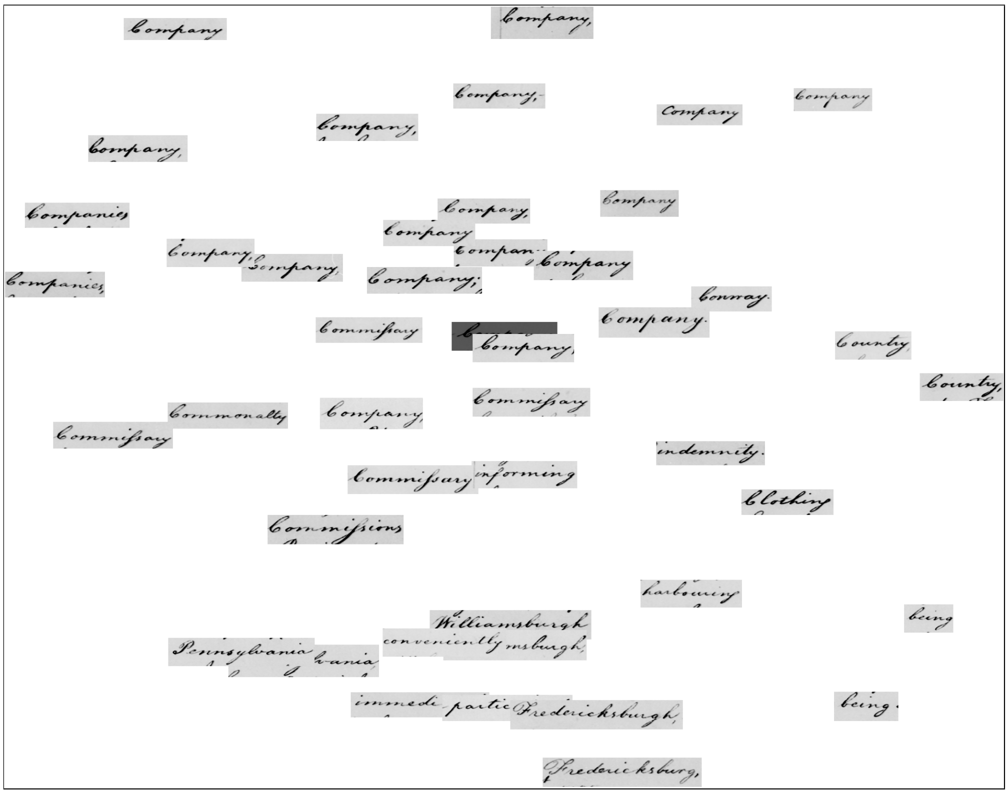


Fig. 11. Example of the multidimensional scaling visualization for the query ***Company*** (plotted in dark gray) in the GW dataset with SIFT features.

We can see that such visualization provides much more information than a simple ranked list. Not only we have information about how the returned word resembles the query, but also how similar retrieved words are among them. We usually see in those spaces how false positive words tend to be clustered together in the outer parts of this space.

Finally, we have been flirting with the idea of presenting to the user a three-dimensional model of the documents in which the words that are similar to the ones the user has queried start to "*pop out*" of the page. As a first draft of this idea we have synthetically generated a 3D anaglyph image that simulates different depth levels from the document page. We use the red-cyan anaglyph

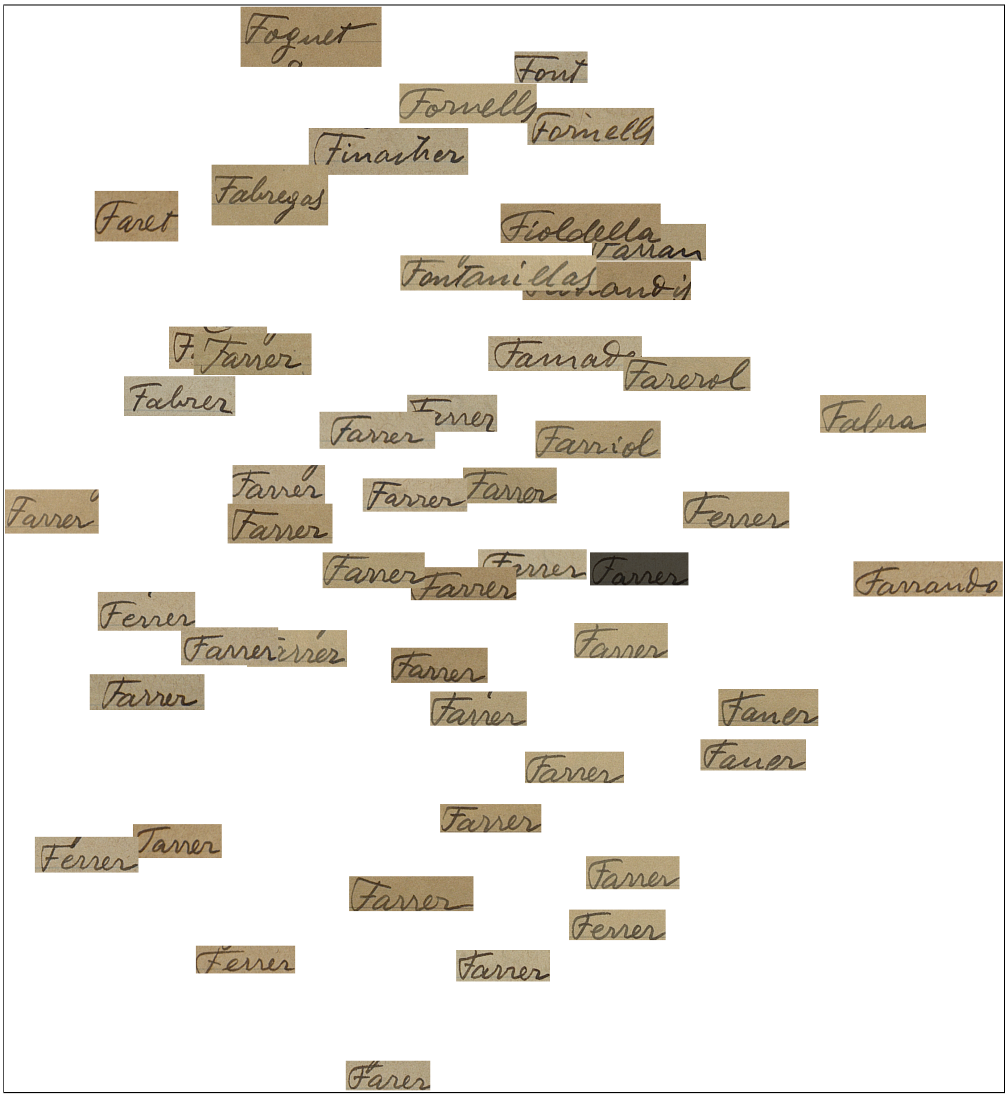
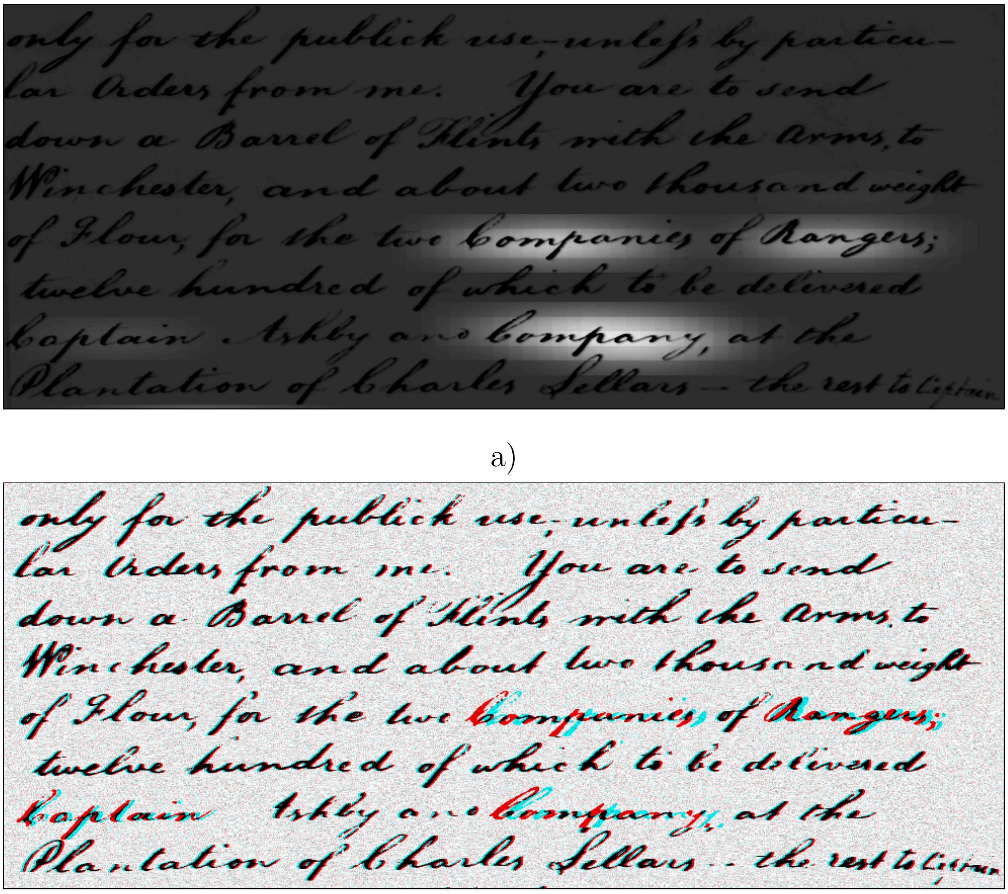


Fig. 12. Example of the multidimensional scaling visualization for the query ***Farrer*** (plotted in dark gray) in the BCN dataset with SIFT features.

image framework. A random speckle noise has been used to model a fake background. Each black pixel from the background have their red and cyan associated points at a certain displacement. Binary versions of the document words are then added to the background with a more accentuated displacement between red and cyan channels. Finally, given a query, the retrieved words that surpass a certain threshold present an emphasized displacement between its red and cyan parts so the word seem to protrude from the document image. We can see an example of this anaglyph image in Figure 13. Of course this visualization has just an aesthetic added value, whereas the use of multidimensional scaling actually delivered added information.



only for the publick use,—unleſs by particular Orders from me. You are to send down a Barrel of Flints with the Arms, to Winchester, and about two thousand weight of Flour, for the two Companies of Rangers; twelve hundred of which to be delivered Captain Ashby and Company, at the Plantation of Charles Sellars — the rest to Captain

a)

only for the publick use,—unleſs by particular Orders from me. You are to send down a Barrel of Flints with the Arms, to Winchester, and about two thousand weight of Flour, for the two Companies of Rangers; twelve hundred of which to be delivered Captain Ashby and Company, at the Plantation of Charles Sellars — the rest to Captain

b)

Fig. 13. Example visualizing word spotting results for the query ***Companies*** in the GW dataset. a) A mask on the probable locations in which the word can be found and b) synthesizing a 3D anaglyph image, the 3D effect can be appreciated when wearing red-cyan anaglyph glasses.

8 Conclusions and Discussion

In this paper we have presented a study on the inclusion of the user in the loop in a handwritten word spotting scenario. By asking the user to cast several queries instead of a single one or to provide relevance assessments on the retrieval results, we achieve significant increases of performance. Several off-the-shelf methods have been implemented and the performance increase has been demonstrated using two datasets of historical handwritten documents and two baseline word spotting approaches based on a bag-of-visual-words model have been proposed.

Considering that word spotting is a retrieval application, it should be natural that user interaction mechanisms such as relevance feedback are also taken into account when proposing new word spotting scenarios. In our particular setup, the best results were obtained by using the Ide dec-hi method when asking few relevance assessments to the user, whereas when it is feasible to ask for more manual effort from the user, the performance of the relevance score method is also competitive.

As a future research line, we would like to extend this user interaction to other word spotting methods. Here the main problem we face is that most of the tested methods are just valid when the queries are represented by a feature vector of fixed size. Many times, handwritten words are represented by features extracted from columns or sliding windows, such as in [30]. In those cases early fusion strategies are hard to apply as well as query reformulation based relevance feedback strategies as the Rocchio or Ide methods.

Two different alternative ways of visualizing the results have been proposed instead of showing the results in a ranked list. The use of multidimensional scaling has been used as a more attractive, useful and informative way of presenting the results to the user for image retrieval applications and can as well benefit handwritten word spotting applications.

Acknowledgment

This work has been partially supported by the Spanish Ministry of Education and Science under projects TIN2009-14633-C03-03 and TIN2012-37475-C02-02.

References

[1] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999.

[2] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (4) (2002) 509–522.

[3] A. Bhardwaj, D. Jose, V. Govindaraju, Script independent word spotting in multilingual documents, in: Proceedings of the International Workshop on Cross Lingual Information Access, 2008.

[4] H. Cao, V. Govindaraju, A. Bhardwaj, Unconstrained handwritten document retrieval, International Journal on Document Analysis and Recognition 14 (2) (2011) 145–157.

[5] J. Chan, C. Ziftci, D. Forsyth, Searching off-line arabic documents, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006.

[6] C. Choisy, Dynamic handwritten keyword spotting based on the NSHP-HMM, in: Proceedings of the International Conference on Document Analysis and Recognition, 2007.

[7] F. Cloppet, H. Daher, V. Églin, H. Emptoz, M. Exbrayat, G. Joutel, F. Lebourgeois, L. Martin, I. Moalla, I. Siddiqi, N. Vincent, New tools for exploring, analysing and categorising medieval scripts, Digital Medievalist 7.

[8] A. del Bimbo, Visual Information Retrieval, Morgan Kaufmann, 1999.

[9] J. Edwards, Y. Teh, D. Forsyth, R. Bock, M. Maire, G. Vesom, Making latin manuscripts searchable using gHMM's, in: Advances in Neural Information Processing Systems, 2004.

[10] D. Fernández, J. Lladós, A. Fornés, Handwritten word spotting in old manuscript images using a pseudo-structural descriptor organized in a hash structure, in: Pattern Recognition and Image Analysis, vol. 6669 of LNCS, 2011, pp. 628–635.

[11] A. Fischer, A. Keller, V. Frinken, H. Bunke, Lexicon-free handwritten word spotting using character HMMs, Pattern Recognition Letters 33 (7) (2012) 934–942.

[12] A. Fornés, V. Frinken, A. Fischer, J. Almazán, G. Jackson, H. Bunke, A keyword spotting approach using blurred shape model-based descriptors, in: Proceedings of the Workshop on Historical Document Imaging and Processing, 2011.

[13] V. Frinken, A. Fischer, H. Bunke, A novel word spotting algorithm using bidirectional long short-term memory neural networks, in: Artificial Neural Networks in Pattern Recognition, vol. 5998 of Lecture Notes on Computer Science, 2010, pp. 185–196.

[14] V. Frinken, A. Fischer, R. Manmatha, H. Bunke, A novel word spotting method based on recurrent neural networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2) (2012) 211–224.

[15] B. Fulkerson, A. Vedaldi, S. Soatto, Localizing objects with smart dictionaries, in: Computer Vision - ECCV, vol. 5302 of LNCS, 2008, pp. 179–192.

[16] G. Giacinto, F. Roli, Instance-based relevance feedback for image retrieval, in: Advances in Neural Information Processing Systems, 2004.

[17] E. Ide, New experiments in relevance feedback, in: SMART Retrieval System: Experiments in Automatic Document Processing, Prentice Hall, 1971.

[18] A. Jain, K. Nandakumar, A. Ross, Score normalization in multimodal biometric systems, Pattern Recognition 38 (12) (2005) 2270–2285.

[19] A. Kesidis, E. Galiotou, B. Gatos, I. Pratikakis, A word spotting framework for historical machine-printed documents, International Journal on Document Analysis and Recognition 14 (2) (2010) 131–144.

[20] J. Kittler, M. Hatef, R. Duin, J. Matas, On combining classifiers, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (3) (1998) 226–239.

[21] T. Konidaris, B. Gatos, K. Ntzios, I. Pratikakis, S. Theodoridis, S. Perantonis, Keyword-guided word spotting in historical printed documents using synthetic data and user feedback, International Journal on Document Analysis and Recognition 9 (2–4) (2007) 167–177.

[22] J. Kruskal, Multi-dimensional scaling by optimizing goodness-of-fit to a nonmetric hypothesis, Psychometrika 29 (1) (1964) 1–27.

[23] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: Proceedings of the Conference on Computer Vision and Pattern Recognition, 2006.

[24] J. Lladós, M. Rusiñol, A. Fornés, D. Fernández, A. Dutta, On the influence of word representations for handwritten word spotting in historical documents, International Journal of Pattern Recognition and Artificial Intelligence 26 (5) (2012) 1263002.1–1263002.25.

[25] J. Lladós, G. Sánchez, Indexing historical documents by word shape signatures, in: Proceedings of the International Conference on Document Analysis and Recognition, vol. 1, 2007.

[26] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[27] R. Manmatha, C. Han, E. Riseman, Word spotting: A new approach to indexing handwriting, in: Proceedings of the Conference on Computer Vision and Pattern Recognition, 1996.

[28] G. Mori, S. Belongie, J. Malik, Efficient shape matching using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (11) (2005) 1832–1837.

[29] G. Pirlo, D. Impedovo, Adaptive score normalization for output integration in multiclassifier systems, IEEE Signal Processing Letters 19 (12) (2012) 837–840.

[30] T. Rath, R. Manmatha, Word spotting for historical documents, International Journal on Document Analysis and Recognition 9 (2–4) (2007) 139–152.

[31] J. Rocchio, Relevance feedback in information retrieval, in: SMART Retrieval System: Experiments in Automatic Document Processing, Prentice Hall, 1971.

[32] J. Rodríguez-Serrano, F. Perronnin, Handwritten word-spotting using hidden Markov models and universal vocabularies, Pattern Recognition 42 (9) (2009) 2106–2116.

[33] Y. Rubner, L. Guibas, C. Tomasi, The earth mover's distance, multidimensional scaling, and color-based image retrieval, in: Proceedings of the ARPA Image Understanding Workshop, 1997.

[34] M. Rusiñol, D. Aldavert, R. Toledo, J. Lladós, Browsing heterogeneous document collections by a segmentation-free word spotting method, in: Proceedings of the International Conference on Document Analysis and Recognition, 2011.

[35] M. Rusiñol, J. Lladós, The role of the users in handwritten word spotting applications: Query fusion and relevance feedback, in: Proceedings of the Thirteenth International Conference On Frontiers in Handwritten Recognition, 2012.

[36] T. Syeda-Mahmood, Indexing of handwritten document images, in: Proceedings of the Workshop on Document Image Analysis, 1997.

[37] K. Terasawa, Y. Tanaka, Slit style HOG feature for document image word spotting, in: Proceedings of the International Conference on Document Analysis and Recognition, 2009.

[38] C. van Rijsbergen, Information retrieval, Butterworth-Heinemann Newton, 1979.

[39] J. Zhang, M. Marszałek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: A comprehensive study, International Journal on Computer Vision 73 (2) (2007) 213–238.