Pattern Anal Applic (2016) 19:963–976
DOI 10.1007/s10044-015-0476-0

THEORETICAL ADVANCES

A segmentation-free word spotting method for historical printed documents

Thomas Konidaris[1] · **Anastasios L. Kesidis**[2] · **Basilis Gatos**[1]

Received: 10 June 2013 / Accepted: 17 April 2015 / Published online: 16 May 2015

Abstract In this paper, a two-step segmentation-free word spotting method for historical printed documents is presented. The first step involves a minimum distance matching between a query keyword image and a document page image using keypoint correspondences. In the second step of the method, the matched keypoints on the document image serve as indicators for creating candidate image areas. The query keyword image is matched against the candidate image areas in order to properly estimate the bounding boxes of the detected word instances. The method is evaluated using two datasets of different languages and is compared against segmentation-free state-of-the-art methods. The experimental results show that the proposed method outperforms significantly the competitive approaches.

Keywords Segmentation-free · Word spotting · Historical documents

✉ Thomas Konidaris
tkonid@iit.demokritos.gr

Anastasios L. Kesidis
akesidis@teiath.gr

Basilis Gatos
bgat@iit.demokritos.gr

[1] Computational Intelligence Laboratory, Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos", Patriarchou Grigoriou St., Aghia Paraskeu, 153 10 Athens, Greece

[2] Department of Surveying Engineering, Technological Educational Institution of Athens, Ag. Spyridona, Egaleo, 122 10 Athens, Greece

1 Introduction

Worldwide libraries hold a vast amount of historical documents in terms of books, papers, drawings, journals, etc. These documents are highly valuable items due to the information they contain as well as the historical importance and rarity that characterizes them. The digitization of such archival and historical collections is an ongoing process that results in digital content which allow access to the information without distorting the original material. It is clear that efficient indexing and retrieval are important prerequisites of any system that manipulates such digital content. Optical character recognition (OCR) is a standard technology that is widely used in indexing documents with noticeable results in contemporary documents. However, historical documents are prone to a number of difficulties such as typesetting imperfections, document degradations and low print quality which decrease the performance level of OCR systems [10, 11, 16, 25].

Word spotting is an alternative methodology for document indexing based on spotting words directly on images without the use of any OCR procedures. Thus, a word spotting system attempts to detect words as a whole rather than to exactly recognize the characters as in OCR. In a typical scenario, the query image is selected from a set of predefined keywords of interest or is interactively defined by the user by cropping a rectangular image area that serves as query example. The word spotting system uses the query and detects similar words in document images based on image matching techniques without any conversion of the images into readable text. Extensive studies have shown that indexing terms in documents automatically using a word spotting system makes it possible to use costly human labor more sparingly than a full transcription would require [24]. Several word spotting

methods rely on a pre-processing step where the document image is segmented into words. The segmented words are then compared to the query image in order to detect potential matches. Recently, segmentation-free methods have also been proposed that do not require any segmentation and the document image is treated as one entity by-passing any errors that may occur due to poor segmentation results. In this line, we propose a segmentation-free word spotting method for historical printed documents. The method is based on local keypoint correspondences and consists of two distinct steps that determine candidate image areas in order to accurately extract the final bounding boxes which indicate the word instances in the document page. The method is evaluated using two different datasets of different languages and the experimental results show that the proposed method outperformed significantly the competitive approaches. The rest of the paper is organized as follows: in Sect. 2, we discuss the recent literature concerning the two main approaches followed for word spotting; the segmentation approach and the segmentation-free approach. Section 3 gives a detailed description of the proposed method. In Sect. 4 the experimental results are presented and in Sect. 5 the conclusions are drawn.

2 Related work

The word spotting literature can be divided into two main categories depending upon whether segmentation of the document image is applied or not. Indeed, there are several methods that are based on page segmentation as a preprocessing step, while others are applied directly to the document image. Additionally, a variety of features are used in order to describe the query word as well as the document image. These features strive to efficiently express the geometric and local information of the visual content and include projection profiles, Gabor features, zones and gradient-based features, to name a few. Keypoint-based local features have been also successfully used in order to describe document images as a set of local feature vectors that are invariant to scale changes, illumination and distortions. The scale-invariant feature transform (SIFT) [21] is a well-known technique in this category that produces an adequate number of distinctive features even for small visual objects. In the following, we summarize some word spotting techniques that rely at least in part on segmentation as well as approaches where no segmentation is required.

2.1 Segmentation-based methods

There are three levels of page segmentation that are typically used for detecting words in documents, namely segmentation into lines [12, 22], words [13, 14, 23, 26, 27], or even characters [2, 9]. Profile features, such as upper or lower word profiles, projection, density or transition profiles have been reported to successfully represent words in a document image that has undergone word level segmentation [24, 26]. Fusion of multiple features is also adopted in several studies in order to improve the word image description. For example, in [28] a multiple feature scheme is used consisting of projection profiles, upper/lower word profiles and background-to-ink transitions. Similarly, Jawahar et al. [7] involve word profiles to describe the outline shape of the word, structural features to extract statistical information like moments or variation and finally Fourier coefficients as a compact representation of the features in the frequency domain. In [13] a hybrid feature scheme based on a combination of projection profiles and upper/lower word profiles is used for matching words segmented from document images. In [9], a word spotting method is proposed based on mesh features and in [31, 37] the feature scheme used is gradient-based binary features. Another feature used for word spotting is based on skeletons and is used in the works of [8, 20]. Gabor features can also be applied for word spotting as proposed in [2]. In Li et al. [19], a word image is decomposed into vertical strokes and a stroke-based coding scheme is built for all the word in the document database. Considering features based on local keypoints, Ataer et al. [1] use SIFT features in order to match segmented words from Ottoman documents. Similarly, in [36] a word image matching method is presented using SIFT descriptors on keypoints that are extracted using the Fast-Corner-Detection algorithm [29]. These features are quantized into visual terms (visterms) using hierarchical K-Means algorithm and indexed using an inverted file. In [32], a word spotting method based on line segmentation is presented. The method uses a sliding window over each line. The matching is performed using dynamic programming and slit style HOG features. In the previous methods, the segmented words are presented as feature vectors and dynamic time warping is an algorithm that has been extensively used to match words based on these vectors [7, 12, 14, 27, 28]. Other matching techniques are based on morphological variants [23], voting schemes [1, 20, 36], similarity distances [8, 13], character or string matching [9, 19] and correlation measures [22, 31, 37]. Overall, document segmentation results to higher level structures that are semantically important and can be further explored. On the other hand, detection methods based on segmentation results are intrinsically prone to errors like over- or under-segmentation as well as well as partial occlusion and mis-segmentation.

2.2 Segmentation-free approaches

Although, there is a very large collection of published work concerning the segmentation approach, in the recent years there is a growing research interest concerning segmentation-free methods. There are cases where documents cannot be segmented correctly leading to insufficient results. The segmentation-free approaches overcome the problems associated to bad segmentation results by treating the document image as a whole. In [5], a template-matching method based on pixel densities is used for locating words in documents without segmenting them. Although the method provides rotation and scale invariance, this is applied to limited extent. In [18], an alphabet is used that is manually selected from each document collection processed. The alphabet is used to create word instances that serve as queries. The features extracted are based on gradient values. In [17], gradient information is also used as features for the word images. Word interest points are matched against document images and try to locate zones of interest presenting similar features. Local image features have been also used in segmentation-free methods trying to benefit from the scale and rotation invariance they offer as well as their robustness to noise. Such methods usually involve a voting scheme in order to detect and localize potential word matches in the document image. In this line, Rusinol et al. [30] opt SIFT features in a bag-of-visual-words approach. The method is applied on both handwritten and printed documents. The SIFT features are extracted using small predefined squared areas that are assumed to cover most of the font sizes. The search space does not correspond to the entire document image but rather to overlapping local patches of fixed geometry. However, several assumptions concerning the size of the patch and the expected font sizes seriously affect the generalization and the applicability of the method. Furthermore, as the authors mention, the performance of the system is highly related to the length of the queried words.

3 Proposed method

In the proposed method, we adopt a segmentation-free word spotting approach in order to overcome the poor segmentation results that usually characterize historical documents. We are based on SIFT features that have been proved to provide robustness concerning low image quality and image degradation. Furthermore, SIFT features are scale and rotation invariant. Unlike [5], we do not need to produce extra information concerning the scales and rotations of the images we process, since this information is embedded into the SIFT keypoints.

However, detection of word instances based solely on direct matching between query and image SIFT keypoints leads to unsatisfactory results. This is due to the spatial scattering of matching correspondences since a query keypoint may be similar to a large number of document page keypoints. These document keypoints do not a priori belong to correct word instances. Furthermore, the existence of multiple word instances in the same document page does not allow the query image to be matched by a sufficient number of correspondences. Such a situation is demonstrated in Fig. 1.

The proposed method does not adopt the original matching process described in the SIFT algorithm, but instead a two-step approach is followed. In the first step,

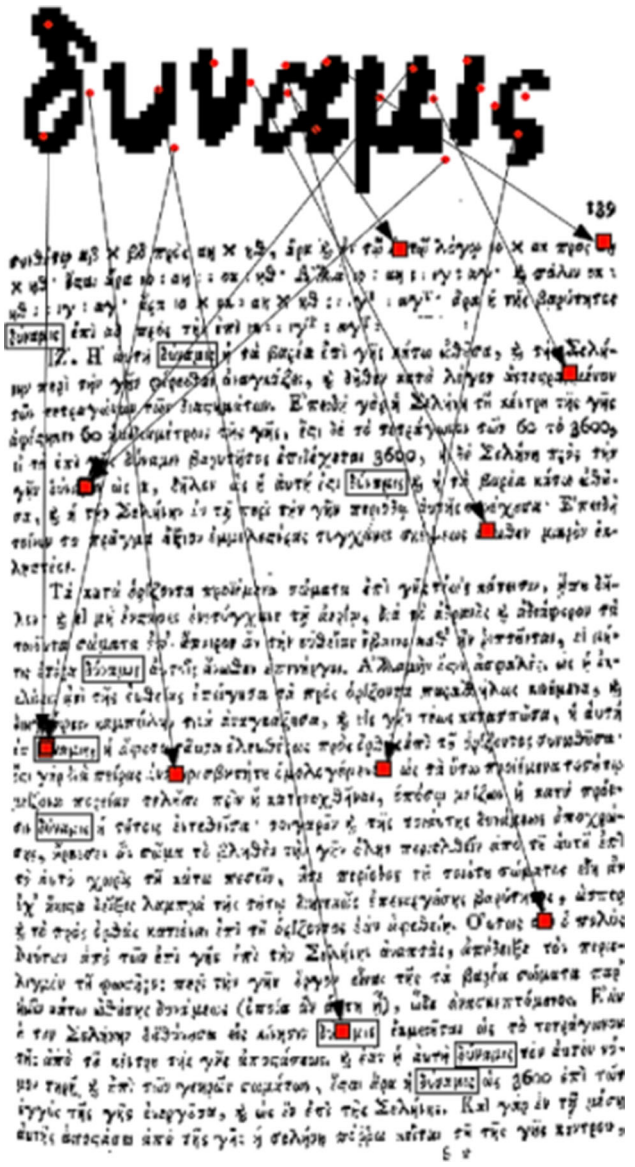


Fig. 1 Keypoint correspondences between the query keyword image and a document page image

for every keypoint in the query keyword image, the nearest K points are found in the entire document page image, without using any presegmentation information in order to narrow down our search space as in the work proposed in [30]. These document keypoints are used as indicators in order to create candidate image areas. In the second step, each candidate image area is matched against the query keyword image. The keypoint correspondences are used by the RANSAC algorithm in order to estimate the final bounding boxes indicating the detected word instances. Furthermore, we use the strength of SIFT descriptors in a way that multiple instances of the desired word can be found on the document page. Figure 2 shows the various steps of the proposed method.

3.1 Detection of candidate image areas

The first step of the proposed method involves the matching of the query keyword keypoints to the document keypoints. The purpose is to find point correspondences on the document image that will serve as indicators of candidate image areas. For each keypoint of the query keyword, we locate the K most similar keypoints on the document image. The value of K is experimentally defined as discussed in Sect. 4. Let f_{query} and f_{doc} be the SIFT feature vectors of the ith keypoint in the query keyword image and the jth keypoint in the document image, respectively. The distance between these two keypoints is calculated as follows:

$$d(i,j) = \cos^{-1}(\langle f^i_{\text{query}}, f^j_{\text{doc}} \rangle), \tag{1}$$

where $\langle f_{\text{query}}, f_{\text{doc}} \rangle$ denotes the dot product between the two normalized vectors.

Each pair of corresponding keypoints defines a candidate image area on the document page. Since we know the relative position of the query keyword keypoint in respect to the edges of the query keyword image, we define a bounding box around the keypoint on the document image taking into account the position of the corresponding keypoint of the query keyword image. Let $p_{\text{query}}(x_{\text{q}}, y_{\text{q}})$ be a point on the query keyword image and $p_{\text{doc}}(x_{\text{d}}, y_{\text{d}})$ be its corresponding point on the document page image. Let dx, dy be the distance of the query keypoint from the left and the top edge of the query keyword image, respectively. The bounding box surrounding the candidate image area is defined by its top-left $(x_{\min}, y_{\min})$ and bottom-right $(x_{\max}, y_{\max})$ corners is given by the following equations:

$$x_{\min} = x_{\text{d}} - \left(\frac{\text{sc}_{\text{doc}}}{\text{sc}_{\text{query}}} \cdot \text{d}x \cdot t_{\text{s}} \right) \tag{2}$$

$$y_{\min} = y_{\text{d}} - \left(\frac{\text{sc}_{\text{doc}}}{\text{sc}_{\text{query}}} \cdot \text{d}y \cdot t_{\text{s}} \right) \tag{3}$$

$$x_{\max} = x_{\text{d}} + \left[(w_{\text{q}} - x_{\text{q}}) \cdot \frac{\text{sc}_{\text{doc}}}{\text{sc}_{\text{query}}} \cdot t_{\text{s}} \right] \tag{4}$$

$$y_{\max} = y_{\text{d}} + \left[(h_q - y_q) \cdot \frac{\text{sc}_{\text{doc}}}{\text{sc}_{\text{query}}} \cdot t_{\text{s}} \right], \tag{5}$$

where w_{q} and h_{q} are the width and height of the query keyword image, respectively. Parameter t_{s} is the boundary extension factor, which gives extra space to the boundaries of the candidate image areas and has been experimentally set to 1.1. The value of the extension factor accommodates all the processed documents presented in this paper and it plays a rather auxiliary role in this part of the method. Variables sc_{query} and sc_{doc} are the scales of the query keypoints p_{query} and p_{doc}, respectively, as provided by the SIFT algorithm. An example showing candidate image areas is shown in Fig. 3.

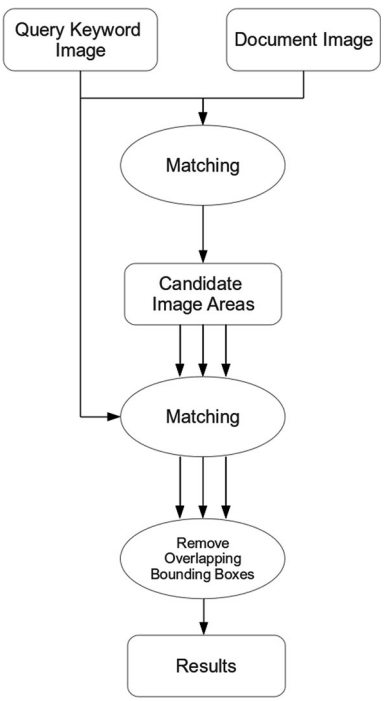


Fig. 2 The architecture of the proposed method

3.2 Detection of word instances

In the previous section, we matched the query keyword image with the entire document page image in order to use the matching keypoints as indicators for creating candidate image areas. These areas cannot guarantee that they contain the query word under consideration. For this reason, the keypoints of the query keyword image are matched

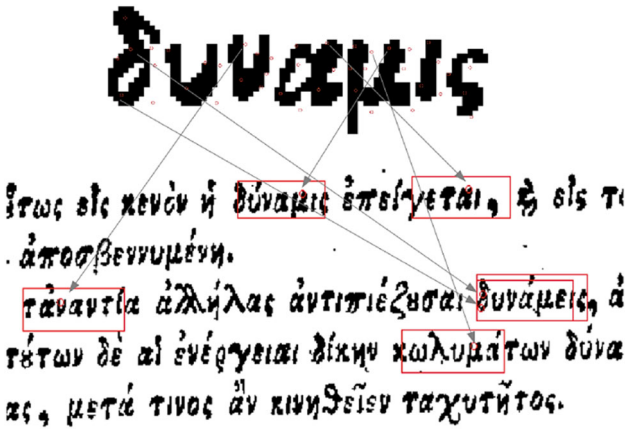


Fig. 3 Candidate image areas that were created from the corresponding keypoints of the document page image

against the keypoints of each candidate image area. For each keypoint on the query keyword image we find the most similar keypoint on the candidate image area using Eq. 1. In order to estimate a model that describes the efficiency of these keypoint correspondences the RANSAC algorithm [4] is involved. RANSAC is an iterative method that can efficiently estimate the parameters of a model even when the measurements contain outliers. Using RANSAC the number of inliers is calculated, that is, the number of corresponding pairs that are conveniently described by the model. Moreover, the keypoint correspondences are used in order to calculated a homography that serves as a transformation matrix from the query keyword image to the candidate image area plane [6]. There must be at least four-point correspondences to calculate the homography matrix. Let $p_{\text{query}}(x_q, y_q)$ be a point in the query keyword image and $p_{\text{area}}(x_c, y_c)$ be the corresponding point in the candidate image area. The transformation between these two points can be given by the following equation:

$$p_{\text{area}} = H \cdot p_{\text{query}}, \tag{6}$$

where H is the homography matrix. The above equation can take the form:

$$\begin{bmatrix} x_q \\ y_q \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix}. \tag{7}$$

An example is shown in Fig. 4. The bounding box of the query image is transformed into the document image resulting to the gray-shaded area.

This process is applied to all candidate image areas aiming to produce a set of bounding boxes that are afterwards ranked according to their matching efficiency. The inliers percent provides an indicator of the goodness of fit regarding the RANSAC model. However, there may be a

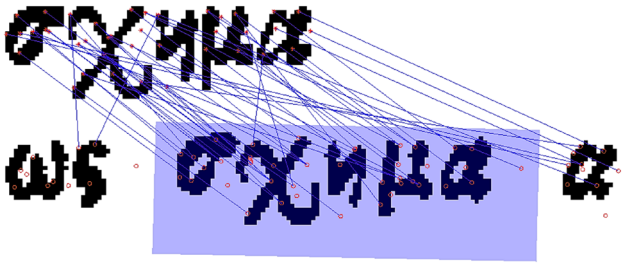
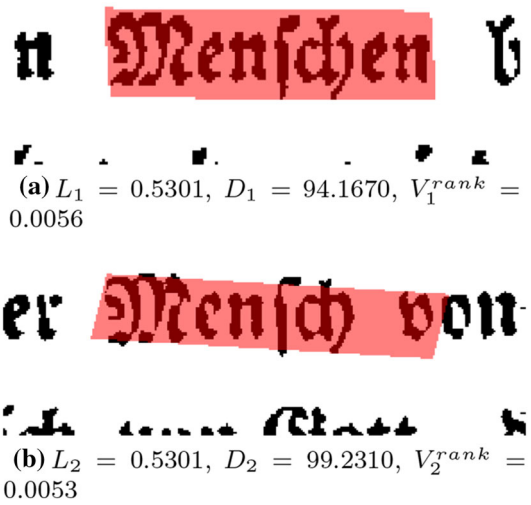


Fig. 4 The application of the homography matrix H that is calculated based on the point correspondences between the query keyword image and the candidate image area



(a) $L_1 = 0.5301$, $D_1 = 94.1670$, $V_1^{rank} = 0.0056$

(b) $L_2 = 0.5301$, $D_2 = 99.2310$, $V_2^{rank} = 0.0053$

Fig. 5 Ranking values for two resulting bounding boxes concerning the query keyword "Menschen", that share the same inliers percent value. **a** $L_1 = 0.5301$, $D_1 = 94.1670$, $V_1^{\text{rank}} = 0.0056$, **b** $L_2 = 0.5301$, $D_2 = 99.2310$, $V_2^{\text{rank}} = 0.0053$

number of detected bounding boxes having equal inliers percent value. In order to distinguish them, we propose to divide the inliers percentage value L by a quantity D which corresponds to the sum of the distances between the query and the candidate image area keypoints. Thus, for a candidate image area the ranking value V^{rank} is calculated as follows:

$$V^{\text{rank}} = \frac{L}{D}. \tag{8}$$

In Fig. 5 we illustrate an example of two detected bounding boxes that share the same inliers percent value. However, the bounding box in Fig. 5a has smaller total distance, thus is ranked higher than the bounding box in Fig. 5b.

3.3 Removing overlapping results

We have seen that the candidate image areas are created using the point correspondences between the query

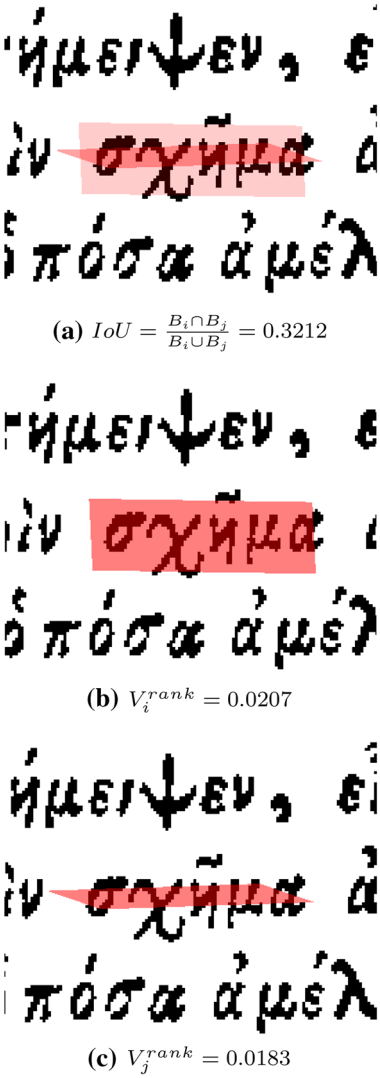
keyword image keypoints and the keypoints of the document page image. There are cases where more than one candidate image areas correspond to the same word in the document image. Therefore, we end up with overlapping bounding boxes, each of them having different ranking values V^{rank} as calculated by Eq. 8. On the document image, two bounding boxes B_i and B_j are considered overlapping if the intersection over union (IoU) measure exceeds some predefined threshold, that is,

$$\text{IoU} = \frac{B_i \cap B_j}{B_i \cup B_j} \geq t_v, \tag{9}$$

where t_v has been experimentally defined equal to 0.3. This parameter prevents duplicate entries in the list of bounding boxes, that in most cases would lead to unreliable results during the evaluation of the method. The bounding box that has the larger ranking value V^{rank} among the overlapping bounding boxes is the one kept while the others are discarded from the list. Figure 6 illustrates an example of resulting bounding boxes concerning the same candidate image area. The two bounding boxes are considered overlapping since their intersection over union ration exceeds the threshold t_v, as shown in Fig. 6a. However, the bounding box in Fig. 6c has a smaller ranking value V^{rank} than the bounding box in Fig. 6b and it is discarded from the list of bounding boxes. The remaining bounding boxes are further filtered out using the following criterion that compares the aspect ratio of the bounding box A_{bb} to the aspect ratio of the query image A_{q} based on a threshold t_{a} that has been experimentally set to 0.4. That is,

$$\left| \frac{A_{\text{bb}} - A_{\text{q}}}{A_{\text{q}}} \right| \geq t_{\text{a}} \tag{10}$$

The main purpose of this parameter is to eliminate irregularly shaped bounding boxes.



(a) $IoU = \frac{B_i \cap B_j}{B_i \cup B_j} = 0.3212$

(b) $V_i^{rank} = 0.0207$

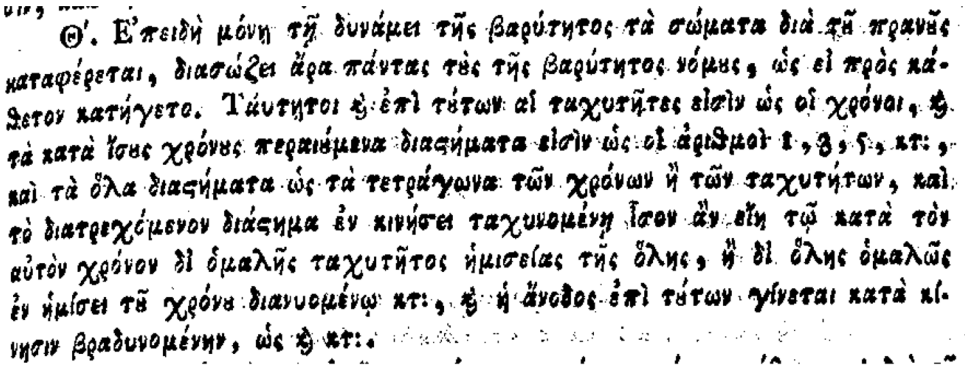
(c) $V_j^{rank} = 0.0183$

Fig. 6 Resulting bounding boxes B_i and B_j for the same word on a document page image. **a** Their Intersection over Union ratio, **b** the bounding box B_i with ranking value of 0.0207, **c** the bounding box B_j with ranking value of 0.0183. The bounding box B_j is discarded since it has lower ranking value V^{rank}

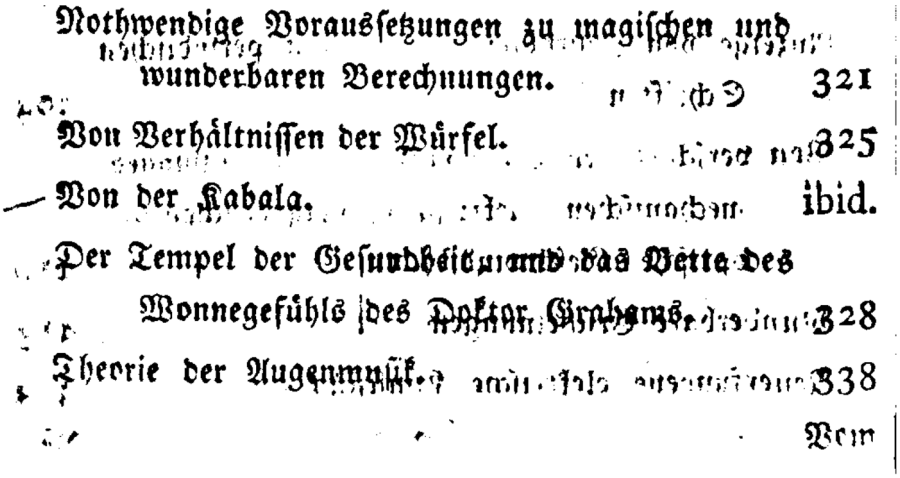
4 Experimental results

The experiments that were conducted in order to evaluate the proposed method include two different datasets. The first dataset consists of 100 pages from a Greek historical machine-printed book of the Renaissance period. The second dataset consists of 100 pages which are part of a German historical machine-printed book of Eckartshausen which was published in 1788 and is owned by the Bavarian State Library [34]. Figure 7 illustrates samples from both datasets. It can be seen that the datasets are characterized by degradation problems that are typical in historical documents, namely, faded characters, bleed-through as well as curved text lines, ink fades, etc. Due to these factors, segmentation of the documents into individual words may lead to insufficient results as shown in Fig. 8. Moreover, the two datasets have been intentionally selected to have different characteristics in order to demonstrate the applicability of the proposed method in a variety of historical machine-printed documents without adjusting its threshold parameters.

Ground truthing can be a very tedious process when a manual approach is followed. Nevertheless, there are methods that can automate the process resulting into a faster ground truth generation [15, 35]. In the proposed method, the ground truth consists of spotted words in the dataset pages that are exact or partial matches of the query keyword, that is, the query keyword may appear in



(a) A sample of the Greek historical book.



(b) A sample of the German historical book.

Fig. 7 Sample pages from both datasets. **a** A sample of the Greek historical book, **b** A sample of the German historical book

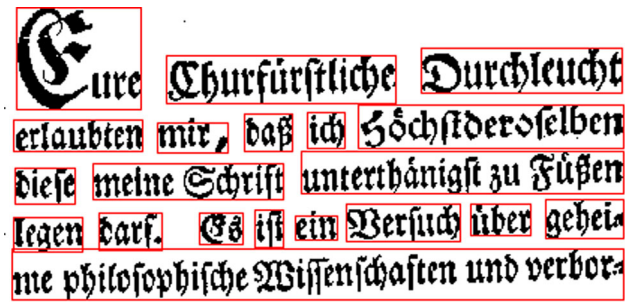


Fig. 8 Badly segmented document image

the dataset either as an individual word or as part of a larger word. For instance, in the Greek dataset, the query "σώμα" may appear as an individual word but also as part of a larger word, e.g., "σώματα" It should be noticed that finding if a query is part of a larger word does not involve any morphological rules or analysis into stems and affixes. Instead, the comparison is on a textual basis that simply tests if the query string is substring of a particular ground truth word. This is also followed in the experiments concerning the German documents. For example, if the query keyword is "auch" then the ground truth includes both individual instances of this word as well as larger words like "brauchen" or "Rauchwerke". Adopting this approach allows the evaluation of the method's performance over all possible instances of the given query. Indeed, since detection and matching is performed on an image basis, it is clear that when the query image matches some particular image area in a document page then this area may correspond either to the query keyword in exact or to a certain part of a larger word.

Evaluation is performed by assigning the detected words to the ground truth instances and measuring how well the corresponding areas overlap in order to judge if they are true or false positives. For this purpose, the IoU measure is used as defined in Eq. 9 where, in this case B_i and B_j denote the bounding boxes of the detected instance and the ground truth, respectively. Thus, in case of a perfect match, the IoU score equals 1. We adopted an IoU threshold of 0.5 which is often used for evaluation purposes in detection methods [3]. Furthermore, we have also experimented with a more relaxed IoU threshold value of 0.3.

Performance is calculated in terms of normalized 11-point precision-vs-recall curves. Precision is defined as the number of relevant results in the results list over the total number of results while recall is defined as the number of relevant results in the results list over the total number of relevant results. The following sections demonstrate the proposed method's applicability in both of these datasets and provide comparative results with other state-of-the-art approaches.

4.1 The Greek dataset

For the evaluation of the Greek dataset a set of seven query keywords is used. Table 1 depicts the query images as well as their frequency of appearance throughout the 100 Greek pages. Even though the documents are printed there is a high visual variability across the instances of each keyword in the document corpus. For example, in Table 2 a few instances of keyword "δυναμις" are shown that appear at different slopes, with variable inter-character spacing, light or heavily inked, etc.

Figure 9 depicts normalized precision–recall curves for all the set of 7 Greek query keywords. For each value of $K = 1 \ldots 5$ a separate precision–recall curve is provided (see Sect. 3). Clearly, for $K = 1$ provides the poorest results while for any value of K higher than one an increase in performance is reported. This implies that, for a given query keypoint, the candidate image area indicated by its most similar page keypoint does not always correspond to an actual word instance. On the other hand, there is no significant difference in the performance achieved for values of $K \geq 3$.

Table 1 The seven queries for the Greek dataset experimentsand the overall number of ground truth instances

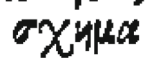
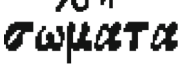
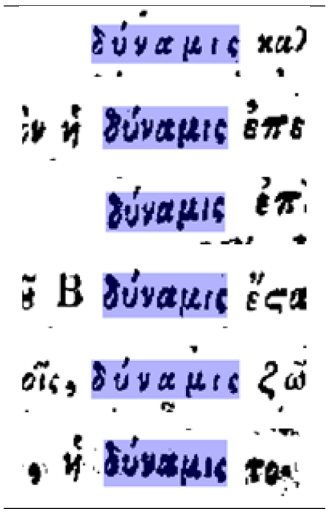
Query Keyword	Number of ground truth instances
	27
	97
	75
	32
	78
	64
	51

Table 2 Various instances of keyword "δυναμις"



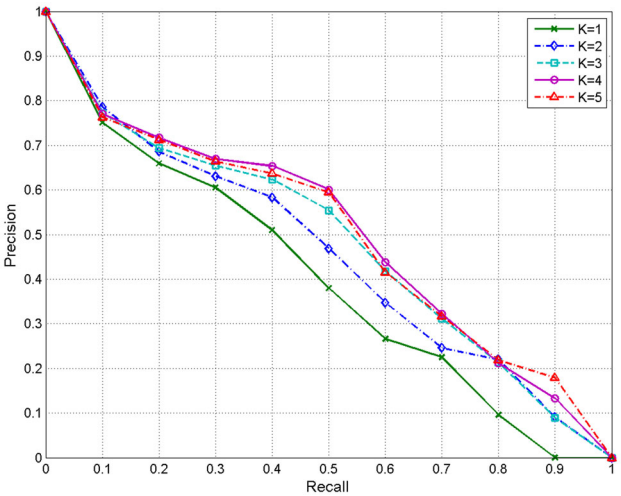
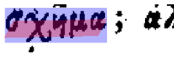
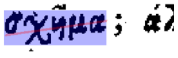
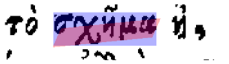
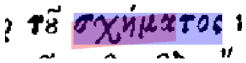
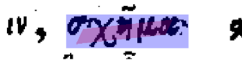
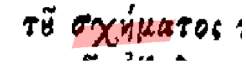
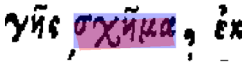
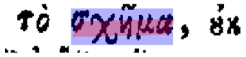
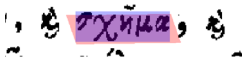
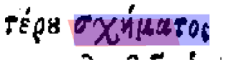


Fig. 9 Mean precision-vs-recall results over the 7 Greek keywords for $K = 1 \ldots 5$

Table 3 demonstrates an example regarding keyword "σχημα". Specifically, the top 10 results are shown. The cropped images depict the area around the detected word. The ground truth and the detected word are shown as blue and orange areas, respectively. It can be seen that the 4th, 6th and 10th instances correspond to longer words that include the query string. The rest of the instances correspond to individual words. This example demonstrates all the possible cases that may appear during the evaluation

Table 3 Top 10 results for query "σχημα"

#	Instance	Result
1.		✗
2.		✗
3.		✓
4.		✓
5.		✗
6.		✗
7.		✓
8.		✗
9.		✓
10.		✓

process. For example, the instances 1, 2, 5 and 8 are failures since the corresponding IoU values are below the matching threshold of 0.5. Instance 6 is considered a mismatch as well; however, this is due to a fault in the ground truth that erroneously excluded this instance from the list of correct instance. On the other hand, instances 3, 4, 7, 9 and 10 are correctly detected either as a individual instances or as part of a larger word.

In Fig. 10, the proposed method is compared against the method of Gatos and Pratikakis [5] as well as the original SIFT approach. The method in [5] performs a block-based template matching while the original SIFT approach detects word instances on the document pages through keypoint matches that satisfy the cosine similarity criterion as described in [21]. For each keypoints pair the bounding box on the document page is determined by Eqs. (2)–(5) as described in Sect. 3.

4.2 The german dataset

The next set of experiments considers the German dataset. A set of 100 keywords were picked out from 100 pages of this dataset. The full list of keywords is shown in Table 4. For evaluation purposes, the ground truth for these keywords has been manually defined by marking all actual instances across the entire dataset. As shown in Fig. 11,

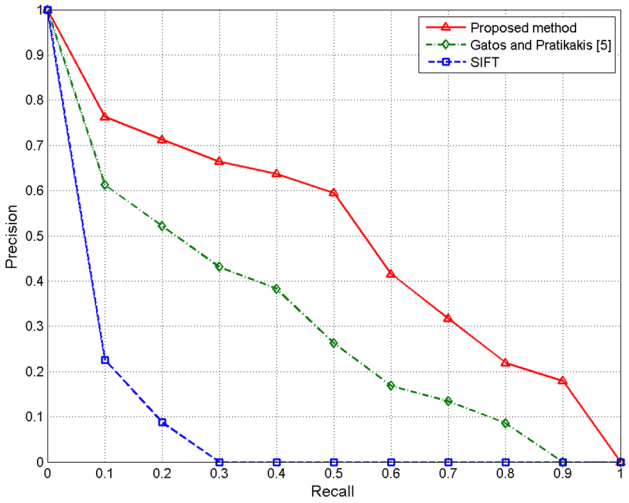


Fig. 10 Precision-vs-recall curves comparing the proposed method to Gatos and Pratikakis [5] and the original SIFT approach

most of the keywords have less than 40 ground truth instances in the document corpus. However, some keywords appear much more often, e.g., "nicht" with 167 instances.

Keywords can also be characterized by their length in terms of character count. In the set of 100 keywords, the length varies significantly starting from 3 characters ("Art", "Man") up to 16 characters long ("Einbildungskraft"). However, the majority of keywords have a length between 4 and 8 characters, as shown in Fig. 12.

Figure 13 depicts normalized 11-point precision–recall curves for all the set of 100 query keywords. A separate precision–recall curve is given for value of $K = 1 \ldots 5$. It can be seen that the performance is improved for all K greater than one. On the other hand, the performance for K values between 2 and 5 is almost identical. Thus, similar performance level can be achieved with $K = 2$ or 3 which undoubtedly requires less computational effort than $K = 5$.

In order to examine how the keyword length affects the applicability of the proposed method, the keywords are grouped according to their length and normalized precision-vs-recall curves are calculated for each group. In accordance with Fig. 12 there are 12 such groups formed. The results for all groups are shown in Fig. 14. It can be seen that the method has a stable performance regardless of the keyword's length, except for very short keywords, i.e., less than 5 letters long.

Table 5 depicts the top 10 matches for query keyword "auch". In this example, all instances except the 6th and the 8th are individual words in the dataset. The 8th instance is considered a true match even though the detected area corresponds to a subpart of a larger word. The instance is considered as positive since the IoU score is above the

Table 4 The German keywords

aller, andern, Art, auch, Auge, Begriffe, bekannt, Bewegung, Bild, denken, deutlich, Dinge, durch, Durchleucht, Einbildung, Einbildungskraft, Empfindungen, Erde, erklären, Erklärung, Erscheinung, Experiment, Fibern, Folgen, ganz, gegen, Geheimnisse, Geheimniß, Geist, Geisterwelt, Gottheit, große, haben, Hand, Hohlspiegel, immer, kann, kein, Kenntnisse, Kette, Kraft, können, Körper, Körperwelt, lassen, Leben, Licht, Liebe, liegt, machen, Man, meine, meiner, Mensch, Menschen, Natur, nicht, noch, ohne, Organ, Person, Phantasie, Sache, Sachen, sagte, schon, Seele, sehen, sehr, selbst, seyn, Sinne, sondern, Spiegel, Sprache, Stolz, Theorie, über, unsere, unter, Ursache, verschiedene, Verstand, Versuche, Vollkommenheit, Wahrheit, Wasser, welche, wenig, werden, Wesen, wieder, wirklich, Wirklichkeit, Wirkung, Wirkungen, wollen, Worte, Zeit, Zimmer

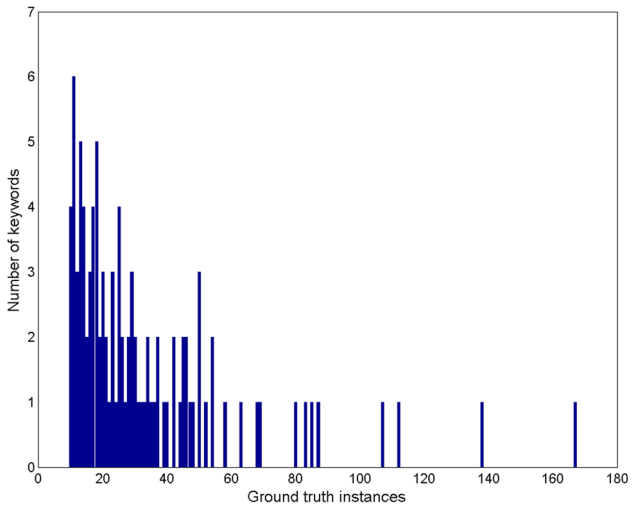


Fig. 11 Distribution of keywords according to their appearance frequency

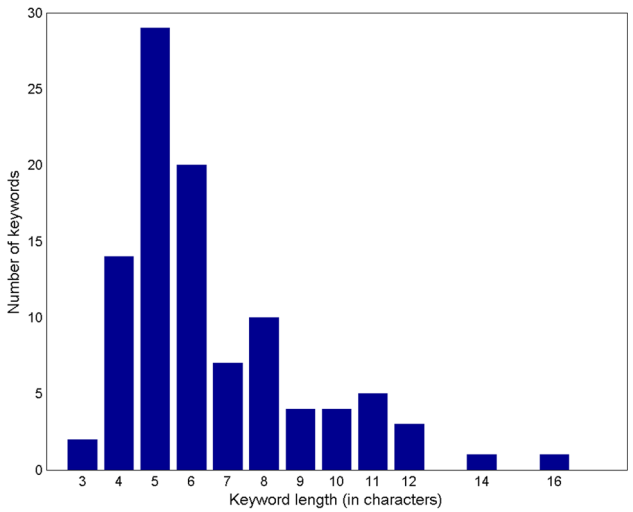


Fig. 12 Distribution of keywords according to their length

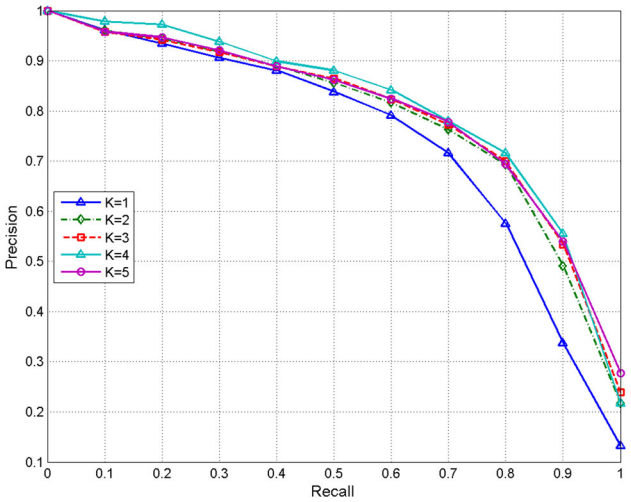


Fig. 13 Mean precision-vs-recall results over all 100 keywords for $K = 1 \ldots 5$

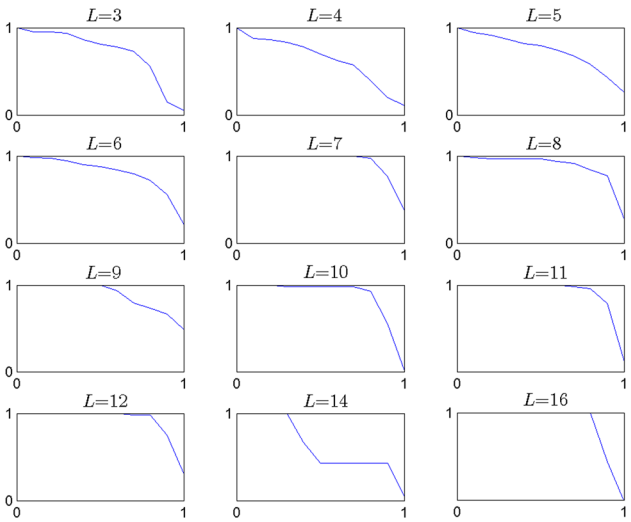
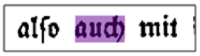
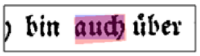
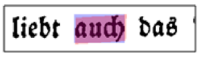
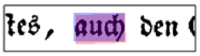
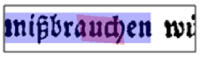
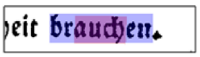
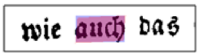
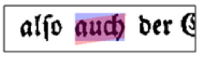


Fig. 14 Precision-vs-recall curves according to keyword length L, for $K = 5$

threshold 0.5. On the other hand, the 6th instance is considered a failure since it does not fulfill the IoU threshold.

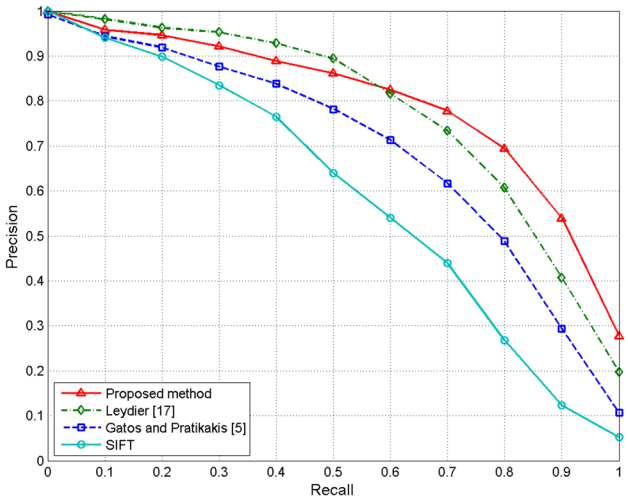
In the German dataset besides the method described in [5] and the original SIFT, we also compare the proposed method with the state-of-the-art method presented in [17]. Figure 15 illustrates 11-point precision-vs-recall curves for all the aforementioned methods. Specifically, the results are given for IoU evaluation threshold equal to 0.5 and 0.3,

Table 5 Top 10 results for query "auch"

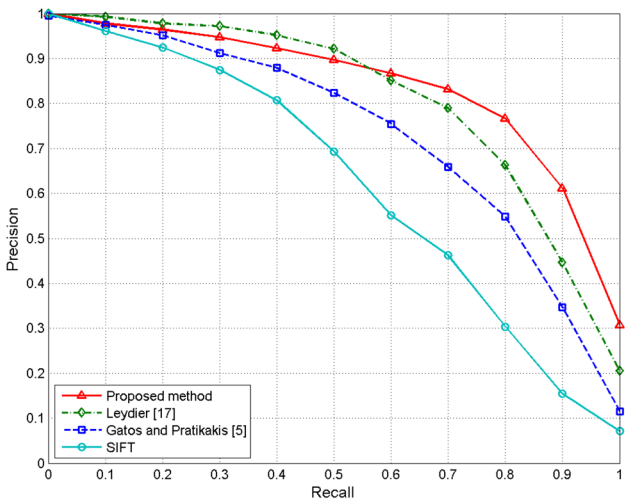
#	Instance	Result
1.		✓
2.		✓
3.		✓
4.		✓
5.		✓
6.		✗
7.		✓
8.		✓
9.		✓
10.		✓

respectively. Tables 6 and 7 summarize the performance of the competitive methods in terms of Mean Average Precision, Geometric Mean Average Precision, R-Precision, bpref and Reciprocal Rank as provided by the well-known trec evaluation tool [33]. It can be seen that the proposed method outperforms the competition in the vast majority of the overall statistics for both cases of the IoU evaluation threshold. Moreover, Fig. 16 demonstrates detailed Mean Average Precision results per keyword length. Again, the proposed method shows better results especially for keyword lengths between 5 and 11 characters where the majority of the query keyword instances belong.

In terms of computational time, the proposed methods takes on average 6 s to process an entire document page (Intel i7, 16GB RAM). The proposed method was implemented in Python using the NumPy and SciPy packages. We need to stress here that the time taken to process a page depends on the number of query keypoints. Moreover, the computational time can be further improved if approaches like feature quantization described in [36] are adopted.



(a)



(b)

Fig. 15 Precision-vs-recall curves comparing the proposed method to Leydier et. al.[17], Gatos and Pratikakis [5] and the original SIFT approach **a** for IoU threshold 0.5 and **b** for IoU threshold 0.3

Table 6 Evaluation metrics regarding the competitive methods for IoU = 0.5

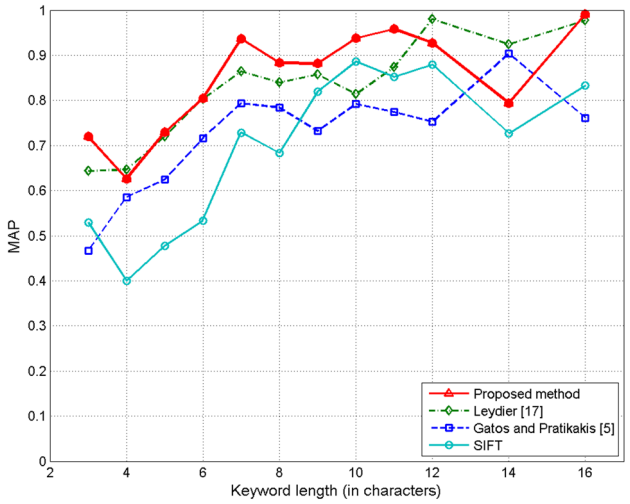
	Proposed	Leydier et al. [17]	Gatos et al. [5]	SIFT
MAP	**0.795**	0.776	0.689	0.584
Geometric MAP	**0.751**	0.747	0.640	0.503
R precision	0.771	**0.774**	0.675	0.604
bpref	**0.927**	0.921	0.871	0.625
Reciprocal rank	**1.000**	**1.000**	0.985	**1.000**

Results in bold indicate best performance

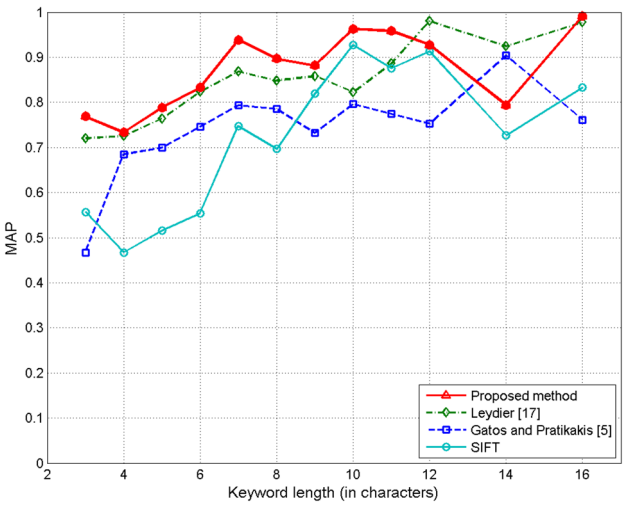
Table 7 Evaluation metrics regarding the competitive methods for IoU = 0.3

	Proposed	Leydier et al. [17]	Gatos et al. [5]	SIFT
MAP	**0.836**	0.808	0.730	0.615
Geometric MAP	**0.809**	0.787	0.697	0.545
R precision	**0.796**	0.792	0.700	0.626
bpref	0.967	**0.990**	0.907	0.651
Reciprocal rank	**1.000**	**1.000**	0.995	**1.000**

Results in bold indicate best performance



(a)



(b)

Fig. 16 Mean average precision per keyword length for **a** for IoU threshold 0.5 and **b** for IoU threshold 0.3

4.2.1 Experiments with non-frequent words

In this section, the performance of the proposed method is examined in case of infrequent query words that have a few instances throughout the dataset. Specifically, three categories of queries are tested, namely, queries with 3, 5 and 8 instances. Each category consists of 15 words. In order to check the influence of the word length in the retrieval performance, each category is further divided into 3 subsets of 5 queries each with word length 5, 7 and 10 (or more) characters, respectively. Table 8 summarizes the query words in the nine subsets of experiments.

Table 8 Query words in the 9 sets of experiments

Length (in characters)		
5	7	≥10
Instances		
3		
Brief	Anblick	Einswerdung
Ferne	Bildung	Mittelpunkt
Lappe	Laterne	Philosophie
Punkt	Schrift	Temperament
Reize	Zukunft	Vorstellung
5		
Fokus	Absicht	Eigenschaft
Linie	Irrthum	Gegenstand
Magie	schwarz	Leidenschaft
Nacht	Laterna	Vorbereitung
Weise	Zeichen	Wissenschaft
8		
Augen	Freunde	Beschaffenheit
Folge	Kapitel	Erkenntniß
Grund	Zustand	Kohlpfanne
Macht	besteht	Leidenschaften
Mitte	weniger	Wissenschaften

Figure 17 depicts average precision (AP) for the words grouped by the number of instances. All three categories demonstrate similar retrieval performance. For example, Figure 17a shows that there are queries with a perfect AP score in spite of the query length (e.g., 'Brief', 'Ferne', 'Lappe', 'Bildung', 'Laterne', 'Einswerdung', 'Mittelpunkt' and 'Philosophie'). The rest of the query words in Figure 17a demonstrate a weaker retrieval performance caused by irrelevant words (i.e., not in the ground truth of

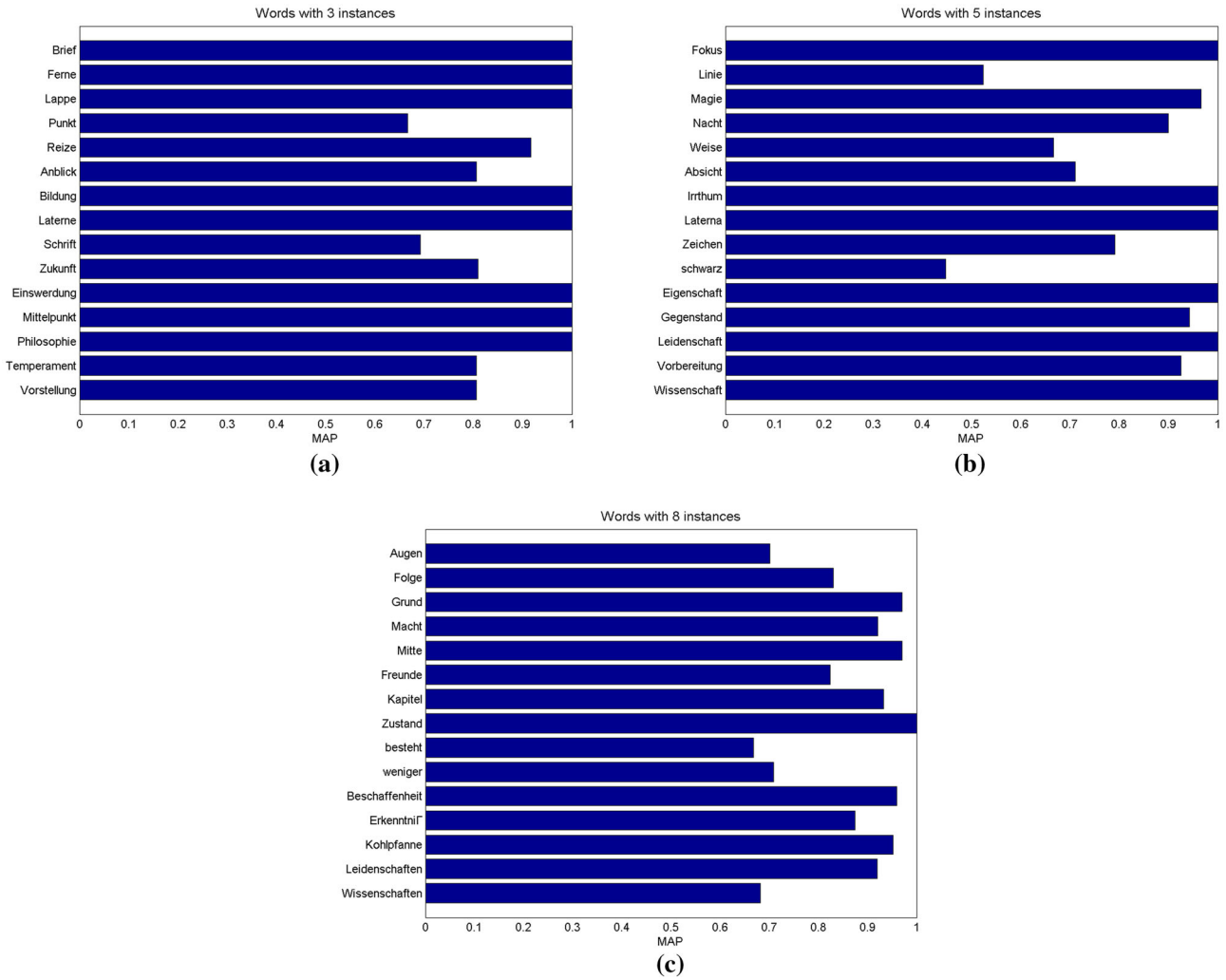


Fig. 17 Average precision for queries with **a** 3 instances, **b** 5 instances, and **c** 8 instances

Table 9 Summary MAP results for the 9 sets of experiments

	Length (in characters)			
	5	7	≥10	
Instances				
3	0.92	0.86	0.92	0.90
5	0.81	0.79	0.97	0.86
8	0.88	0.83	0.88	0.86
	0.87	0.83	0.92	0.87

the query) that appear in the top positions of the resulting ranked list. Again, such kind of errors are irrelevant of the query length. Similar results are obtained in the other two categories shown in Fig. 17b, c.

Table 9 provides summary MAP values for each subset of experiments. Additionally, marginal MAP values per instance and per query length are provided as well as the overall MAP score. For example, the overall MAP for the 15 query words with length 3 is 0.90. It can be seen that despite some slight performance differences the retrieval efficiency is in general unrelated to the number of instances and the query length.

5 Conclusions

In this paper, we present a novel approach for segmentation-free word spotting. The method uses the strength of SIFT keypoint descriptors twofolds, initially in order to create candidate image areas on the entire document page. Then, the candidate image areas are matched against the query keyword image in order to create the final bounding boxes that correspond to word instances on the document page. In case of overlapping word instances the best result is chosen and the list of detected words is further pruned using the aspect ratio of the word's dimensions. The proposed method performs well on two different datasets and outperforms

competitive approaches. Future work concerns the automatic determination of parameter K as a trade-off between speed and performance as well as the application of the proposed method to handwritten historical documents.

References

1. Ataer E, Duygulu P (2007) Matching ottoman words: an image retrieval approach to historical document indexing. In: Proceedings of the 6th ACM international conference on image and video retrieval, pp 341–347
2. Cao H, Govindaraju V (2007) Template-free word spotting in low-quality manuscripts. In: 6th international conference on advances in pattern recognition (ICAPR'07), pp 45–53
3. Everingham M, Gool LV, Williams C, Winn J, Zisserman A (2007) The PASCAL visual object classes challenge 2007 results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html
4. Fishler MA, Bolles RC (1981) A paradigm for model fitting with applications to image analysis and automated cartography. Commun Assoc Comput Mach 24(6):381–395
5. Gatos B, Pratikakis I (2009) Segmentation-free word spotting in historical printed documents. In: 10th international conference on document analysis and recognition (ICDAR'09), Barcelona, Spain, pp 271–275
6. Hartley R, Zisserman A (2003) Multiple view geometry in computer vision, 2nd edn. Cambridge University Press, Cambridge
7. Jawahar CV, Balasubramanian A, Meshesha M (2004) Word-level access to document image datasets. In: Proceedings of the workshop on computer vision, graphics and image processing (WCVGIP), pp 73–76
8. Keaton P, Greenspan H, Goodman R (1997) Keyword spotting for cursive document retrieval. In: Workshop on document image analysis, San Juan, Puerto Rico, pp 74–81
9. Kim S, Park S, Jeong C, Kim J, Park H, Lee G (2005) Keyword spotting on korean document images by matching the keyword image. In: Digital libraries: implementing strategies and sharing experiences, vol 3815, pp 158–166
10. Kluzner V, Tzadok V, Shimony Y, Walach E, Antonacopoulos A (2009) A complete optical character recognition methodology for historical documents. In: Tenth international conference on document analysis and recognition (ICDAR), Barcelona, pp 501–505
11. Koerich A, Sabourin R, Suen CY (2003) Devanagari ocr using a recognition driven segmentation framework and stochastic language models. Int J Doc Anal Recognit (IJDAR) 6(2):126–144
12. Kolcz A, Alspector J, Augusteijn M, Carlson R, Popescu GV (2000) A line-oriented approach to word spotting in handwritten documents. J Pattern Anal Appl 3(2):153–168
13. Konidaris T, Gatos B, Ntzios K, Pratikakis I, Theodoridis S, Perantonis SJ (2007) Keyword-guided word spotting in historical printed documents using synthetic data and user feedback. Int J Doc Anal Recognit (IJDAR) 9(24):167–177 (special issue on historical documents)
14. Konidaris T, Gatos B, Perantonis SJ, Kesidis A (2008) Keyword matching in historical machine-printed documents using synthetic data, word portions and dynamic time warping. In: The eighth IAPR workshop on document analysis systems, pp 539–545
15. Kornfield EM, Manmatha R, Allan J (2007) Further explorations in text alignment with handwritten documents. Int J Doc Anal Recognit (IJDAR) 10(1):39–52
16. Lebourgeois F, Henry J-L, Emptoz H (1992) An ocr system for printed documents. In: Proceedings of IAPR workshop on machine vision applications, Tokyo, Japan, pp 83–86
17. Leydier Y, LeBourgeois F, Emptoz H (2007) Text search for medieval manuscript images. Pattern Recognit 40:3552–3567
18. Leydier Y, Ouji A, LeBourgeois F, Emptoz H (2009) Towards an omnilingual word retrieval system for ancient manuscripts. Pattern Recognit 42(9):2089–2105
19. Li L, Lu SJ, Tan CL (2007) A fast keyword-spotting technique. In: Ninth international conference on document analysis and recognition (ICDAR), pp 68-72
20. Lladós J, Sánchez G (2007) Indexing historical documents by word shape signatures. In: Ninth international conference on document analysis and recognition (ICDAR), pp 362–366
21. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
22. Marcolino A, Ramos V, Ramalho M, Pinto JC (2000) Line and word matching in old documents. In: Fifth IberoAmerican symposium on pattern recognition (SIAPR), pp 123–135
23. Meshesha M, Jawahar CV (2008) Matching word images for content-based retrieval from printed document images. Int J Doc Anal Recognit (IJDAR) 11(1):29–38
24. Murugappan A, Ramachandran B, Dhavachelvan P (2011) A survey of keyword spotting techniques for printed document images. Artif Intell Rev 35(2):119136
25. Natarajan P, Bazzi I, Lu Z, Makhoul J, Schwartz RM (1999) Robust ocr of degraded documents. In: International conference on document analysis and recognition (ICDAR), pp 357–361
26. Rath TM, Manmatha M (2007) Word spotting for historical documents. Int J Doc Anal Recognit (IJDAR) 9(24):139–152
27. Rath TM, Manmatha R (2003) Features for word spotting in historical manuscripts. In: International conference of document analysis and recognition, pp 218–222
28. Rath TM, Manmatha R (2003) Word image matching using dynamic time warping. In: Computer vision and pattern recognition, pp 521–527
29. Rosten E, Drummond T (2006) Machine learning for high-speed corner detection. In: European conference on computer vision, vol 1, pp 430–443
30. Rusinol M, Aldavert D, Toledo R, Lladós J (2011) Browsing heterogeneous document collections by a segmentation-free word spotting method. In: 11th international conference on document analysis and recognition (ICDAR'11), China, pp 63–67
31. Srihari SN, Srinivasan H, Huang C, Shetty S (2006) Spotting words in latin, devanagari and arabic scripts, Indian. J Artif Intell 16(3):2–9
32. Terasawa K, Tanaka Y (2009) Slit style hog feature for document image word spotting. In: 10th international conference on document analysis and recognition (ICDAR'09), pp 116–120
33. TREC_EVAL. http://trec.nist.gov/trec_eval
34. von Eckartshausen C (1778) Aufschlüsse zur Magie aus geprüften Erfahrungen über verborgene philosophische Wissenschaften und verdeckte Geheimnisse der Natur. Bavarian State Library
35. Yalniz IZ, Manmatha R (2011) A fast alignment scheme for automatic ocr evaluation of books. In: 11th international conference on document analysis and recognition (ICDAR'11), Beijing, China, pp 754–758
36. Yalniz IZ, Manmatha R (2012) An efficient framework for searching text in noisy document images. In: Proceedings of document analysis systems (DAS), pp 48–52
37. Zhang B, Srihari SN, Huang C (2004) Word image retrieval using binary features. In: Document recognition and retrieval XI (SPIE), pp 45–53