Pattern Recognition 68 (2017) 310–332

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

A survey of document image word spotting techniques

Angelos P. Giotis[a,b,*], Giorgos Sfikas[b], Basilis Gatos[b], Christophoros Nikou[a]

[a] Department of Computer Science and Engineering, University of Ioannina, Greece
[b] Computational Intelligence Laboratory, Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos", GR-15310 Athens, Greece

ARTICLE INFO

Article history:
Received 26 February 2016
Revised 5 November 2016
Accepted 21 February 2017
Available online 27 February 2017

Keywords:
Word spotting
Retrieval
Document indexing
Features
Representation
Relevance feedback

ABSTRACT

Vast collections of documents available in image format need to be indexed for information retrieval purposes. In this framework, word spotting is an alternative solution to optical character recognition (OCR), which is rather inefficient for recognizing text of degraded quality and unknown fonts usually appearing in printed text, or writing style variations in handwritten documents. Over the past decade there has been a growing interest in addressing document indexing using word spotting which is reflected by the continuously increasing number of approaches. However, there exist very few comprehensive studies which analyze the various aspects of a word spotting system. This work aims to review the recent approaches as well as fill the gaps in several topics with respect to the related works. The nature of texts and inherent challenges addressed by word spotting methods are thoroughly examined. After presenting the core steps which compose a word spotting system, we investigate the use of retrieval enhancement techniques based on relevance feedback which improve the retrieved results. Finally, we present the datasets which are widely used for word spotting, we describe the evaluation standards and measures applied for performance assessment and discuss the results achieved by the state of the art.

1. Introduction

A great amount of information in libraries and cultural institutions exist all over the world and need to be digitized so as to preserve it and protect it from frequent handling. Among others, Google has put an effort to digitize books on a large scale [1,2], thereby providing support to the document understanding research community. In order to create digital libraries which allow efficient searching and browsing for future users, thousands of digitized documents have to be transcribed or at least indexed at a certain degree. However, the automatic recognition of poor quality printed text and especially, handwritten text, is not feasible by traditional OCR approaches which mainly suffice for modern printed documents with simple layouts and known fonts. Most of the constraints encountered by recognition systems stem from difficulties in segmenting characters or words, the variability of the handwriting and the open vocabulary. For this reason, more flexible information retrieval and image analysis techniques are required.

1.1. Document indexing using image retrieval methods

The actual problem behind building digital libraries lies on the retrieval of digitized documents in terms of reliable extraction and access to specific information. While a document image processing system analyzes different text regions so as to convert them to machine-readable text using OCR, a document image retrieval system searches whether a document image contains particular words of interest, without the need for correct character recognition, but by directly characterizing image features at character, word, line or even document level.

On one hand, *recognition-based* retrieval relies on the complete recognition of documents either at character level using OCR, or at word level using *word recognition* methods. In the latter case, the goal is to correctly classify a query word into a labeled class, or else, obtain its transcription. Most methods of this type require prior transcription of text-lines, words or characters to train character or word models. During the search phase, a text dictionary or lexicon is used and only words from that lexicon can be used as candidate transcriptions in the recognition task. These methods usually rely on hidden Markov models (HMMs) [3,4], conditional random fields (CRFs) [5], neural networks (NNs) [6,7] or they might follow a hybrid approach by combining different classifiers, such as support vector machines (SVMs) with HMMs [8,9] or HMMs with NNs [10]. An obvious drawback of these approaches is that they

* Corresponding author at: Department of Computer Science and Engineering, University of Ioannina, Greece.
E-mail addresses: agiotis@cs.uoi.gr (A.P. Giotis), sfikas@iit.demokritos.gr (G. Sfikas), bgat@iit.demokritos.gr (B. Gatos), cnikou@cs.uoi.gr (C. Nikou).

http://dx.doi.org/10.1016/j.patcog.2017.02.023
0031-3203/

have to deal with the inherent handwriting variability and handle a large number of word and character models. Nevertheless, the scope of this work does not focus on recognition-based retrieval methods and thus, we only briefly refer to them.

On the other hand, the *recognition-free* retrieval which is also known in the literature as *word spotting* or *keyword spotting* is the main subject of this study. The goal here is to retrieve all instances of user queries in a set of document images which may be segmented at text lines or words. Actually, the user formulates a query and the system evaluates its similarity with the stored documents and returns as output a ranked list of results which are most similar to the query. The process is totally based on matching between common representations of features, such as color, texture, geometric shape or textual features, while conversion of whole documents into machine readable format and recognition do not take place at all. Therefore, the selection and use of proper features and robust matching techniques are the most important aspects of a word spotting system.

Word spotting methods may be divided into multiple categories according to various factors. Depending on how the input is specified by the user we can distinguish *query-by-example* (QBE) from *query-by-string* (QBS) methods. In the QBE scenario, the user selects an image of the word to be searched in the document collection, whereas in the QBS paradigm, the user provides an arbitrary text string as input to the system. Another way to categorize word spotting methods depends on whether training data are used offline, either to learn character and word models or tune the parameters of the system. This way we can distinguish *learning-based* from *learning-free* approaches. Finally, word spotting methods which can be directly applied to whole document pages are considered as *segmentation-free*, in contrast with *segmentation-based* methods, where a segmentation step has to be applied at line or word level during preprocessing.

Word spotting was initially proposed in the speech recognition community [11]. Its application was adopted later on for printed [12,13] and handwritten [14] document indexing. While early approaches were based on raw features extracted directly from image pixels [14,15], the state of the art is to characterize document images with more complex features based on gradient information, shape structure, texture, etc. (see Section 4.1).

1.2. Applications

There are a variety of applications of word spotting for document indexing and retrieval including the following:

- retrieval of documents with a given word in company files,
- searching online in cultural heritage collections stored in libraries all over the world,
- automatic sorting of handwritten mail containing significant words (e.g. “urgent”, “cancelation”, “complain”) [16],
- identification of figures and their corresponding captions [17],
- keyword retrieval in pre-hospital care reports (PCR forms) [18],
- word spotting in graphical documents such as maps [19],
- retrieval of cuneiform structures from ancient clay tablets [20],
- assisting human transcribers in identifying words in degraded documents, especially those appearing for the first time.

Although word spotting and word recognition belong to two separate retrieval paradigms, they sometimes interact by assisting one another. For instance, the authors in [21] propose a keyword spotting approach relying on a NN-based recognition system. On the contrary, in [22], word spotting contributes as a means of bootstrapping a handwriting recognition system, in terms of selecting new elements from the retrieved results. These elements can be used to augment the training set through a semi-supervised procedure, thus increasing the final recognition accuracy while at the same time avoiding the costly manual annotation process.

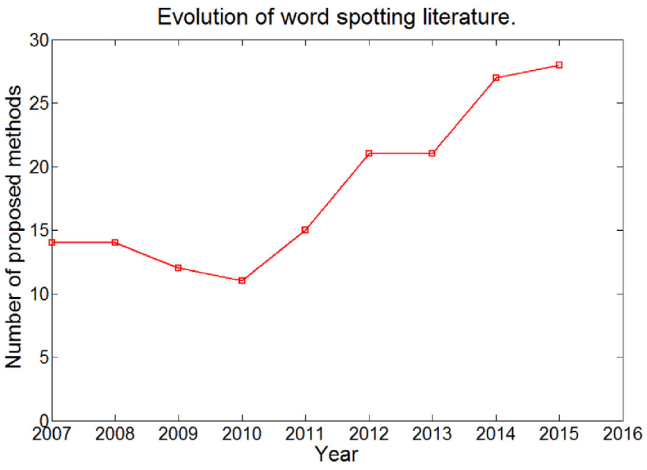

Fig. 1. Word spotting approaches published over the last decade.

1.3. Evolution of the related works

In order to track the recent literature, we present some statistics related to the evolution of word spotting methods over the last decade. The research community concentrates on indexing historical documents on a grand scale using word spotting and thus, we consider that the whole process remains an open problem. To the best of our knowledge, Fig. 1 provides a concise view of the various word spotting approaches for offline, handwritten or printed documents, which were published in conferences and journals since 2007. As it can be seen in Fig. 1, there is an increased number of papers over the past few years which confirms the growing interest of the community in word spotting.

1.4. Contributions and outline of the paper

Apart from the proposed methods, there also exist a number of surveys for word spotting, either for a specific script, or a particular domain (machine-printed, handwritten), or even for a variety of applications. Murugappan et al. [23] present a study for word spotting in printed documents. The authors divide the word spotting methods according to a character-based and a word-based representation depending on the features used in each case. Their work implies that character-based approaches provide satisfactory results if character segmentation is easy to obtain, whereas word-based approaches can deal with touching characters efficiently and analyze the shapes of the words without explicit character recognition. In addition, a comparative study for segmentation and word spotting methods is presented in [24] for handwritten and printed text in Arabic documents. The segmentation techniques rely on horizontal and vertical profile features and scale space segmentation. The features under comparison are geometrical moments and word profiles, whereas the similarity computation is carried out using the cosine metric and dynamic time warping (DTW). An explicit view of the various aspects of a word spotting system is presented by Marinai et al. [25]. Therein, the different features used for each technique are categorized according to the layer at which the similarity computation is applied (pixel/column features, connected components, word level features etc.). Image representations (i.e. feature vectors) with respect to the specific feature types are also analyzed along with the respective similarity measures. Finally, the work of Tan et al. [26] underlines the necessity for content-based image retrieval as an economical alternative to OCR,

relying on proper selection of features, representation and similarity measures. Word spotting is defined under a framework of categories with respect to the word image representation.

Nevertheless, a considerable number of word spotting approaches proposed over the last years as well as several techniques involved for the improvement of the performance yet remain unexplored. This survey aims to review the recently proposed methods and complete the missing parts of other studies in the word spotting literature. To this end, we analyze the nature of text sources along with the inherent difficulties addressed by word spotting methods. Among the main steps of the word spotting system, namely, feature extraction, representation and similarity computation, we also investigate the preprocessing stage with respect to binarization, segmentation and normalization techniques. Furthermore, we present the benefits accrued from relevance feedback methods employed in the retrieval phase of a word spotting task, either by involving the user to select true query instances or in a completely unsupervised way. Subsequently, we examine whether direct comparison among different methods is straightforward or not, since the evaluation measures and protocols applied for assessing the performance may differ substantially. Finally, we present the most commonly used datasets along with the experimental results published by the state-of-the-art methods and discuss about the performance obtained in each case.

The rest of our work is structured as follows. In Section 2, we describe the challenges involved in document image word spotting. In Sections 3 and 4, we present the core steps of the word spotting pipeline with respect to the preprocessing and feature extraction, the input of the spotting system and the different similarity metrics applied among common representations upon the extracted features. Section 5 describes a number of techniques which enhance the retrieved results from the image matching step based on relevance feedback, data fusion and re-ranking. In Section 6, we present the most common datasets along with some distinct measures used to evaluate word spotting systems and examine the results achieved by the state-of-the-art methods. Finally, conclusions are drawn in Section 7.

2. Challenges in document image word spotting

Keyword spotting in document images presents several challenges which are related to the nature of the original documents. In this section, we first investigate the various text sources used by word spotting methods and subsequently overview the corresponding challenges.

2.1. Nature of text addressed in word spotting

Regarding the nature of documents which have been addressed so far by the research community for word spotting, we can distinguish various categories depending on factors such as the age of the text, its alphabet, the underlying language and the source which created the text (e.g. human or machine). Table 1 illustrates the various scripts addressed by most of the representative works for word spotting, during the period considered in this work.

Historical documents typically contain text written in a language that is no longer in use. Contrary to *modern* documents, the alphabet, the writing style or the accents are different. Historical documents usually suffer from degradations such as stained paper, faded ink or ink bleed through, winkles and unknown graphical symbols, as opposed to modern text, thus hampering the readability and in turn the word spotting process.

So far, word spotting has been applied to various scripts, such as Arabic, Chinese, Devanagari, Greek and Latin. These scripts differ from each other owing to factors such as the writing direction, the size of the alphabet (number of characters), possible diacritic marks (polytonic Greek text) and cursiveness. For example, documents in Arabic scripts are written from right to left, in horizontal direction and are fully cursive. On the contrary, text in Latin script is written from left to right in horizontal direction only, cursively in some cases. Chinese scripts contain thousands of characters and are written in two dimensions, either from left to right horizontally, or from top to bottom vertically. Devanagari scripts are written horizontally, from left to right in a complex cursive way, whereas Greek scripts are written from left to right without cursiveness. Furthermore, each separate character of the Chinese scripts has specific meanings or semantics, in contrast with the isolated characters of other scripts.

Many of the proposed techniques for word spotting in a specific language may be directly applied to a different language on the ground that it is written in a relevant script. However, the application of a word spotting method in different scripts is not straightforward, since it heavily depends on the features which are extracted before image matching takes place. For instance, profile or pixel-based features [37,92] are suitable for obtaining representations which allow for word spotting in heterogeneous documents regardless of the underlying language. This is contrasted with structural features and shape codes [88,103] which are defined to capture the specific shapes of the writing symbols of a language.

One other aspect of the documents addressed by word spotting techniques is related to the creation of the respective text. *Handwritten* documents, either historical or modern, always suffer from variability in writing style, not only among different authors but also for documents of the same writer. This is not the case though for *machine-printed* text where variations mainly concern the font type. An exception is the case of *woodblock-printed* documents of Chinese and Mongolian scripts which present intra-writer variability for the same author. Word spotting in handwritten text is generally considered more challenging than spotting printed text, as apart from variations in writing style, handwriting is also unconstrained. For instance, words may be skewed, characters may be slanted, non-text content such as symbols may be present and letters may be broken or connected in a cursive manner. Nevertheless, historical printed documents also present challenges for word spotting because of degradations such as missing data, non-stationary noise due to illumination changes during the scanning process, low contrast, show through or warping effects etc.

Indexing documents contained in large databases around the globe is not the only area of application for image retrieval methods. Online handwritten text presents a growing significance due to the increasing use of PDAs, Tablet PCs, and digital pens. Understanding such documents may be useful, for instance, in the case of a smart meeting room which allows participants to search, browse or organize handwritten notes taken with digital pens during a meeting. However, an important difference between online and offline text lies on the features which are extracted from each of the respective sources. Instead of focusing on color, texture or geometric shape, features related to the pen tip trace and the stroke's characteristics are extracted, such as its width, height, the pen's pressure and others. Example works for online text can be found in the literature, regarding either word spotting [104,105] or recognition [106]. In this work though, we only consider offline documents.

2.2. Challenges addressed by existing methods

Degradations involved in historical documents, pre-hospital care reports and other text sources hinder the overall performance of a word spotting system. For example, low image quality directly affects the following segmentation and feature extraction stages of a word spotting system.

Table 1
Text sources addressed by word spotting methods.

Publications	Context	Language	Script	Type
Aldavert et al. [27,28], Zagoris et al. [29]	Historical	English	Latin	Handwritten
Zhang and Tan [30], Fornés et al. [31]	Historical	English	Latin	Handwritten
Roy et al. [32], Rothacker et al. [33,34]	Historical	English	Latin	Handwritten
Mondal et al. [35], Dovgalecs et al. [36]	Historical	English	Latin	Handwritten
Rath et al. [37], Zhong et al. [38]	Historical	English	Latin	Handwritten
Cao et al. [18], Wagan et al. [39]	Modern	English	Latin	Handwritten
Kumar et al. [40]	Modern	English	Latin	Handwritten
Retsinas et al. [41], Krishnan et al. [42]	Historical, modern	English	Latin	Handwritten
Almazán et al. [43], Liang et al. [44]	Historical, modern	English	Latin	Handwritten
Wilkinson et al. [45], Fischer et al. [46]	Historical, modern	English	Latin	Handwritten
Ghosh and Valveny [47]	Historical, modern	English	Latin	Handwritten
Kessentini et al. [48,49], Choisy [50]	Modern	French	Latin	Handwritten
Howe [51,52], Frinken et al. [21]	Historical	English, German	Latin	Handwritten
Puigcerver et al. [53], Riba et al. [54]	Historical	Spanish	Latin	Handwritten
Fink et al. [55]	Historical	German	Latin	Handwritten
Chatbri et al. [56]	Modern	French	Latin	Handwritten, machine-printed
Lladós et al. [57], Wang et. al. [58]	Historical	English, Spanish	Latin	Handwritten
Oosten et al. [59], Der Zant et al. [60]	Historical	Dutch	Latin	Handwritten
Kovalchuk et al. [61], Almazán et al. [62]	Historical	English	Latin	Handwritten, machine-printed
Mondal et al. [63,64]	Historical	English, French	Latin	Handwritten, machine-printed
Sfikas et al. [65]	Historical	Greek	Greek	Handwritten, machine-printed
Rodríguez-Serrano and Perronnin [66]	Historical, modern	English, French, Arabic	Latin, Arabic	Handwritten
Sudholt et al. [67]	Historical, modern	English, Spanish, Arabic	Latin, Arabic	Handwritten
Leydier et al. [68]	Historical	Middle English, Semitic, Chinese	Latin, Arabic, Chinese	Handwritten
Terasawa and Tanaka [69]	Historical	English, Japanese	Latin, Chinese	Handwritten
Abidi et al. [70], Sagheer et al. [71]	Historical	Urdu	Arabic	Handwritten
Khayyat et al. [72], Li et al. [73]	Modern	Farsi	Arabic	Handwritten
Kumar et al. [74], Wshah et al. [75]	Modern	English, Urdu, Hindi	Latin, Arabic, Devanagari	Handwritten
Srihari and Ball [76]	Modern	English, Urdu, Hindi	Latin, Arabic, Devanagari	Handwritten
Huang et al. [77]	Modern	Chinese	Chinese	Handwritten
Giotis et al. [78]	Modern	Greek	Greek	Handwritten
Saabni et al. [79]	Modern	Arabic	Arabic	Handwritten
Shah et al. [80]	Modern	Pashto	Arabic	Handwritten
Can and Duygulu [81], Rusin˜ol et al. [82]	Historical	English, Ottoman	Latin, Arabic	Handwritten, machine-printed
Wei et al. [83,84]	Historical	Kanjur	Mongolian	Woodblock-printed
Ranjan et al. [85], Li et al. [86]	Modern	English	Latin	Machine-printed
Zagoris et al. [87], Bai et al. [88]	Modern	English	Latin	Machine-printed
Louloudis et al. [89], Roy et al. [90]	Historical	French	Latin	Machine-printed
Papandreou et al. [91]	Historical	French	Latin	Machine-printed
Gatos and Pratikakis [92]	Historical	German	Latin	Machine-printed
Sousa et al. [93]	Historical	Portuguese	Latin	Machine-printed
Marinai [94]	Historical	Latin	Latin	Machine-printed
Konidaris et al. [95], Kesidis et al. [96]	Historical	Greek	Greek	Machine-printed
Xia et al. [97]	Historical	Chinese	Chinese	Machine-printed
Hassan et al. [98], Krishnan et al. [99]	Modern	English, Indian, Gujarati	Latin, Bangla, Devanagari	Machine-printed
Shekhar et al. [100], Yalniz et al. [101]	Modern	English, Indian	Latin, Telugu	Machine-printed
Meshesha and Jawahar [102]	Modern	English, Amharic, Hindi	Latin, Amharic, Devanagari	Machine-printed

Apart from possible degradations, handwritten documents usually present high variability in writing style, meaning, the same query word may differ substantially among its instances. This calls for features which are distinctive enough to be detected inside the query instances, yet not too dependent on a specific writing style. Most methods that deal with multi-writer word spotting rely on annotated data to learn a model able to capture the basic structure or semantic information of a word, regardless of the writing style.

The need for adequate training data poses another challenge for word spotting, since they are not always easy to obtain. For instance, historical handwritten documents are unconstrained and thus often render the transcription process difficult even for palaeographers. Methods that do not require training present a solid advantage in this respect.

Text cursiveness found in handwritten documents, overlapping sub-word components existing in Arabic scripts and many punctuation marks or graphical symbols lying in historical documents may lead to inaccurate segmentations. In that sense, methods that avoid potential error-prone segmentations tackle this challenge.

It is often expected that the user has to find a particular instance of a query in order to initiate the search for similar instances. In some cases though, it is more preferable for the user to insert an arbitrary string to be searched for. However, there are QBS methods which are not able to perform out of vocabulary (OOV) word spotting, namely, only a limited number of keywords, which are known during training, can be used as queries.

In Section 2.1, we mentioned the dependence of a word spotting method on a specific language, let alone a particular alphabet. A learning-based method able to perform well in different languages for a relevant script is not essentially suitable for a different script, unless new training data are used. On the contrary, script-independent approaches deal with this matter.

Table 2
Challenges addressed by word spotting methods.

Challenges	Publications
Robustness to degradations	[18,21,27–38,41–47,51–55,57–71,81–84,89–97]
Multi-writer conditions	[18,21,28,38,40,41,43,45–50,52,54,55,57,58,66,67,70–81,83,84]
Learning-free scenario	[28–31,33,36,37,39,41,51,54–58,61–64,68–70,80–82,84,86–92,95,96,99,101]
Segmentation-free methods	[30,33,34,36,47,54–56,61,62,68,82,92]
OOV spotting (QBS)	[21,27,32,34,38,42–49,52,53,65,67,68,88]
Script-independence	[28,41,66,68,69,74–76,81,82,98–102]
Chinese/Japanese characters	[68,69,77,83,84,97]
Scalability	[27,29,31,32,34–39,41–45,47,48,52–55,57,61,62,65,67,82,89,90,92,99]

Chinese and Japanese documents have a large number of character classes (almost over 5000) and they present no explicit differences between inter-character and inter-word spaces. To cope with this challenge some keyword spotting methods follow the strategy of over-segmenting the text lines into primitive segments and adopt a character classifier to assign a small number of high-confidence classes to the input character pattern.

Word spotting methods need to be accurate enough for successful indexing while at the same time fast enough for high scalability. One way to achieve computationally efficient retrieval is to use fixed-length feature representations, since they are faster to compare than variable length sequences, as we will discuss in Section 4.2. Table 2 summarizes the aforementioned challenges along with the respective key methods which address them.

3. Basic document image analysis technologies involved

Although the intermediate stages of a word spotting system may vary across different methods, we can distinguish some common steps. Document images are initially preprocessed in order to enhance the subsequent feature extraction step. After appropriate features have been extracted, a common representation is selected to describe both the documents at a specific level (word, line or page) and the query, which in most cases is a single word provided either as an image or a text string. The next part lies on the matching algorithm applied between the representations of the query and the documents. This matching outcome is used at a later stage for retrieving the desired information. In the following, we will discuss some basic technologies involved during the preprocessing step.

3.1. Binarization

Binarization is the starting step of most word spotting systems and refers to the conversion of the original input to a binary black-and-white image. It can provide a good starting point for segmentation as well as feature extraction. For instance, some methods which perform text-line segmentation using connected components analysis require the documents to be properly binarized. Similarly, contour-based features extracted from skeletons or outer contours are heavily dependent on the binarization outcome.

Otsu's global thresholding [107] is one of the most commonly used binarization techniques in the literature [19,87,103,108–110]. This method selects a global threshold value from all possible thresholds as the one minimizing the intra-class variance of the thresholded black and white pixels. Can et al. [81] obtained similar results to Otsu's method using another global thresholding technique in which the threshold is based on the mean intensity value of the gray-scale image. Other global thresholding approaches can be found in [47,61,93,111].

In the case though of degraded document collections which usually suffer from non-uniform illumination, image contrast variation, bleeding-through or smear effects, more efficient local thresholding techniques are required. For instance, Sauvola's technique [112] calculates a local threshold which is adapted to the neighborhood of each pixel according to the local mean value and the local standard deviation inside the neighborhood which is defined by a sliding window. Methods based on local thresholding can be found in [40,44,113–115]. Some methods also include an image enhancement step. Fink et al. [55] preprocess images to improve the overall contrast between the script and the document background. To this end, they employ histogram equalization to the intensity channel in an YCrCb color space and subsequently use a 9x9 median filter to reduce the background noise. Kumar et al. [40] normalize the background light intensity using an adaptive linear or non-linear function [116] that best fits the background. The background normalized image is further enhanced by Histogram Normalization. Finally, the normalized image is binarized using an adaptive thresholding algorithm. Cao et al. [18] follow a probabilistic approach [117] to binarize documents and remove inherent grid lines. They model degraded images with MRFs where the prior is learnt from a training set of high quality binarized images, whereas the probabilistic density is learnt on-the-fly from the gray-level histogram of input images.

Several state-of-the-art approaches for binarizing degraded documents rely on hybrid schemes which combine global and local thresholding. The authors in [64,92,95,96] use the technique proposed in [118] which consists of five steps: a preprocessing procedure using a low-pass Wiener filter, a rough estimation of foreground regions, a background surface calculation by interpolating neighboring background intensities, a thresholding by combining the calculated background surface with the original image and finally a post-processing step that improves the quality of text regions and preserves stroke connectivity. Wei et al. [83] make use of three global thresholding methods to extract regions of interest (ROI) from gray-level images. Each ROI is then processed by a modified Sauvola's algorithm with variant sizes of the small windows. Howe [51] employs the method proposed in [119] which optimizes a global energy function based on the Laplacian operator upon the local likelihood of foreground and background labels, the Canny edge detection to identify likely discontinuities and a graph cut implementation to find the minimum energy solution of the objective function.

However, there is often a tradeoff between the amount of missing data and accurate data after binarization is applied and therefore some works [30,33,37,68,69,80,82,120–123] prefer to perform directly on the gray-scale image. For example, Zhang et al. [30] propose an illumination invariant descriptor of gray-scale document images using features extracted from keypoints. If the images suffer from low resolution, the authors report a low number of detected keypoints which in turn yields a reduced number of retrieved query instances, despite the high accuracy of those retrieved. Leydier et al. [68,121] prefer to separate the text from the background using a gradient norm threshold instead of binarizing the document image. This renders their proposed gradient-based features more informative in high magnitude zones computed on the gray-scale image. Similarly, Terasawa and Tanaka [69] deem

a background removal more suitable for their method. The background is removed using a simple thresholding technique such that the graylevel information which is important for the proposed gradient-based features is not affected. To exploit fast matching algorithms which can only be applied to binary images, Shah and Suen [80] extract features directly from gray-scale images and then convert the resulting feature vectors into their binary equivalents using an encoding scheme with four bits per feature value. A drawback of this approach is that the final feature vector has high dimensionality. Cao et al. [120] consider the gray-scale images more preferable when dealing with heavily degraded documents, such as carbon medical forms, where the binarized version is not even readable by a human.

Other methods [70,79] work on both gray-scale and binary images aiming to combine the advantages of each type. Abidi et al. [70] employ a set of profile-based features which can be extracted from either gray-scale or binary images to match partial words in Arabic script. To examine the discriminative power of each independent feature, they evaluate the retrieval percentage of five features obtained from binary images and one feature extracted from the gray-scale version, which proved to outperform the other five features. Nevertheless, the authors report that the combined information from all features improves the word spotting performance. Saabni and Bronstein [79] propose a multi angular descriptor of either binary or gray-scale word images. The descriptor is based on multiple view points obtained from rings out of the shape of the word and therefore is not significantly affected by the binarization step.

3.2. Segmentation

Segmentation-based word spotting methods involve a segmentation preprocessing stage in order to segment the document pages at word or line level. Although segmentation can be considered as a simple task for modern machine-printed documents, segmentation of handwritten or historical documents is still an open research problem due to the significant challenges that are involved. These include variations in inter-line or inter-word gaps, overlapping and touching text parts, existence of accents, punctuation marks and decorative letters, local text skew and slant.

In the following, we present a categorization of the general text line techniques together with one representative reference per each category. (a) Projection-based methods: the horizontal image projections are analyzed in order to detect hills (correspond to text lines) and valleys (correspond to white spaces between text lines). Although these methods are usually applied to machine-printed documents, they can also be used for handwritten documents [124]. (b) Smearing methods: the white runs in a certain direction are analyzed and eliminated under several conditions [125]. (c) Grouping methods: low-level elements such as pixels or related components are grouped together based on several rules [126]. (d) Methods based on Hough transform: a set of points is projected to the Hough space in order to detect lines [127].

Concerning word segmentation, the proposed techniques usually first calculate the distances of adjacent components using the bounding box, the Euclidean, the run-length or the convex hull distance [128]. At a next step, these distances are classified as inter-word or intra-word [129].

Some segmentation-based word spotting methods assume that datasets are already segmented to text lines or words while others perform a respective segmentation step. Example word spotting methods based on horizontal projection profiles for text line separation, followed by vertical profiles for word segmentation can be found in [44,98,108,130–132]. Rodriguez-Serrano and Perronin [16] use horizontal projection profiles to obtain text lines. For each line they compute the convex hulls of connected components and define a distance between neighboring components as the minimum distance between their convex hulls. Distances larger than a threshold are likely to correspond to word gaps. Kumar et al. [40] extract text lines using the algorithm proposed by Shi et al. [133] which uses a steerable filter to convert a down-sampled version of the input document image into an Adaptive Local Connectivity Map (ALCM). Connected component based grouping is done to extract each text line. Word segmentation is then done by finding convex hulls for each connected component and learning the distribution over the distances between the centroids of the convex hulls for within and between word gaps.

The Run Length Smoothing Algorithm (RLSA) [134] is a common smearing technique for segmenting document pages into text lines and words. RLSA examines the white runs existing in the horizontal and vertical directions. For each direction, white runs with length less than a threshold are eliminated. The horizontal and vertical length thresholds are usually defined proportionally to the average character height. The application of RLSA results in a binary image where characters of the same word become connected to a single connected component. Then, a connected component analysis is applied in order to extract the final word segmentation result. Example works using RLSA can be found in [63,64,95,96]. RLSA works well for printed documents but usually presents poor results in handwritten historical documents where inter-word spaces are variable. Mondal et al. [64] evaluate a number of DTW-based sequence alignment techniques under conditions of perfect (manual) and error-prone (RLSA-based) word segmentations and confirm that DTW works well only in the first case. Otherwise, they propose a Continuous Dynamic Programming (CDP) method which performs robust partial matching at line (or piece of line) level.

Most works in Arabic scripts [71,72,109,135] are only able to perform on partial word level. Pieces of Arabic Words (PAW) are obtained either manually or from connected component analysis on the segmented words. Each word in the Arabic script consists of one or more PAW, each of which contains only one major connected component (CC) and some or none minor CCs. These minor CCs are often called diacritics and dots. Major and minor CCs can be distinguished by their size and location. Khayyat et al. [72] smear the documents with a morphological dilation using a binary dynamic adaptive mask [136] to extract text lines. Then they extract major and minor components from PAW.

Chinese scripts also show variations between inter-character and inter-word spaces. Most keyword spotting methods follow the strategy of over-segmenting the text lines into primitive segments. For instance, Huang et al. [77] segment the document image into text lines using a graph-based clustering algorithm [137]. Each line is then over-segmented into primitive segments using the algorithm of [138]. Candidate characters generated by concatenating consecutive segments form a candidate segmentation lattice.

In a language independent scenario, Srihari et al. [76] perform text line segmentation using a clustering method. For word segmentation, the problem is formulated as a classification problem as to whether or not the gap between two adjacent CCs in a line is word gap or not. An artificial neural network with features characterizing the CCs was used for this classification task.

3.3. Normalization

The segmentation is usually followed by a normalization step in which several variabilities are removed. For instance, handwritten documents present challenges such as text skew and slant or warping effects accrued during the scan process. Wang et al. [139] handle text skew by combining projection profiles with Hough transform to separate the text according to the skew angle of each line. To cope with different writing styles, most ap-

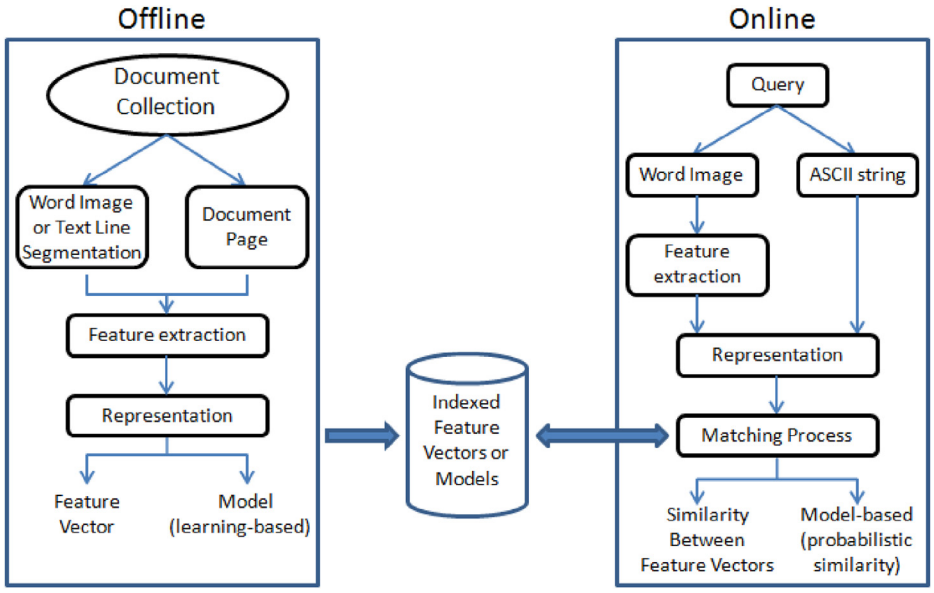

Fig. 2. General word spotting system architecture.

proaches based on line segmentation [21,46,140], as well as word-based methods, such as the works of Rodríguez-Serrano and Perronnin [16,66], determine the skew angle by a regression analysis based on the bottom-most black pixel of each image column extracted via a sliding window. Then, the skew of the text line is corrected by rotation. After estimating the slant angle based on a histogram analysis, a shear transformation is applied to the image. Moreover, a vertical scaling procedure is applied to normalize the height with respect to the lower and upper baseline and finally, horizontal scaling normalizes the width of the text line with respect to the estimated number of letters. Scale normalization at word level is also applied for handwritten and printed documents. In [95,96], the segmented words are resized to fit in a fixed bounding-box while preserving their aspect ratio, whereas in [61] each candidate word is resized to fit a fixed-size rectangle regardless of its size and aspect ratio.

4. Keyword spotting system architecture

In this section, we examine the main steps of the word spotting pipeline. Fig. 2 illustrates a general purpose word spotting system where the whole procedure is divided in an offline and an online phase. In the offline stage, features are extracted from word images, text lines or whole pages which are then represented by feature vectors. In the case where training data are used, feature vectors are usually modelled with statistical models (e.g. HMMs). In the online phase, a user formulates a query either by selecting an actual example from the collection (QBE), or by typing an ASCII text word (QBS). Depending on the query type, a common representation with that of the offline phase is used to describe the query and then a matching process is applied between these representations in order to obtain a similarity score which in turn yields a ranking list of results according to their similarity with the query.

The most common distinction of word spotting approaches depends on how the input is specified. Each type (QBE or QBS) has its own merits and handicaps. One obvious drawback of QBE methods is that the search is constrained for words that appear at least once in a document collection since an actual instance of the query word is required to trigger it. QBS approaches on the other hand allow arbitrary textual queries without the need to find a particular query occurrence. Herein, the keyword representation is usually accrued from trained character models. However, in the case where labeled data are not available or inadequate, an alternative solution is to artificially generate the query input in ASCII text from character images selected either manually or in a semi-supervised way. In this context, the word spotting task is also referred to as *word retrieval* [44,68,86,95,96,141,142].

In the following, we review the main steps of a word spotting system with respect to the extracted features, the representation defined to describe both documents and queries at a specific level and the similarity measures used to compare them. A consise view of some key approaches considered in this work is presented in Table 3. Since most word spotting approaches belong to various distinct categories, we mainly divide them according to the representation used in each case.

4.1. Feature extraction

The appropriate selection of features has a great impact on the performance of a word spotting system as well as of numerous other computer vision applications. Girshick et al. [154] state that progress made on various visual recognition tasks in the last decade relied considerably on the use of SIFT [155] and HoG [156] features. Particularly in word spotting applications, Rodríguez-Serrano and Perronnin evaluate the performance of different feature types using DTW [143] and HMMs [16]. In both cases, the authors show that their proposed local gradient histogram features outperform other profile-based or geometrical features. Other word spotting approaches [57,110,157] also confirm the effect of features on the final performance.

In general, we can distinguish two broad categories of features. *Global* features are extracted from the object of interest which can be either a word image or a document region as a whole. Examples of such features are the width, height, or the aspect ratio of the word image, the number of foreground pixels, moments of background pixels and others. On the contrary, *local* features may be detected independently at different regions of the input image, which may be a text line, word or primitive word parts. For instance, the pixel densities, the position or the number of holes, valleys, dots and crosses at keypoints or regions are local features.

Table 3
Overview of key techniques according to the core steps of the word spotting pipeline.

Publications	Query	Features	Representation	Similarity
[28,36,57,82,122]	QBE	SIFT	BoVW	Cosine, Euclidean
[108]	QBE	SIFT	BoVW	Symmetric KL-divergence
[101]	QBE	SIFT	BoVW	Longest Common Subsequence
[33,55]	QBE	SIFT	BoF-HMM	Viterbi decoding probability
[34]	QBS	SIFT	BoF-HMM	Viterbi decoding probability
[46,140]	QBS	Geometrical	HMM	Viterbi decoding probability
[143]	QBE	Local Gradient Histogram (LGH)	HMM	Viterbi decoding probability
[144]	QBE	Geometrical, pixel counts	SC-HMM, HMM	Viterbi decoding probability
[16]	QBE	Geometrical, pixel counts, LGH	SC-HMM, HMM	Viterbi decoding probability
[38]	QBE	Pixel values	NN internal representation	NN learned similarity
[41]	QBE	Gradient-based (POG)	Fixed-length vector	Euclidean
[145]	QBE	Zoning/NN layer activations	Fixed-length vector	Euclidean
[67]	QBE/QBS	Pixel values/PHOC	Fixed-length vector	Bray-Curtis dissimilarity
[45]	QBE/QBS	Pixel values/PHOC/DCTofW	Fixed-length vector	Euclidean
[42]	QBE/QBS	NN layer activations	Fixed-length vector	Euclidean
[43,65,146]	QBE/QBS	SIFT/PHOC	Fixed-length vector	Euclidean
[47,147]	QBE	SIFT/PHOC	Fixed-length vector	Euclidean
[61]	QBE	HoG, LBP	Fixed-length vector	Euclidean
[29,71,84]	QBE	Gradient/profile-based	Fixed-length vector	Euclidean
[98]	QBE	Shape Context	Fixed-length vector	Euclidean
[89,148]	QBE	Adaptive Zoning	Fixed-length vector	Euclidean
[31]	QBE	Blurred Shape Model	Fixed-length vector	Euclidean
[149]	QBE	Characteristic Loci	Fixed-length vector	Euclidean
[39]	QBS	Gradient-based	Fixed-length vector	Euclidean
[95,96]	QBS	Standard Zoning	Fixed-length vector	Euclidean
[120]	QBS	Gabor (grayscale)	Fixed-length vector	Euclidean
[141]	QBS	Global, Profiles	Fixed-length vector	Dot Product
[87]	QBE	Global, Profiles	Fixed-length vector	Minkowski distance
[92]	QBE	Standard Zoning	Fixed-length vector	Square distance-based
[80]	QBE	Zoning/Profile-based	Fixed-length vector	Correlation-based
[62,150]	QBE	HoG	Fixed-length vector	Cosine Distance
[131]	QBS	Moment-based	Fixed-length vector	Cosine Distance
[30,105]	QBE	Dali, SIFT	Heat Kernel Signature	Euclidean-based
[56]	QBE	Point Distribution Histogram	Variable-length	Histogram Intersection
[35,63,151]	QBE	Profiles, Moments, Gabor	Variable-length	DTW-based
[37,83,152]	QBE	Word profiles	Variable-length	DTW-based
[70,109]	QBE	Global, profile-based	Variable-length	DTW-based
[79]	QBE	Multi Angular Descriptor	Variable-length	DTW-based
[91]	QBE	Adaptive Zoning	Variable-length	DTW-based
[69]	QBE	Slit Style HoG	Variable-length	DTW-based
[102]	QBS	Profiles, Moments, DFT	Variable-length	DTW-based
[153]	QBE	Wavelet coefficients	Variable-length	Earth Movers Distance
[68,121]	QBS	Gradient-based (ZoI)	Variable-length	Cohesive Elastic Matching
[88]	QBS	Column-based	Word Shape Coding	Sequence alignment
[86]	QBS	Column-based	Word Shape Coding	Edit Distance
[103]	QBS	Character shape features	Word Shape Coding	Edit Distance
[115]	QBE	Profile-based	Graph-based	Edit Distance-based
[90]	QBE	Character primitives	Graph-based	Edit Distance-based
[58,132,139]	QBE	Structural, Shape Context	Graph-based	Edit Distance-based
[54]	QBE	Graphemes of convex groups	Graph-based	Edit Distance-based

We should note here that approaches based only on global features are obsolete in the recent literature.

Local features from the other hand are very common and are used either solely or in combination with global features. Local features extracted from raw pixels to directly represent document images were outperformed throughout the years by higher level features. A typical example of higher level features comprise the upper and lower word profiles, the number of foreground pixels and the number of transitions from background to foreground. These column features, also known as word profiles, were popularized by Rath and Manmatha [37,158] and adopted by many other researchers. They are extracted from each column of the word image or the text line and concatenated to variable-length sequences of features which describe text regions (e.g. words) as a whole.

Geometrical column features are also widely used with the sliding window approach in [21,46,140,159,160]. These typically contain three global and six local features. The global features are the moments of the black pixels distribution within the window. The local features are the position of the top-most and that of the bottom-most black pixel, the inclination of the top and bottom contour of the word at the actual window position, the number of vertical black/white transitions and the average gray scale value between the top-most and bottom-most black pixel. These features also form a variable-length sequence of features, usually modelled with HMMs or NNs, which can adapt better to writing style variations.

Zoning features [89,91,95,96] have also been proved quite efficient statistical features which provide high speed and low complexity word matching. They are usually calculated by the density of pixels or other pattern characteristics in the zones that the pattern frame is divided. Their application to printed documents yields satisfactory results which is not always the case for handwritten documents.

Neural network-based models typically use raw pixel intensity information as their input [67,145]. From a theoretical stand-point, using image information with little or no preprocessing is a valid practice in the case of NNs, as intermediate net layer activations can be considered as the image features, dynamically learned dur-

ing network training. Following this rationale a step further, in a number of works the NN is used purely as a feature extractor [42,145]. The image is fed-forward through the NN, and the activations of one or more layers are used to form feature vectors.

Gradient-based features are also widely used as higher level local features. This family of features tends to be superior over the word profiles for multi-writer word spotting since it can also capture the directions of the strokes, which are discriminative for distinguishing different words. Typical examples of this type are the Histograms of Gradients (HoG) [156] as well as the features extracted using the Scale Invariant Feature Transform (SIFT) [155]. Similar to SIFT, HoG computes a histogram of gradient orientations in a certain local region. One of the main differences between SIFT and HoG is that HoG normalizes such histograms in overlapping local blocks and makes a redundant expression. HoG features are computed in a rigid grid while SIFT features are either densely sampled in local patches of the image or extracted from keypoints (e.g. corners). Several variants of HoG and SIFT features have been successfully used for word spotting [69,143,157].

Pattern features are computed by placing primitives in local image regions and analyzing the relative differences. Pattern analysis is quite useful in texture information representations. Examples of this type are the Local Binary Patterns (LBP) [161] and Gabor features [162]. LBP features mainly focus on the gradient information about the local pattern and they can preserve more local information than the features extracted from only one pixel wide column. They are usually combined with gradient-based features to yield a more discriminative representation for word spotting [61]. Gabor features are related with Gabor wavelets for human perception simulation, which are computed by convolving images with Gabor filters. Application of this type of features can be found in [35,120].

Apart from statistical features (e.g. SIFT, HoG), structural features, such as graphemes from connected components, adjacent line segments or graphs arranged into tree structures have also found their way in word spotting. The main motivation behind selecting such features is that the structure of the handwriting is more stable than the pure appearance of its strokes. This is especially important when dealing with the elastic deformations of different handwriting styles. Such structural features may be extracted from the contour [48,58,79] or the skeleton [44,51,54,56–58,78] of an image.

Advanced gradient, structural and concavity (GSC) features [163] are a good choice for Arabic scripts [74–76]. They are multi-resolution features that combine three different attributes of the character shape, the gradient (representing the local orientation of strokes), the structural features (which extend the gradient to longer distances and provide information about stroke trajectories) and the concavity features (which capture stroke relationships at long distances).

Finally, a recently proposed technique introduced the idea of using attributes as features for word spotting [146]. Attributes are semantic properties that can be used to describe images and categories since they can transfer information from different training words and lead to compact signatures. The selection of these attributes is usually a task-dependent process, so for their application to word spotting they are defined as word-discriminative and appearance-independent properties. In a nutshell, they combine visual (features) and textual (labels) information to encode a word image representation which is robust to writing styles, enables both QBE and QBS and is fast to compare.

4.2. Representation

After a set of features has been extracted, a suitable representation of their values has to be defined in order to allow efficient comparison between the query image and the documents at a specific level. *Variable-length* representations describe word images or text lines as a time series, usually using a window that slides over the image in the writing direction. In contrast, *fixed-length* representations extract a single feature vector of fixed size which characterizes the document region as a whole.

Variable-length representations adopt the sequential nature of handwritten words formed by the concatenation of individual characters. Nevertheless, since two words may have different numbers of characters or widths, defining a distance between feature vectors is not straightforward. In this case, a standard practice is to use sequence alignment techniques such as the DTW.

Probabilistic representations are also very popular and have proven to be more effective than variable-length vectors obtained directly from image features. These typically consist of character or word models which represent the sequential features and are trained from annotated data usually based on hidden Markov models [33,46,50,66] as well as neural networks [21,22,83].

Word Shape Coding (WSC) [86,88,103] is also another way to represent sequential features on stroke level. Particularly, each word image is encoded as a sequence of symbols roughly corresponding to characters. In most cases the symbol set has a lower cardinality with respect to the character set in the original language and it is easier to recognize. Each word is represented by a symbol string. Due to the reduced number of symbol classes, one-to-one correspondences between a symbol and a character are uncertain and therefore a symbol string can be mapped to several words.

A growing interest in *graph-based* representations [44,54,58,132,139] is also reported by the research community. Such representations are defined on structural features extracted from connected components or strokes, along with their spatial arrangements. Although structural features are considered language dependent as they capture the specific shapes of the writing symbols of a language, graph-based representations of such features may perform well in terms of speed and accuracy under large variability in writing style.

Fixed-length representations present a clear advantage over sequential representations, as the fixed-size feature vectors can be compared using standard distances such as the Euclidean distance, or any statistical pattern recognition technique. This way image matching is reduced to a much faster nearest neighbor search problem. In some cases, fixed-length descriptions are formed directly from the extracted features without involving some learning step.

There are cases though where variable-length representations are pooled to fixed-length feature vectors using an encoding scheme. In this spirit, many researches from the document analysis community deem the word spotting problem as an object detection task based on matching techniques between features extracted from keypoints. However, the keypoint matching framework presents the same drawbacks as the sequential methods since an alignment between the keypoint sets has to be computed. In order to avoid exhaustively matching all keypoint pairs, the bag-of-features paradigm from the information retrieval field was adopted as the *Bag-of-Visual-Words* (BoVW) [164]. This consists in an holistic and fixed-length image representation while keeping the discriminative power of local features such as SIFT. The BoVW representation relies on the following steps:

1. Keypoints are extracted from the document images at a specific level using an appropriate detector.
2. Keypoints or shape descriptors evaluated upon them, are clustered and similar descriptors are assigned to the same cluster. Each cluster corresponds to a visual word that is a representation of the features shared by the descriptors belonging to that cluster.

3. Each image region is described by a vector containing the occurrences of each visual word in that image.

Instead of using keypoints to build the visual codebook, recent approaches prefer to densely sample features over regular fixed-size grids [82,122] since the larger amount of descriptors extracted from an image, the better the performance of the BoVW model is. Descriptors having a low gradient magnitude are directly discarded. One main drawback of BoVW 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. [165] proposed the Spatial Pyramid Matching (SPM) method which takes into account the visual word distribution over the fixed-size patch by creating a pyramid of spatial bins.

Another way to form fixed-length descriptions from variable-length representations is the *Fisher vector* [166]. Assuming that a set of features such as SIFT are extracted from a dense grid, corresponding for instance to a word image, the next step is to train a Gaussian mixture model (GMM) using SIFT descriptors from all input images of the document collection. Subsequently, Fisher vectors are calculated for each image as a function of their SIFT description and the gradients of the GMM with respect to its parameters. This yields a fixed-length, highly discriminative representation that can be seen as an augmented BoVW description which encodes higher order statistics. Fisher vectors have previously been used with success in various fields of computer vision [167,168].

Relevant examples of pooling features to fixed-length vectors can be found in [79]. The authors employ the Boostmap algorithm described in [169] to embed the feature space of variable-length representations which are matched with DTW into a Euclidean space for faster comparisons. In the same direction, Wei et al. [84] use DFT on variable-length word profiles to create fixed-length vectors.

Finally, of note is the NN-based model proposed in [38]. In this work, a convolutional neural network accepts pairs of word images as inputs and returns a similarity score in the output. Image description is not explicitly expressed as either a variable or fixed-length vector. Hence, there is no image descriptor in the classical sense and images are processed and represented internally throughout the NN layer pipeline.

4.3. Matching process

The matching task is composed of the similarity computation between the feature representations of the query, which may be a feature vector, a graph, or a statistical model and the document image at word, line or page level. The system performance is greatly affected by the suitable selection of the matching technique. Actually, an improper choice of a matching algorithm may lead to lower performance despite the potentially good choices of features and representations for a particular case.

4.3.1. Word to word matching

This family of approaches requires the document images to be segmented at word level and the matching is carried out directly between the representations of the query and each word image. Apart from the query type (template image or string), we can further distinguish *learning-free* from *learning-based* techniques.

Many of the proposed methods follow the learning-free paradigm under the QBE scenario. For instance, Rath and Manmatha [37] compare variable-length sequences of features extracted from word profiles using DTW for word spotting in historical handwritten documents. In the same direction, many variants of DTW-based word spotting methods have been proposed. Adamek et al. [170] employ DTW to align convexity and concavity features extracted from single closed contours for spotting words in historical handwritten documents. In historical printed text, Khurshid et al. [115] propose an approach to initially align features (S-characters) extracted from connected components at character level by DTW and subsequently compare the resulting character prototypes at word level using a segmentation-driven edit distance. Rodríguez-Serrano and Perronnin [143] confirm the superiority of local gradient histogram features over the word profiles for multi-writer handwritten word spotting using DTW. Papandreou et al. [91] propose an adaptive zoning description that can also be matched by DTW for printed documents.

Fixed-length representations are also very common in the QBE learning-free case. Gatos et al. [148] introduce the idea of adaptive zoning features for QBE word spotting in a historical printed document dataset. These features are extracted after adjusting the position of every zone based on local pattern information. The adjustment is performed by moving every zone towards the pattern body according to the maximization of the local pixel density around each zone. In the same dataset, a size-normalization technique along with zoning and profile features to compute the dissimilarity between two word images is proposed in [171]. The distance is based on a combination of a windowed Hausdorff measure and a robust curvature estimation using integral invariants. Another learning-free fixed-length representation which is based on zoning characteristics is proposed in [95] and uses the L_1 distance metric. Moreover, characteristic Loci features [172], which are a particular case of the shape context descriptor, have been used by Fernandez et al. [149]. They are extracted from keypoints, represented by histograms of Locu numbers in a fixed-length vector and compared using the Euclidean distance. In Retsinas et al. [41], projections of image gradients are combined in a Radon transform-like procedure to form fixed-length vectors which are compared using the Euclidean distance.

Aldavert et al. [28] propose an unsupervised QBE method based on the BoVW framework which is enhanced by several improvements recently proposed in computer vision, though not exploited by the document analysis community. Particularly, they encode descriptors using the sparse coding technique proposed in [173], known as Locality-constrained Linear Coding (LLC). To make visual words more discriminative, they add spatial information using a Spatial Pyramid Matching (SPM) [165] as well as a power normalization technique during the pooling process. The query fixed-length vector is then matched with the vectors of the dataset word images using the Euclidean distance.

With respect to the learning-free paradigm, there are also methods which allow textual queries as inputs (QBS). Bhardwaj et al. [131] extract high order geometrical moments from binary word images as global features to form fixed-length feature vectors which are compared using the cosine similarity. A template is used to generate the query word image corresponding to the query text inserted as input by the user. Another type of techniques that falls in this category relies on representations using word shape coding (WSC). Image matching is usually performed among code strings by means of the minimum *Edit Distance* or by some sequence alignment procedure. The Edit Distance between two strings is given by the minimum number of operations needed to transform one string to the other, where the operation is an insertion, a deletion, or a substitution of a single character. For example, Bai et al. [88] extract features such as character ascenders, descenders, deep eastward and westward concavity, holes, i-dot connectors and horizontal-line intersection. These features are represented using word shape coding and the resulting vectors are compared by a sequence alignment technique. The main advantage of WSC approaches is that arbitrary textual queries can be used without involving training on labeled images. However, such approaches have become obsolete over the years as they are language

dependent and feasible mainly in printed documents with already known fonts.

A probabilistic representation for learning-based QBE word spotting in multi-writer text is proposed in [16]. The query and the dataset word images are represented as sequences of feature vectors extracted using a sliding window in the writing direction and they are modeled using statistical models. Particularly, the authors use multiple instances of a potential query for training a HMM. During the matching process, the similarity between the query and a word image is obtained by the posterior probability of the candidate image, given the model. This probability is calculated using either a continuous HMM (c-HMM) or a semi-continuous HMM (sc-HMM) and a GMM as a universal vocabulary for score normalization. Among different types of features, they report the best performance when local gradient histogram features (LGH) are chosen. However, the method is constrained to queries for which at least one instance appears in the training set. This issue is tackled in their extended work [66] for spotting out-of-vocabulary (OOV) words using a sc-HMM. In fact, the model's parameters are estimated on a pool of unsupervised samples which allow the model to adapt online to the query image. Moreover, the similarity computation between two sc-HMMs is simplified to a DTW between their Gaussian mixture weight vectors which reduces the computational cost.

Concerning learning-based methods, recently there has been much interest in using neural networks for keyword spotting. When dealing with word image description and word to word matching, convolutional neural networks (CNN) or similar feed-forward networks that include convolutional layers in their architecture have been used. These networks work typically either by producing in their output a suitable descriptor of the input word image [67], or by using network layer activations to create input word image descriptors [42,145]. Again a typical distance that is used is the Euclidean, with the exception of Sudholt et al. [67] who use the Bray-Curtis dissimilarity. The Bray-Curtis dissimilarity is a metric that has been shown to work well with spatial pyramid representations [174]. In Zhong et al. [38], a neural network that accepts pairs of word images has been proposed. This model directly outputs similarity scores for the input pair. In order to deal with the fact that neural networks require a comparably large training set, the technique of *jittering* or data augmentation has been used to augment training sets. Following this technique, a number of simple affine transformations can be applied on the training word images to create new training images and boost NN performance [42,45,67]. Pretraining on a generic set and then refining the network on a different second set, which typically should be qualitatively closer to the test set, is another standard practice [42,45]. In Sfikas et al. [145], only pretraining with a generic set is performed, skipping refining altogether. They proceed using combinations of intermediate layer activations as features, aiming to capture more abstract textual features in this manner.

It is interesting to notice that there exist learning-based methods which can deal with both types of query formulation (QBE and QBS). In the seminal work of Almazán et al. [43,146], the authors have proposed a model to learn projections from an image space and a text-string space to a common latent subspace using Kernel Common Subspace Regression (CSR). Vectors in the latent subspace correspond to a common fixed-length representation, computable both for word images and text strings. Dense SIFT descriptors are extracted from word images, encoded to Fisher vectors, while their labels are used to create Pyramidal Histogram of Characters (PHOC) descriptors. PHOC encode textual information in the form of a spatial pyramid of character histograms, treating the absence or presence of unigrams and bigrams as text attributes. At test time, dataset word images and the query are projected to a Euclidean common latent subspace and compared using nearest neighbour search. A number of recent works have been inspired by the work of Almazán et al., further extending or adapting the base model [42,45,65,67]. In Sudholt et al. [67], PHOC descriptors are computed using a deep CNN, while in Krishnan et al. [42] a deep CNN is used to create word image descriptions. In Wilkinson et al. [45], a triplet CNN is used, accepting pairs of positive word matches plus a negative. Also, a new text descriptor is proposed, dubbed DCT of Words (DCTofW). The work of Aldavert et al. [27] also combines visual (SIFT) and textual information obtained from character n-gram models to allow example-based or textual queries. Word images are represented by fixed-length vectors and matched using the cosine similarity.

4.3.2. Word to line matching

This family of methods requires the documents to be segmented at text lines. A window slides over the text lines in order to extract column-based features. We can distinguish two main types of approaches.

In the first category, there are learning-free QBE methods that represent the query and the text lines with sequences of feature vectors and word spotting is applied as a subsequence matching task. In this framework, Terasawa and Tanaka [69] extract Slit Style HoG features from the query image and the text lines using a sliding window. These features are a modification of HoG which is based on gradient distribution. Variable length sequences representing the query and the text lines are then matched using a DTW-based technique which uses Continuous Dynamic Programming (CDP). CDP computes similarities between the query sequence and all the possible subsequences of a text line. Similarly, Mondal et al. [63] make use of word profiles and propose a flexible sequence matching technique which is based on DTW and has the ability to find subsequences in a sliding window-oriented approach, permits one-to-many and many-to-one correspondences while at the same time skipping outliers.

The second category is mainly composed of learning-based QBS methods. Therein, representations of features extracted via a sliding window are modeled using statistical models, such as HMMs [46,140,160] and recurrent neural networks [21,159].

For instance, a HMM-based method which learns character models for word spotting in handwritten text is proposed in [140]. Initially, text line images are normalized to reduce variability in writing style. Each text line image is represented by a sequence of feature vectors which is obtained by a sliding window of one pixel width moving from left to right over the image. At each window position, 9 geometrical (Section 4.1) features are extracted. A character HMM has a standard number of states, each emitting observable feature vectors with output probability distributions given by a GMM. Character models are trained offline using labeled text line images. Then, a text line model is created as a sequence of letter models according to the transcription. The probability of this text line model to emit the observed feature vector sequence of the line image is maximized by iteratively adapting the initial output probability distributions and the transition probabilities between states with the Baum–Welch algorithm [175].

A HMM-Filler model which can generate any sequence of characters is created using all trained letter HMMs. For a given text line image which is modeled by the Filler model, the likelihood of the observed feature vector sequence is computed using the Viterbi algorithm [175]. This way the Filler model can be used once to compute offline the Viterbi decoding for all given text line images. In the online phase, a textual query is represented by a keyword model which is build from character and Filler HMMs. A Viterbi score is also computed for this keyword model and a given text line image and the final matching score between the query and the specific text line is a likelihood ratio of the keyword and Filler text line models, normalized by the length of the query word. This

work is improved in [46] by integrating character n-gram language models into the spotting task.

An important drawback of this approach is the large computational cost of the keyword-specific HMM Viterbi decoding process needed to obtain the confidence scores of each word to be spotted. To counter this issue, Toselli et al. [160] propose a technique to compute such confidence scores, directly from character lattices produced during a single Viterbi decoding process using only the Filler model, meaning that no explicit keyword-specific decoding is needed.

Another learning-based QBS method which makes use of the same features, the same representation and employs bidirectional long-short term memory (BLSTM) recurrent NNs is presented in [21]. The input layer contains one node for each of the 9 geometrical features extracted at each position of the sliding window, the hidden layer consists of the long short-term memory (LSTM) cells and the output layer contains one node for each possible character along with a special node to indicate "no character". The output activation of the nodes in the output layer are normalized to form a vector indicating the probability for each letter to occur at a particular position. The output of the network is therefore a matrix of probabilities for each letter and each position. A score is assigned to each path through the matrix by multiplying all probability values along the path. The letters visited along the optimal path (the one with the maximum score) give the spotted letter sequence. To spot a query keyword inside a text line, the character probability sequence is extended by an additional special symbol. By adding this symbol at the beginning and at the end of the keyword, the algorithm finds the best path through the output matrix that passes through the letters of the keyword at their most likely position while the rest of the text line has no influence. The keyword spotting score hence reflects the product of all character probabilities of the optimal subsequence that starts with the space before the first character of the keyword and ends with the space after its last character. This score is also normalized by the length of the query word.

4.3.3. Word to page matching

One of the major issues of the preprocessing stage is that possible segmentation errors are regularly conveyed in the spotting phase. Particularly, accurate word segmentations are difficult to obtain in handwritten and degraded documents. For this reason, several *segmentation-free* word spotting techniques have emerged.

Leydier et al. [68,121] compute local keypoints over a document page in order to detect regions of interest. Gradient features are then extracted from these zones of interest and the query image. The user inserts a textual query which is artificially generated from manually selected character images. The query image feature vector is then matched with that of each zone using an elastic matching method between different pixel-wise gradient matchings. In a similar fashion, Zhang and et al. [30,123] detect regions of interest by computing local keypoints over the document pages. Features based on the Heat Kernel Signature (HKS) [176] are extracted from these regions and used through a costly distance computation in a language independent manner, though not scalable in large datasets.

The most common approach is to use a patch-based framework [33,36,62,82,92] in which a window slides over the whole document. In this framework, perfect segmentations are not expected and elements from surrounding words will appear within a patch. Gatos et al. [92] detect salient text regions on a document page using a RLSA-based smoothing. A block-based extraction of pixel densities is then applied for the query image and the salient regions which are matched using a template matching process satisfying invariance in terms of translation, rotation and scaling. Rusiñol et al. [82] represent document regions with a fixed-length descriptor based on the BoW representation of SIFT features extracted via a sliding window over the whole page. In this case, comparison of regions is much faster since a dot-product or Euclidean distance can be used. In addition, Latent Semantic Indexing (LSI) is used to learn a latent space where the distance between word representations is more meaningful than that in the original space. Rothacker et al. [33] also make use of the BoW to feed a HMM obtaining a robust representation of the query in a patch-based framework. The HMM is trained on-the-fly from the specific query.

However, when following a sliding-window approach there are too many possible targets to consider, depending on the number of scales and the stride length. This leads to an increase in the number of false matches and the computational demands. To this end, Kovalchuk et al. [61] propose the extraction of a set of overlapping candidate targets as groups of connected components that satisfy location and scale constraints to fit a standard-size bounding box. Subsequently, they combine HoG and LBP features to form a fixed-size feature vector representing the query and the candidate target images which are matched using the Euclidean distance.

Almazán et al. [62] use HoG features to describe the query image and the document pages in a fixed grid using a sliding window. In order to speed up the sliding-window approach, both Almazán et al. [62] and Rusiñol et al. [122] make use of the product quantization method [177] to compress the descriptor size. In the same direction, Ghosh et al. [147] perform QBS word spotting by avoiding the costly computation of the attribute-based representation over a sliding-window at query-time, which is previously employed in [43] for segmentation-based word spotting. This is achieved by pre-computing an integral representation of the attributes at the cost of discrimination.

Moreover, Riba et al. [54] employ a graph representation relying on a codebook of graphemes which are extracted from shape convexities upon the vectorial approximation of the skeleton graph. These graphemes are used as stable units of handwriting, along with their spatial relationships. Segmentation-free word spotting is achieved by localizing the query word graph as a subgraph of the entire graph representing the whole document. The image matching is performed using an approximate graph Edit Distance method based on a bipartite-graph matching [178] between the two graphs. This time-consuming graph matching is improved by a graph indexing approach that makes use of binary embeddings during preprocessing.

5. Retrieval enhancement

In this section, we present a number of methods which are used to improve the retrieved results of a word spotting system in terms of incorporating the information of the ranked lists obtained from user queries. This is done either by involving the user to select positive query instances in a supervised process, or in an purely unsupervised manner.

5.1. Supervised relevance feedback

The ranked lists of the images which are most similar to the query usually contain many false positive instances. In order to improve the performance of content-based image retrieval systems, several boosting mechanisms have been proposed over the years. *Relevance feedback* is a common technique of this type of approaches. The idea is to examine the results that are initially returned from a given query and to use information about whether or not those results are relevant. This feedback about relevance allows to provide an enhanced result list in the subsequent iterations. Relevance feedback is also used in more general information retrieval applications such as multimedia retrieval (MMR) [179], aiming to refine the multimedia data representation. The proper

extraction of semantic information from multimedia data sources is a challenging task since these sources include directly perceivable media such as audio, image and video, indirectly perceivable sources such as text, bio-signals as well as not perceivable sources like bio-information, stock prices, etc. Particularly for word spotting, we can distinguish two main families of approaches, namely, *supervised* and *unsupervised* methods.

In the case of supervised relevance feedback, also known as *explicit feedback*, the user provides relevance judgements using either a binary or a graded relevance system. Herein, we can further notice two more categories. On one hand, relevance feedback methods may follow the idea of *query reformulation*. Its goal is to search, given the relevance assessments, a new query point in the vector domain that is closer to the positive instances and farther to the negative ones than the original query point. On the other hand, *re-ranking* methods attempt to reorganize the initial ranked list by means of the relevance judgements, without casting any new query.

The works of Bhardwaj et al. [131] and Cao et al. [114] adopt the query reformulation idea to improve the retrieved results based on the widely-used Rocchio's formula [180]. At each relevance feedback iteration, the Rocchio's algorithm reformulates the query feature vector by adjusting the values of its individual features according to the relevance information. In a similar way, Konidaris et al. [95] and Kesidis et al. [96] propose to include the user in the retrieval phase by selecting positive instances from the initial ranked list obtained from synthetic query strings. Since the initial results are based on an heterogeneous comparison between synthetic keywords and real images, the accuracy might not be adequate. Consequently, the transition from synthetic to real data is made feasible by exploiting relevant judgements and use them to perform new queries thus leading to an increased performance. Of great interest is also the work of Rusiñol et al. [110] where relevance feedback is tested both under the query reformulation scenario and the re-ranking scheme. Particularly, Rocchio's method [180] and a related variant are compared with a relevance score [181] (re-ranking). This score is assigned for each word image of the initial ranked list as the ratio between the nearest relevant and the nearest non-relevant word images for this particular image. These relevance scores are then used to form the final ranked list.

5.2. Unsupervised feedback and re-ranking

The obvious benefits of supervised relevance feedback lie on the fact that the user judgements are assigned for only a small portion of all possible candidate targets of the query image inside the document collection. However, this manual process still remains costly and sometimes, even error-prone, i.e. for historical degraded and cursive documents where the visual information is not distinctive enough. This gives rise to unsupervised methods where it is more preferable to automatically select instances from the retrieved results. *Pseudo-relevance feedback* [182] is a characteristic example of this type of techniques. In this case, the top-N results from the ranked list are considered as relevant. Subsequently, an unsupervised re-ranking scheme is used on these top ranked results in order to select a number of elements from the reordered list. These elements are finally added into the query for *query expansion* to obtain a new improved ranked list. The process repeats iteratively until the desirable performance is reached.

Regarding the unsupervised re-ranking scheme, Almazán et al. [62] apply a second ranking step which considers only the best candidates retrieved by an initial efficient ranking step and uses more discriminative features encoded with the costly Fisher vector representation. Once the results retrieved by the sliding-window search are re-ranked using more informative features, a number of top-ranked window regions are used for query expansion. Then the expanded query set is used as the new positive samples of the query model. Although this set may also contain negative samples the accuracy seems to improve per each iteration. In the same spirit, Ghosh and Valveny [147] use a re-ranking step to compensate for the loss of accuracy accrued from an approximate solution of the powerful attribute-based representation in order to transit from segmentation-based to segmentation free word spotting. In other words, they use the top-N candidates from the ranked list given by the initial ranking obtained with the sliding window search and then re-rank them using the more discriminative original representation. Shekhar and Jawahar [182] follow a similar pseudo-relevance feedback paradigm. Therein, the top-N retrieved results are re-ranked according to a score which integrates information from SIFT descriptors and BoVW representation with spatial information, which was missing on the indexing stage. Concisely, the spatial pyramid is used to calibrate the score of each region of the word independently.

5.3. Data fusion

Pseudo-relevance feedback methods may sometimes result into several ranked lists which need to be combined into a final ranked list. *Data fusion* methods accept two or more ranked lists and merge them into a single ranked list thus providing a better effectiveness than any original ranked list. There are two main categories of data fusion techniques. Methodologies which use the similarity values from each ranked list in order to produce the final ranked list are known as *score-based*, while those which use the ranking information from each list in order to create the final ranking are defined as *rank-based*.

It is interesting to notice that the work of Rusiñol et al. [110] also proposes three different data fusion techniques. The idea is to deal with variability in writing style by casting multiple queries and combine the results. An *early fusion* method combines feature vectors accrued from different queries before the retrieval phase. This is done by averaging the query image descriptors and then normalizing by the L_2-norm. The second method is a *late fusion* score-based technique (CombMAX) which assigns to each word in the collection its maximum score across the different casted queries. The third fusion technique is a rank-based method, called Borda Count [183]. Herein, the top most image on each ranked list gets n votes, where n is the dataset size. Each subsequent rank gets one vote less than the previous rank. The final ranked list is obtained by adding all the votes per image and re-sorting.

Louloudis et al. [89] also make use of three rank-based fusion methods in order to combine multiple lists obtained from different word spotting systems applied to the same query. Particularly, the authors consider the same preprocessing steps and matching algorithms and test two different feature types for the same query. This results into two different ranked lists. The first combination method (Rank Position) takes into account only the rank positions of the corresponding words. The second method is the Borda Count and the third method which seems to outperform the other two is a Minimum Ranking method. Therein, for each retrieved word the minimum rank position on all ranked lists is considered as the distance measure.

Finally, the authors in [84] present five scored-based and three rank-based fusion methods to merge multiple ranked lists obtained from each top-ranked instance on the initial ranking list. Since the similarity scores among separate ranked lists may differ both in range and distribution, they also suggest to normalize these scores using a number of score normalization techniques.

Table 4
List of word spotting methods that use certain databases.

Databases	Methods
IAM	Bhardwaj et al. [131], Kumar et al. [40,74], Frinken et al. [21], Toselli and Vidal [160], Fischer et al. [46], Almazán et al. [43], Wshah et al. [75], Ghosh and Valveny [47,147], Sudholt et. al. [67], Wilkinson et al. [45], Krishnan et al. [42]
GW	Leydier et al. [121], Bhardwaj et al. [131], Rusiñol et al. [82,110,122], Lladós et al. [57], Rodríguez-Serrano and Perronnin [66], Frinken et al. [21], Almazán et al. [43,62,150,157], Liang et al. [44], Aldavert et al. [27,28], Howe [51,52], Dovgalecs et al. [36], Zhang et al. [30], Fischer et al. [46], Rothacker et al. [33,34], Kovalchuk et al. [61], Mondal et al. [63], Zagoris et al. [29], Wang et al. [58], Ghosh and Valveny [47,147] Sudholt et al. [67], Wilkinson et al. [45], Krishnan et al. [42], Zhong et al. [38]
H-KWS 2014 Bentham	Kovalchuk et al. [61], Almazán et al. [146], Howe et al. [51], Leydier et al. [68], Pantke et al. [188], Aldavert et al. [28], Yao et al. [189]
H-KWS 2014 Modern	Kovalchuk et al. [61], Almazán et al. [146], Howe et al. [51], Leydier et al. [68], Pantke et al. [188], Aldavert et al. [28]
KWS-2015 Bentham	Rothacker et al. (PRG) [185], Rusiñol et al. (CVC) [185], Leifert et al. (CITlab) [185], Sfikas et al. [145]

6. Evaluation

The ranked list of results obtained from a word spotting system for a number of different queries is finally used to evaluate its accuracy. In this section, we introduce the databases which are publicly available and most widely used for word spotting. After describing the importance of having a common evaluation scheme for direct comparison between methods, we present the distinct measures used for assessing the performance. Finally, we present and discuss the results achieved by the state of the art in various word spotting applications. With respect to the results, we have two sources of information. The first one contains results from two keyword spotting competitions, namely, H-KWS 2014 [184] and KWS-2015 [185], which were organized in conjunction with the ICFHR 2014 and ICDAR 2015 conferences, respectively. The second source derives from the results reported by the recently published papers.

6.1. Databases

The *IAM*[1] database [186] consists of 1539 pages of modern handwritten English text, written by 657 writers. Pages are segmented and annotated, comprising 13,353 text lines and 115,320 words.

The *George Washington*[2] (GW) database [187] contains 20 pages of historical English text written by George Washington and his associates in 1755. The writing styles present only small variations and it can be considered a single-writer dataset. Pages are segmented and annotated, comprising 656 text lines and 4894 words. This is the most commonly used dataset for comparing different word spotting methods.

The *H-KWS 2014 Bentham* and *H-KWS 2014 Modern* datasets[3] were used in the H-KWS 2014 competition. The first one contains 50 pages from a document collection written by the English philosopher and reformer Jeremy Bentham (1748–1832) and his secretarial staff. It contains significant variability in writing style and font size as well as noise. The second one is composed of 100 modern handwritten document pages written by 25 authors in four different languages (English, French, German and Greek).

The *KWS-2015 Bentham*[4] dataset contains 70 document pages containing 15,419 segmented word images and was used in the KWS-2015 competition.

To alleviate the process of cross-referencing the results among new word spotting methods and the ones considered in this work, Table 4 discriminates the proposed approaches for the aforementioned databases. We note that, to our knowledge, some of the proposed methods applied in the KWS-2015 Bentham dataset are not published yet and therefore we mention the respective groups and refer to the specific contest. As we will show in Section 6.3, apart from the competitions, we mainly focus on the results reported by the methods presenting the best comparison degree, in terms of the employed evaluation protocols and experimental setups.

6.2. Evaluation protocols and measures

Many word spotting methods published in the recent years vary in assumptions and settings on which they depend. More specifically, some studies require the words to be segmented during preprocessing, while others require segmentation at line level or no segmentation at all. In addition, some methods are meant to perform well on a particular language, while others are able to deal with different languages and sometimes even heterogeneous scripts. There are also methods that target only printed text or specific writing styles, whereas others cope with handwriting variability. Moreover, some works rely on substantial prior learning using annotated data, while others are applied on unlabeled sets.

Apart from this wide variety of procedures and targets, there is also a huge discrepancy among methods that follow different evaluation protocols. This lack of homogeneity may lie on the distinct evaluation metrics, the sets of queries used for a specific dataset, the occurrence frequency of different queries, the number of pages or folds used for validation and testing for learning-based methods and others. The notable work of Rusiñol et al. [122] includes a review of the results obtained from various word spotting methods when tested on the English manuscript from the George Washington collection [187]. The inhomogeneity of these results somewhat confirms this discrepancy.

Consequently, one must take seriously into account the aforementioned aspects before evaluating a word spotting method, so as to make it directly comparable to as many approaches as possible. This way the results reported in the related literature will become more beneficial for new publications. Table 5 presents a clear view of the word spotting methods considered in this work with respect to the variable categories which are related to the evaluation issues mentioned above. Concisely, we consider the level at which segmentation is applied during preprocessing (word, line) and use the term “free” for methods that perform no segmentation at all. We then take into account whether annotated data is used for training or not. The variability of the handwriting with

[1] http://www.iam.unibe.ch/fki/databases/iam-handwriting-database.
[2] http://www.iam.unibe.ch/fki/databases/iam-historical-document-database.
[3] http://vc.ee.duth.gr/H-KWS2014/.
[4] http://transcriptorium.eu/~icdar15kws/data.html.

Table 5
Review of word spotting methods according to the employed evaluation procedure.

Publications	Segmentation	Learning	Writing style	Evaluation index
[21,46,72–75,160]	Line	Yes	Multi-writer	MAP
[30,33,61,62,82]	Free	No	Single	MAP
[31,57,58,110,142]	Word	No	Multi-writer	MAP
[29,35,39,51,108]	Word	No	Single	MAP
[88,90,95,96]	Word	No	Printed	Precision/Recall
[54,55,122]	Free	No	Multi-writer	MAP
[18,43,66]	Word	Yes	Multi-writer	MAP
[99,100,152]	Word	Yes	Printed	MAP
[102,115,141]	Word	No	Printed	Precision/Recall, F-measure
[27,52]	Word	Yes	Single	MAP
[36,56]	Free	No	Single	Precision/Recall
[80,149]	Word	No	Multi-writer	Precision/Recall
[48,49]	Line	Yes	Multi-writer	Precision/Recall
[89,91]	Word	No	Printed	Detection rate
[97,111]	Character	Yes	Printed	Precision/Recall
[92]	Free	No	Printed	Precision/Recall, F-measure
[139]	Line	No	Multi-writer	MAP
[78]	Word	Yes	Multi-writer	F-measure
[135]	Word-part	No	Multi-writer	Detection rate
[63]	Line	No	Single	F-measure
[153]	Word	No	Single	Precision/Recall
[70]	Word	No	Single	Precision/Recall, F-measure
[44]	Word	Yes	Single	MAP at rank 10
[79]	Word	Yes	Multi-writer	Precision rate
[182]	Word	No	Printed	MAP
[83]	Word	Yes	Multi-writer	Precision/Recall, F-measure
[71]	Word-part	Yes	Multi-writer	Precision/Recall rates
[87]	Word	No	Printed	Mean Precision/Recall
[120]	Word	Yes	Single	Precision/Recall
[147]	Free	Yes	Multi-writer	MAP
[151]	Line	No	Printed	MAP
[34]	Free	Yes	Single	MAP
[84]	Word	No	Multi-writer	R-Precision
[68]	Free	No	Multi-writer	R-Precision

respect to the number of authors is also taken into consideration (single author or multiple writers) except for printed documents. As we can see in Table 5, there are some distinct evaluation indices in the word spotting literature which are defined as follows. *Precision* is the fraction of retrieved words that are relevant to the query:

$$P = \frac{|\{\text{relevant instances}\} \cap \{\text{retrieved instances}\}|}{|\{\text{retrieved instances}\}|}$$

Recall is the fraction of relevant words that are successfully retrieved:

$$R = \frac{|\{\text{relevant instances}\} \cap \{\text{retrieved instances}\}|}{|\{\text{relevant instances}\}|}$$

The *F-measure* is defined as the harmonic mean of the precision and the recall:

$$F = 2 \cdot \frac{P \cdot R}{P + R}$$

The *R-Precision* index is defined as the Precision at a specific Recall value where $P = R$. In the case that precision should be determined for the k top retrieved words, *P@k* is defined by:

$$P@k = \frac{|\{\text{relevant instances}\} \cap \{\text{k retrieved instances}\}|}{|\{\text{k retrieved instances}\}|}$$

This measure defines how successfully the methods produce relevant results to the first k positions of the ranked list. Finally, the *Average Precision* index (AP) is defined as the average of the precision value obtained after each relevant word is retrieved:

$$AP = \frac{\sum_{k=1}^{n} (P@k \times rel(k))}{|\{\text{relevant instances}\}|}$$

where $rel(k)$ is an indicator function equal to 1 if the word at rank k is relevant and 0 otherwise. The mean value of the Average Precision over all queries used in a word spotting task defines the *Mean Average Precision* (MAP). In Table 5 it is easy to observe that this index is the most dominant, thereby indicating its objectiveness and reliability.

In the case though where non-binary relevance assessments are provided beforehand, the Normalized Discounted Cumulative Gain (NDCG) index can be used in order to handle small variations of the query word that can be found in the datasets. The NDCG measures the performance of a retrieval system based on the graded relevance of the retrieved entities. It varies from 0.0 to 1.0, with 1.0 representing the ideal ranking of the entities. For example, the words “fort” and “Fort” may have a relevance judgement equal to 0.9. It is defined by:

$$nDCG = \frac{DCG}{IDCG}$$

where:

$$DCG = rel_1 + \sum_{i=2}^{n} \frac{rel_i}{\log_2(i+1)}$$

where rel_i is the relevance judgement at position i, and *IDCG* is the ideal *DCG* which is computed from the perfect retrieval result.

Finally, we would like to emphasize some crucial points of the performance evaluation of a word spotting system. As it is previously mentioned, the *relevance criterion* determines which query instances should be considered as retrieved and which of the retrieved as relevant. In the case of segmented words, the relevance criterion is a trivial choice as it states directly whether a retrieved word image is correctly classified as the word being searched for.

Actually, the larger the entity that is searched for occurrences of the query is, the less meaningful the relevance criterion becomes. In other words, when line-based methods are evaluated, this criterion only states if a retrieved line indeed contains the keyword, without any particular information of the relative location

Table 6
Experimental results achieved by the winners of each respective track.

Method	Bentham				Modern			
	P@5	MAP	NDCG (Binary)	NDCG	P@5	MAP	NDCG (Binary)	NDCG
Almazán et al. [146]	0.724	0.513	0.744	0.764	0.706	0.523	0.757	0.757
Kovalchuk et al. [61]	0.609	0.416	0.638	0.56	0.539	0.263	0.483	0.483

inside the line. Therefore, the evaluation measures could overestimate the performance. Not to mention that such a binary relevance assessment would yield a completely biased evaluation in a segmentation-free method where a retrieved word area would be considered as relevant if it just contained an actual instance in the document page. Due to this issue, the relevance criterion for segmentation-free word spotting systems should take into account the location information. A widely used measure in the literature considers the intersection over union (IoU) percentage between the retrieved word area and the ground-truth one. If this overlap ratio exceeds a specific threshold (usually 50%) the retrieved result is deemed as relevant. By these means, the system is evaluated in terms of how accurately the query instance is retrieved. We should also note here that in a segmentation-free method under the QBE scenario, the query itself should be taken into account in the final hit list, since it could be missing from the retrieved regions.

6.3. Evaluation results

Regarding the first handwritten keyword spotting competition [184], an evaluation framework was established for assessing QBE keyword spotting approaches. The competition was divided in two distinct tracks. A segmentation-based track, where the location of word images inside the document pages was provided and a fully segmentation-free track. For each track, 50 document images of the H-KWS 2014 Bentham dataset and 100 document images of the H-KWS 2014 Modern dataset (25 pages per language) were used for testing at the competition, resulting in a total number of 300 document images for both tracks. The query set of each dataset contained word image queries of length greater than 6 letters appearing more than 5 times. The measures employed in the performance evaluation of the submitted word spotting algorithms are the Precision at Top 5 Retrieved words (*P*@5), the Mean Average Precision (MAP) and the Normalized Discounted Cumulative Gain (NDCG) for both binary and non-binary relevance judgements. In the segmentation-free track, an overlap percentage criterion was used to consider a retrieved result as relevant based on three overlap thresholds (0.6, 0.7, 0.8).

Five distinct research groups have participated in the competition with three methods for the segmentation-based track and four methods for the segmentation-free track. However, we present only the results achieved by the winners of each track. For more details about the methods participating and the results obtained in each case we refer the reader to [184]. The winner of the segmentation-based track is the learning-based method of Almazán et al. [146] which relies on the attribute representation of visual and textual features. We should note here that the authors adapt their system to the Bentham and the Modern benchmarks, by training attributes in the George Washington and the IAM datasets, respectively.

The winner of the segmentation-free track is the learning-free method of Kovalchuk et al. [61] based on the fixed-length representation of HOG and LBP features. The results obtained by these methods are presented in Table 6. The third row corresponds to the segmentation-based track, whereas the last row stands for the segmentation-free track. For the segmentation-free track we only present the results obtained on average for all the threshold values of the overlap percentage criterion.

The second handwritten keyword spotting competition [185] was divided into two distinct tracks, namely, a learning-free (TRACK I) and a learning-based (TRACK II) track whereas each track included two optional assignments. A segmentation based assignment at word level and a segmentation-free assignment compose Track I. The training-based track was divided in QBE and QBS methods in a segmentation-free framework. Participants could submit to one or both of assignments, depending on the capabilities or restrictions of their systems. The evaluation set consisted of 70 document images from the KWS-2015 Bentham dataset, containing 15,419 segmented words. The query set consists of 243 keywords of different lengths (6-15 characters). Each of these queries is represented by 6 or less different instances, comprising a total of 1421 query images. All queries occur at least 4 times in the evaluation set.

For each assignment, a baseline system was provided to the participants in order to compare their methods and tune the parameters of their systems, using a validation set of 10 document images, containing 3234 words. The query set for the validation partition included 95 images of 20 different keywords, extracted from the training page images as well. An additional set of 423 document images, manually segmented and transcribed into 11,144 lines, was also handed to the participants competing in Track II as training data. No other training sets were allowed in this track.

Mean average precision (MAP) and *P@k* were used to evaluate the solution of each participant corresponding to a particular assignment of each track. If a participant submitted solutions for both assignments, the MAP scores of each assignment were combined to produce a single ranking for each track. The combination rule was designed in order to favor participants with a flexible system without hampering those with a highly-specialized method. In segmentation-free scenarios, an overlap ratio of 0.7 between the retrieved area and the ground truth one was required to accept a result as a true positive.

Six research groups submitted final solutions to the evaluation system. Four of them participated in Track-I and the other two in Track-II. We will only present the results achieved by the winners of each track. To our knowledge, the proposed methods of the winning systems are not yet published. For this reason, we only mention the respective groups and refer the reader to [185] for more details about the baseline systems, the participant systems as well as the results achieved in each case.

The winner of the learning-free track was the Pattern Recognition Group (PRG - Leonard Rothacker, Sebastian Sudholt, Gernot A. Fink), from TU Dortmund University of Germany and submitted solutions for both assignments. The winner of the learning-based track was the Computational Intelligence Technology Lab. (CITlab - Gundram Leifert, Tobias Strauß, Tobias Grüning, Roger Labahn) from the University of Rostock, Germany who also submitted solutions in both assignments of Track II. Tables 7 and 8 illustrate the results for each case, respectively.

In order to provide further insight of the state-of-the-art performance achieved in word spotting, we present the results reported by the recently published methods in the GW and IAM databases. Although these datasets are widely used, there exists no

Table 7
Results for the winner of Track I.

Assignment	Segm. based		Segm. free	
Group	MAP	P@5	MAP	P@5
PRG	0.4244	0.4605	0.2761	0.3434

Table 8
Results for the winner of Track II.

Assignment	QBS		QBE	
Group	MAP	P@5	MAP	P@5
CITlab	0.8711	0.8737	0.8521	0.8552

standard experimental setup and each work adapts it to the needs of their proposed algorithm. For instance, learning-based methods use cross validation and do not evaluate the method on the same data used to fit their model. This reduces the amount of queries as query words must appear both in train and test folds. However, we choose these specific datasets since the reported results are comparable between various methods, at least at a certain degree. In this context, we review these results in Tables 9 and 10 by distinguishing various methods according to the query formulation, the segmentation level required, the use of learning with labeled data (i.e. number of training, validation and testing folds) and the employed experimental setup (i.e. query list). In each case, the MAP measure is used for performance assessment.

With respect to the GW database, Almazán et al. [43] partition the dataset into two sets at word level containing 75% and 25% of the words. The first set is used to learn the attributes representation and the calibration, as well as for validation purposes, whereas the second set is used for testing purposes. The experiments are repeated four times with different training and testing partitions and the results are averaged. In the QBE case, each word of the test set is used as a query in a leave-one-out style. Moreover, the query image is removed from the test set and queries without relevant occurrences are discarded. This setup is also used by Sudholt et al. [67], Krishnan et al. [42] and Wilkinson et al. [45]. In the QBS case, Almazán et al. [43] use only words that also appear in the training set as queries. This setup is also used by Fischer et al. [46,140] and Frinken et al. [21]. In [46] though, punctuation marks are treated as individual words and they are excluded from the query list. This reduces the number of queries leading to an increased performance. Sudholt et al. [67] use all words appearing more than once in the test set as queries. This is also followed by [42,45].

Rodríguez-Serrano and Perronnin [66] split the dataset uniformly into five folds, one for training, one for validating the parameters and three folds for testing their method. For each run, they compute the MAP of the test queries, using the best validation parameters. This process is repeated for all 20 different combinations of their setup and the results are averaged. Aldavert et al. [28] use as queries all dataset word images which appear at least 10 times and contain three or more characters. The query images themselves are also discarded from the retrieved results during evaluation. Kovalchuk et al. [61] employ the same setup as [28] for word-based word spotting and further perform segmentation-free word spotting. In the segmentation-free case, the query image is included in the retrieved areas when assessing the performance and a retrieved region is considered as relevant if it overlaps more than 50% with the ground truth one.

In the segmentation-free paradigm, Rothacker et al. [33], Almazán et al. [62] and Rusiñol et al. [122] use all word images as queries to retrieve candidate regions inside the document pages of the GW collection. The overlap percentage criterion used in [33,62] is set to 20%. In addition, Almazán et al. [62] also use a 50% overlap criterion in their reported results rendering their work directly comparable with that of Rusiñol et al. [122].

The experimental setup employed in the IAM benchmark is common for most of the reported results. There is an official partition for text line recognition which splits the pages into three different sets. The first one is used for training and contains 6161 lines, the validation set contains 1840 lines and the test set contains 1861 lines. These sets are writer independent, i.e., each writer contributed solely to one of the three sets. Although stop words are excluded from queries, they still appear in the dataset and act as distractors. The IAM dataset also contains a set of lines whose transcription is uncertain. These lines are excluded from training and testing. Only words that appear in the training set are used as queries. Almazán et al. [43] retrieve whole lines that are correct if they contain the query word, so as to compare their approach with Fischer et al. [46,140] and Frinken et al. [21]. To this end, all the words of each line are grouped as a single entity and the distance between a query and a line is defined by the distance between the query and the closest word in the line. We should note here that the results reported by Fischer et al. [140] in Tables 9 and 10 are evaluated in [21] through a common experimental setup which allows direct comparison. Sudholt et al. [67], Krishnan et al. [42] and Wilkinson et al. [45] follow the same protocol as Almazán et al. [43] for training while at query time they use all words appearing more than once in the test set as queries.

6.4. Results discussion

Regarding the results presented in both competitions it is concluded that training-based methods can achieve much higher performance than training-free approaches which mostly rely on knowledge about geometric and structural properties of handwritten images without incorporating information obtained from the respective transcriptions. In that sense, training-based methods are the best choice if training data are available, to build efficient systems in terms of scalability and performance. However, training data obtained from documents written in a particular language, render the system's adaptivity dependent on a language written in a corresponding script. This can be also confirmed by the work of Almazán et al. [146] who perform training on GW and testing on Bentham. Segmentation-free word spotting methods should also be given attention since they still have much room for improvement and they are part of a relatively new and unexplored research topic. Actually, approaches that bypass the segmentation step present a clear advantage in historical document collections where perfect word or line segmentations are hindered by various factors. Therefore, future competitions in this field should focus on such aspects to finally help understanding the relative capabilities and requirements of the different approaches to keyword spotting.

As for the performance achieved by the state-of-the-art methods presented in Tables 9 and 10, we can distinguish the top results per each distinct category for the GW and IAM benchmarks. We particularly consider the segmentation level as the main categorization factor between different approaches. In the GW dataset, the top MAP obtained under the QBE scenario, for word-based spotting using training data is reported by Wilkinson et al. [45] (0.980) superseding the result of the previous state-of-the-art method of Almazán et al. [43] (0.929). Although comparisons with the reference systems are not fully straightforward, the advantage of this method over other methods that do not rely on supervised learning is clear. In the same direction, though under the learning-free paradigm, the results reported by Aldavert et al. [28] (0.765) are quite promising. Among the QBS methods, the work of Almazán et al. [43] (0.939) as well as recent NN-based models (for example, Wiliknson et al. [45], 0.936), all give excel-

Table 9
State-of-the-art performance for the GW database.

Reference	Query	Segmentation	Learning	Setup	MAP
Almazán et al. [43]	QBE	Word	4-fold cross validation: 2 training, 1 validation and 1 testing folds	All words in test set as queries	0.929
Sudholt et al. [67]	QBE	Word	Same as [43]	Same as [43]	0.967
Krishnan et al. [42]	QBE	Word	Same as [43]	Same as [43]	0.944
Wilkinson et al. [45]	QBE	Word	Same as [43]	Same as [43]	0.980
Rodríguez-Serrano and Perronnin [66]	QBE	Word	5-fold cross validation: 1 training, 1 validation and 3 testing folds	All words in training set as queries	0.531
Aldavert et al. [28]	QBE	Word	N/A	All words with $\geq$ 10 occurrences and $\leq$ 3 letters as queries	0.765
Kovalchuk et al. [61]	QBE	Word	N/A	Same as [28]	0.663
Almazán et al. [43]	QBS	Word	4-fold cross validation: 2 training, 1 validation and 1 testing folds	All words in training set appearing in all 4 folds as queries	0.939
Sudholt et al. [67]	QBS	Word	Same as [43]	All words appearing more than once in the test set are used as queries	0.926
Krishnan et al. [42]	QBS	Word	Same as [43] (QBS)	Same as [67] (QBS)	0.928
Wilkinson et al. [45]	QBS	Word	Same as [43] (QBS)	Same as [67] (QBS)	0.936
Fischer et al. [140]	QBS	Line	Same as [43] (QBS)	Same as [43] (QBS)	0.600
Frinken et al. [21]	QBS	Line	Same as [43] (QBS)	Same as [43] (QBS)	0.840
Fischer et al. [46]	QBS	Line	Same as [43] (QBS)	Same as [43] (QBS), excluding punctuation marks from query list	0.738
Kovalchuk et al. [61]	QBE	Free	N/A	Same as [28], 50% overlap	0.501
Rothacker et al. [33]	QBE	Free	N/A	All words as queries, 20% overlap	0.611
Almazán et al. [62]	QBE	Free	N/A	All words as queries, 20% overlap	0.688
Almazán et al. [62]	QBE	Free	N/A	All words as queries, 50% overlap	0.591
Rusiñol et al. [122]	QBE	Free	N/A	All words as queries, 50% overlap	0.613

Table 10
State-of-the-art performance for the IAM database.

Reference	Query	Segmentation	Learning	Setup	MAP
Almazán et al. [43]	QBS	Word	3-fold cross validation: 1 training, 1 validation and 1 testing folds	All words in training set appearing in all 3 folds as queries	0.806
Sudholt et al. [67]	QBS	Word	Same as [43]	All words appearing more than once in the test set are used as queries	0.829
Krishnan et al. [42]	QBS	Word	Same as [43]	Same as [67]	0.915
Wilkinson et al. [45]	QBS	Word	Same as [43]	Same as [67]	0.894
Fischer et al. [140]	QBS	Line	Same as [43]	Same as [43]	0.360
Fischer et al. [46]	QBS	Line	Same as [43]	Same as [43]	0.550
Frinken et al. [21]	QBS	Line	Same as [43]	Same as [43]	0.780

lent results that are numerically very close to one another. Also, they give superior numerical results compared to the line-oriented methods reported. Nonetheless, their approach requires the pages to be segmented at word level during training, which is not the case for the three line-oriented approaches. In the segmentation-free case and under the training-free and QBE paradigm, the best results are reported by Rusiñol et al. [122] (0.613). In the IAM dataset, the top MAP is reported by the NN-based model of Krishnan et al. [42] (0.915) for QBS word spotting. Other NN-based approaches come very close to this figure, confirming again the usefulness of neural networks in word spotting.

There also exists a computational analysis for some of the state-of-the-art methods. More specifically, for the IAM dataset, the QBS method of Almazán et al. [43] requires about 1 s to compare all 5000 queries against all 16,000 dataset words on an 8-core Intel Xeon W3520 at 2.67 GHz with 16Gb of RAM. Actually, it involves only one matrix multiplication to compare all queries using the attributes embedded with Common Subspace Regression (CSR), which is about 0.2 ms per query. This is heavily contrasted with the work of Frinken et al. [21] which needs a few milliseconds to compare a keyword with a single text line. In the segmentation-free framework, the work of Almazán et al. [62] requires less than 15 ms on average to match a query image with a single document page. To our knowledge this complexity does not correspond to their full system. It rather employs the Exemplar Word Spotting system with product quantization (without re-ranking and query expansion) with a lower MAP (0.518) than that reported in Table 9 for the GW dataset. The method of Kovalchuk et al. [61] on the other hand takes 33 ms on average to match a query image with all the 20 pages of the GW collection obtaining 0.501 MAP.

7. Conclusion

In this survey, we presented a comprehensive study on word spotting for indexing documents available all over the world, written in various scripts or fonts. After examining the nature of the documents used by the research community, we described the intermediate steps of a word spotting system, namely, preprocessing, feature extraction, representation and similarity measures which are used to retrieve instances of user inserted queries. Subsequently, we overviewed a number of boosting techniques which enhance the outcome of the image matching step. Evaluation standards applied for the performance assessment of a word spotting system were also examined along with the need for a commonly established protocol to allow straightforward comparison with the state of the art. Finally, we presented the results reported by the state of the art in the most commonly used databases. In that sense, we aimed to fulfill one of the major purposes of this work which is to provide solid background for new researchers who are interested in extending their knowledge in the text understanding area.

In an attempt to compare different word spotting systems, we end up with the following conclusions. The research community is moving towards scalable systems that could effectively deal with the large amount of documents. At the same time, the general objective of a word spotting system is to reduce the user interference as much as possible in terms of preprocessing, parameter tuning and relevance feedback. To this end, learning-based systems which train on adequate annotated data might be more suitable than learning-free methods. Since most learning-based methods allow the user to cast arbitrary text queries without the need for manually picking an example to trigger the search, they might yield a more preferable solution for large scale indexing and retrieval. Generally, a learning-based method achieves higher performance than a learning-free method, especially in documents which present writing style variability and are mainly written in languages of a corresponding script. However, such a method will most likely fail if tested on languages written in a substantially different script without retraining on newly annotated data. Training may also result in overfitting to a particular writing style or font. Recent works are promising in this respect though adaptiveness between completely different scripts is still a goal to be reached. In the case where it is difficult to obtain labeled data, learning-free approaches provide a more practical solution. In that sense, we may say that it always depends on the application field and the available resources.

Acknowledgment

This work has been supported by the OldDocPro project (ID 4717) funded by the GSRT as well as the European Union's H2020 grant READ (Recognition and Enrichment of Archival Documents) (Ref: 674943), https://read.transkribus.eu/.

References

[1] S. Levy, Google's two revolutions, 2004, (Newsweek) http://www.newsweek.com/googles-two-revolutions-123507.
[2] L. Vincent, Google book search: document understanding on a massive scale, in: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), 2, 2007, pp. 819–823.
[3] M. Liwicki, H. Bunke, Combining on-line and off-line systems for handwriting recognition, in: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), 1, 2007, pp. 372–376.
[4] A.L. Bianne-Bernard, F. Menasri, R.H. Mohamad, C. Mokbel, C. Kermorvant, L. Likforman-Sulem, Dynamic and contextual information in HMM modeling for handwritten word recognition, IEEE Trans. Pattern Anal. Mach. Intell. 33 (10) (2011) 2066–2080.
[5] S. Shetty, H. Srinivasan, S. Srihari, Handwritten word recognition using conditional random fields, in: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), 2, 2007, pp. 1098–1102.
[6] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, J. Schmidhuber, A novel connectionist system for unconstrained handwriting recognition, IEEE Trans. Pattern Anal. Mach. Intell. 31 (5) (2009) 855–868.
[7] X. Zhang, C.L. Tan, Unconstrained handwritten word recognition based on trigrams using BLSTM, in: Proceedings of the 22th International Conference on Pattern Recognition (ICPR), 2014, pp. 2914–2919.
[8] A. Ahmad, C. Viard-Gaudin, M. Khalid, Lexicon-based word recognition using support vector machine and hidden Markov model, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 161–165.
[9] S. Prum, M. Visani, J. Ogier, Cursive on-line handwriting word recognition using a bi-character model for large lexicon applications, in: Proceedings of the 12th International Conference on Frontiers for Handwriting Recognition (ICFHR), 2010, pp. 194–199.
[10] S. España-Boquera, M. Castro-Bleda, J. Gorbe-Moya, F. Zamora-Martinez, Improving offline handwritten text recognition with hybrid HMM/ANN models, IEEE Trans. Pattern Anal. Mach. Intell. 33 (4) (2011) 767–779.
[11] J. Rohlicek, W. Russell, S. Roukos, H. Gish, Continuous hidden Markov modeling for speaker-independent word spotting, in: Proceedings of the 14th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 1989, pp. 627–630 vol.1.
[12] S. Khoubyari, J.J. Hull, Keyword location in noisy document images, in: Proceedings of the 2nd Annual Symposium on Document Analysis and Information Retrieval, 1993, pp. 217–231.
[13] F. Chen, L. Wilcox, D. Bloomberg, Word spotting in scanned images using hidden Markov models, in: Proceedings of the 18th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 5, 1993, pp. 1–4.
[14] R. Manmatha, C. Han, E. Riseman, Word spotting: a new approach to indexing handwriting, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 1996, pp. 631–637.
[15] P. Keaton, H. Greenspan, R. Goodman, Keyword spotting for cursive document retrieval, in: Proceedings of the 1st Workshop on Document Image Analysis (DIA), 1997, pp. 74–81.
[16] J.A. Rodríguez-Serrano, F. Perronnin, Handwritten word-spotting using hidden Markov models and universal vocabularies, Pattern Recognit. 42 (9) (2009) 2106–2116.
[17] K. Khurshid, C. Faure, N. Vincent, Fusion of word spotting and spatial information for figure caption retrieval in historical document images, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 1, 2009, pp. 266–270.
[18] H. Cao, A. Bhardwaj, V. Govindaraju, A probabilistic method for keyword retrieval in handwritten document images, J. Pattern Recognit. 42 (12) (2009) 3374–3382.
[19] A. Tarafdar, U. Pal, J.-Y. Ramel, N. Ragot, B. Chaudhuri, Word spotting in Bangla and English graphical documents, in: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR), 2014, pp. 3044–3049.
[20] L. Rothacker, D. Fisseler, G. Muller, F. Weichert, G.A. Fink, Retrieving cuneiform structures in a segmentation-free word spotting framework, in: Proceedings of the 3rd International Workshop on Historical Document Imaging and Processing (HIP), 2015, pp. 129–136.
[21] V. Frinken, A. Fischer, R. Manmatha, H. Bunke, A novel word spotting method based on recurrent neural networks, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2) (2012) 211–224.
[22] V. Frinken, A. Fischer, M. Baumgartner, H. Bunke, Keyword spotting for self–training of BLSTM NN-based handwriting recognition systems, Pattern Recognit. 47 (3) (2014) 1073–1082.
[23] A. Murugappan, B. Ramachandran, P. Dhavachelvan, A survey of keyword spotting techniques for printed document images, Artif. Intell. Rev. 35 (2) (2011) 119–136.
[24] M. Kchaou, S. Kanoun, J. Ogier, Segmentation and word spotting methods for printed and handwritten Arabic texts: a comparative study, in: Proceedings of the 13th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2012, pp. 274–279.
[25] S. Marinai, B. Miotti, G. Soda, Digital libraries and document image retrieval techniques: a survey, in: M. Biba, F. Xhafa (Eds.), Learning Structure and Schemas from Documents, Studies in Computational Intelligence, 375, Springer Berlin Heidelberg, 2011, pp. 181–204.
[26] C. Tan, X. Zhang, L. Li, Image based retrieval and keyword spotting in documents, in: D. Doermann, K. Tombre (Eds.), Handbook of Document Image Processing and Recognition, Springer London, 2014, pp. 805–842.
[27] D. Aldavert, M. Rusiñol, R. Toledo, J. Lladós, Integrating visual and textual cues for query-by-string word spotting, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 511–515.
[28] D. Aldavert, M. Rusiñol, R. Toledo, J. Lladós, A study of bag-of-visual-words representations for handwritten keyword spotting, Int. J. Doc. Anal. Recognit. 18 (3) (2015) 223–234.
[29] K. Zagoris, I. Pratikakis, B. Gatos, Segmentation-based historical handwritten word spotting using document-specific local features, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 9–14.
[30] X. Zhang, C. Tan, Segmentation-free keyword spotting for handwritten documents based on heat kernel signature, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 827–831.

[31] 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 10th Workshop on Historical Document Imaging and Processing, 2011, pp. 83–90.

[32] U. Roy, N. Sankaran, K. Sankar, C. Jawahar, Character n-gram spotting on handwritten documents using weakly-supervised segmentation, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 577–581.

[33] L. Rothacker, M. Rusiñol, G.A. Fink, Bag-of-features HMMs for segmentation-free word spotting in handwritten documents, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 1305–1309.

[34] L. Rothacker, G.A. Fink, Segmentation-free query-by-string word spotting with bag-of-features HMMs, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 661–665.

[35] T. Mondal, N. Ragot, J. Ramel, U. Pal, A fast word retrieval technique based on kernelized locality sensitive hashing, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 1195–1199.

[36] V. Dovgalecs, A. Burnett, P. Tranouez, S. Nicolas, L. Heutte, Spot it! Finding words and patterns in historical documents, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 1039–1043.

[37] T.M. Rath, R. Manmatha, Word spotting for historical documents, Int. J. Doc. Anal. Recognit. 9 (2–4) (2007) 139–152.

[38] Z. Zhong, W. Pan, L. Jin, H. Mouchère, C. Viard-Gaudin, SpottingNet: learning the similarity of word images with convolutional neural network for word spotting in handwritten historical documents, in: Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 295–300.

[39] A.I. Wagan, S. Bres, H. Emptoz, Word spotting in Alice's adventures underground using multi scale integral orientation features, in: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS), 2010, pp. 417–424.

[40] G. Kumar, Z. Shi, S. Setlur, V. Govindaraju, S. Ramachandrula, Keyword spotting framework using dynamic background model, in: Proceedings of the 12th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2012, pp. 582–587.

[41] G. Retsinas, G. Louloudis, N. Stamatopoulos, B. Gatos, Keyword spotting in handwritten documents using projections of oriented gradients, in: Proceedings of the 12th IAPR Workshop on Document Analysis Systems (DAS), 2016, pp. 411–416.

[42] P. Krishnan, K. Dutta, C.V. Jawahar, Deep feature embedding for accurate recognition and retrieval of handwritten text, in: Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 289–294.

[43] J. Almazán, A. Gordo, A. Fornés, E. Valveny, Word spotting and recognition with embedded attributes, IEEE Trans. Pattern Anal. Mach. Intell. 36 (12) (2014) 2552–2566.

[44] Y. Liang, M. Fairhurst, R. Guest, A synthesised word approach to word retrieval in handwritten documents, Pattern Recognit. 45 (12) (2012) 4225–4236.

[45] T. Wilkinson, A. Brun, Semantic and verbatim word spotting using deep neural networks, in: Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 307–312.

[46] A. Fischer, V. Frinken, H. Bunke, C. Suen, Improving HMM-based keyword spotting with character language models, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 506–510.

[47] S.K. Ghosh, E. Valveny, Query by string word spotting based on character bi-gram indexing, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 881–885.

[48] Y. Kessentini, C. Chatelain, T. Paquet, Word spotting and regular expression detection in handwritten documents, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 516–520.

[49] Y. Kessentini, T. Paquet, Keyword spotting in handwritten documents based on a generic text line HMM and a SVM verification, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 41–45.

[50] C. Choisy, Dynamic handwritten keyword spotting based on the NSHP-HMM, in: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), 1, 2007, pp. 242–246.

[51] N.R. Howe, Part-structured inkball models for one-shot handwritten word spotting, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 582–586.

[52] N.R. Howe, Inkball models for character localization and out-of-vocabulary word spotting, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 381–385.

[53] J. Puigcerver, A. Toselli, E. Vidal, Word-graph-based handwriting keyword spotting of out-of-vocabulary queries, in: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR), 2014, pp. 2035–2040.

[54] P. Riba, J. Lladós, A. Fornés, Handwritten word spotting by inexact matching of grapheme graphs, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 781–785.

[55] G. Fink, L. Rothacker, R. Grzeszick, Grouping historical postcards using query-by-example word spotting, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 470–475.

[56] H. Chatbri, P. Kwan, K. Kameyama, An application-independent and segmentation-free approach for spotting queries in document images, in: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR), 2014, pp. 2891–2896.

[57] J. Lladós, M. Rusiñol, A. Fornés, D. Fernandez, A. Dutta, On the influence of word representations for handwritten word spotting in historical documents, Int. J. Pattern Recognit. Artif. Intell. 26 (05) (2012) 1263002.

[58] P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Lladós, A. Fornés, A novel learning-free word spotting approach based on graph representation, in: Proceedings of the 11th IAPR International Workshop on Document Analysis Systems (DAS), 2014, pp. 207–211.

[59] J.P. Van Oosten, L. Schomaker, Separability versus prototypicality in handwritten word retrieval, in: Proceedings of the 13th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2012, pp. 8–13.

[60] T. van Der Zant, L. Schomaker, K. Haak, Handwritten-word spotting using biologically inspired features, IEEE Trans. Pattern Anal. Mach. Intell. 30 (11) (2008) 1945–1957.

[61] A. Kovalchuk, L. Wolf, N. Dershowitz, A simple and fast word spotting method, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 3–8.

[62] J. Almazán, A. Gordo, A. Fornés, E. Valveny, Segmentation-free word spotting with exemplar SVMs, Pattern Recognit. 47 (12) (2014) 3967–3978.

[63] T. Mondal, N. Ragot, J.Y. Ramel, U. Pal, Flexible sequence matching technique: application to word spotting in degraded documents, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 210–215.

[64] T. Mondal, N. Ragot, J.-Y. Ramel, U. Pal, Performance evaluation of DTW and its variants for word spotting in degraded documents, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 1141–1145.

[65] G. Sfikas, A.P. Giotis, G. Louloudis, B. Gatos, Using attributes for word spotting and recognition in polytonic greek documents, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 686–690.

[66] J.A. Rodríguez-Serrano, F. Perronnin, A model-based sequence similarity with application to handwritten word spotting, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2108–2120.

[67] S. Sudholt, G.A. Fink, PHOCNet: a deep convolutional neural network for word spotting in handwritten documents, in: Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 277–282.

[68] Y. Leydier, A. Ouji, F. LeBourgeois, H. Emptoz, Towards an omnilingual word retrieval system for ancient manuscripts, Pattern Recognit. 42 (9) (2009) 2089–2105.

[69] K. Terasawa, Y. Tanaka, Slit style HoG feature for document image word spotting, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 116–120.

[70] A. Abidi, A. Jamil, I. Siddiqi, K. Khurshid, Word spotting based retrieval of Urdu handwritten documents, in: Proceedings of the 13th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2012, pp. 331–336.

[71] M.W. Sagheer, N. Nobile, C.L. He, C.Y. Suen, A novel handwritten Urdu word spotting based on connected components analysis, in: Proceedings of the 20th International Conference on Pattern Recognition (ICPR), 2010, pp. 2013–2016.

[72] M. Khayyat, L. Lam, C.Y. Suen, Learning-based word spotting system for Arabic handwritten documents, Pattern Recognit. 47 (3) (2014) 1021–1030.

[73] N. Li, J. Chen, H. Cao, B. Zhang, P. Natarajan, Applications of recurrent neural network language model in offline handwriting recognition and word spotting, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 134–139.

[74] G. Kumar, V. Govindaraju, A Bayesian approach to script independent multilingual keyword spotting, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 357–362.

[75] S. Wshah, G. Kumar, V. Govindaraju, Statistical script independent word spotting in offline handwritten documents, Pattern Recognit. 47 (3) (2014) 1039–1050.

[76] S.N. Srihari, G.R. Ball, Language independent word spotting in scanned documents, in: Proceedings of the 11th International Conference on Asian Digital Libraries (ICADL), 2008, pp. 134–143.

[77] L. Huang, F. Yin, Q.-H. Chen, C.-L. Liu, Keyword spotting in unconstrained handwritten Chinese documents using contextual word model, Image Vis. Comput. 31 (12) (2013) 958–968.

[78] A. Giotis, D. Gerogiannis, C. Nikou, Word spotting in handwritten text using contour-based models, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 399–404.

[79] R. Saabni, A. Bronstein, Fast keyword searching using 'boostmap' based embedding, in: Proceedings of the 13th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2012, pp. 734–739.

[80] M. Shah, C. Suen, Word spotting in gray scale handwritten Pashto documents, in: Proceedings of the 12th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2010, pp. 136–141.

[81] E.F. Can, P. Duygulu, A line-based representation for matching words in historical manuscripts, Pattern Recognit. Lett. 32 (8) (2011) 1126–1138.
[82] M. Rusiñol, D. Aldavert, R. Toledo, J. Lladós, Browsing heterogeneous document collections by a segmentation-free word spotting method, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 63–67.
[83] H. Wei, G. Gao, Y. Bao, A method for removing inflectional suffixes in word spotting of Mongolian Kanjur, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 88–92.
[84] H. Wei, G. Gao, X. Su, A multiple instances approach to improving keyword spotting on historical Mongolian document images, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 121–125.
[85] V. Ranjan, G. Harit, C. Jawahar, Enhancing word image retrieval in presence of font variations, in: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR), 2014, pp. 2709–2714.
[86] L. Li, S. Lu, C. Tan, A fast keyword-spotting technique, in: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), 1, 2007, pp. 68–72.
[87] K. Zagoris, E. Kavallieratou, N. Papamarkos, A document image retrieval system, Eng. Appl. Artif. Intell. 23 (6) (2010) 872–879.
[88] S. Bai, L. Li, C. Tan, Keyword spotting in document images through word shape coding, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 331–335.
[89] G. Louloudis, A. Kesidis, B. Gatos, Efficient word retrieval using a multiple ranking combination scheme, in: Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (DAS), 2012, pp. 379–383.
[90] P. Roy, J. Ramel, N. Ragot, Word retrieval in historical document using character-primitives, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 678–682.
[91] A. Papandreou, B. Gatos, G. Louloudis, An adaptive zoning technique for efficient word retrieval using dynamic time warping, in: Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage (DATeCH), 2014, pp. 147–152.
[92] B. Gatos, I. Pratikakis, Segmentation-free word spotting in historical printed documents, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 271–275.
[93] J. Sousa, J. Gil, J. Pinto, Word indexing of ancient documents using fuzzy classification, IEEE Trans. Fuzzy Syst. 15 (5) (2007) 852–862.
[94] S. Marinai, Text retrieval from early printed books, Int. J. Doc. Anal. Recognit. 14 (2) (2011) 117–129.
[95] 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, Int. J. Doc. Anal. Recognit. 9 (2–4) (2007) 167–177.
[96] A.L. Kesidis, E. Galiotou, B. Gatos, I. Pratikakis, A word spotting framework for historical machine-printed documents, Int. J. Doc. Anal. Recognit. 14 (2) (2011) 131–144.
[97] Y. Xia, K. Wang, M. Li, Chinese keyword spotting using knowledge-based clustering, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 789–793.
[98] E. Hassan, S. Chaudhury, M. Gopal, Word shape descriptor-based document image indexing: a new DBH-based approach, Int. J. Doc. Anal. Recognit. 16 (3) (2013) 227–246.
[99] P. Krishnan, C. Jawahar, Bringing semantics in word image retrieval, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 733–737.
[100] R. Shekhar, C. Jawahar, Document specific sparse coding for word retrieval, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 643–647.
[101] I.Z. Yalniz, R. Manmatha, An efficient framework for searching text in noisy document images, in: Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (DAS), 2012, pp. 48–52.
[102] M. Meshesha, C.V. Jawahar, Matching word images for content-based retrieval from printed document images, Int. J. Doc. Anal. Recognit. 11 (1) (2008) 29–38.
[103] S. Lu, C.L. Tan, Retrieval of machine-printed Latin documents through word shape coding, Pattern Recognit. 41 (5) (2008) 1799–1809.
[104] E. Indermuhle, V. Frinken, A. Fischer, H. Bunke, Keyword spotting in online handwritten documents containing text and non-text using BLSTM neural networks, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 73–77.
[105] H. Zhang, D.H. Wang, C.L. Liu, Character confidence based on n-best list for keyword spotting in online Chinese handwritten documents, Pattern Recognit. 47 (5) (2014) 1880–1890.
[106] B. Zhu, A. Shivram, S. Setlur, V. Govindaraju, M. Nakagawa, Online handwritten cursive word recognition using segmentation-free MRF in combination with P2DBMN-MQDF, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 349–353.
[107] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man Cybern. 9 (1) (1979) 62–66.
[108] E. Ataer, P. Duygulu, Matching Ottoman words: an image retrieval approach to historical document indexing, in: Proceedings of the 6th ACM International Conference on Image and Video Retrieval (CIVR), 2007, pp. 341–347.
[109] A. Abidi, I. Siddiqi, K. Khurshid, Towards searchable digital Urdu libraries - a word spotting based retrieval approach, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 1344–1348.
[110] M. Rusiñol, J. Lladós, Boosting the handwritten word spotting experience by including the user in the loop, Pattern Recognit. 47 (3) (2014) 1063–1072.
[111] P.P. Roy, U. Pal, J. Lladós, Query driven word retrieval in graphical documents, in: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS), 2010, pp. 191–198.
[112] J. Sauvola, M. Pietikainen, Adaptive document image binarization, Pattern Recognit. 33 (2000) 225–236.
[113] R.F. Moghaddam, M. Cheriet, Application of multi-level classifiers and clustering for automatic word spotting in historical document images, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 511–515.
[114] H. Cao, V. Govindaraju, A. Bhardwaj, Unconstrained handwritten document retrieval, Int. J. Doc. Anal. Recognit. 14 (2) (2011) 145–157.
[115] K. Khurshid, C. Faure, N. Vincent, Word spotting in historical printed documents using shape and sequence comparisons, Pattern Recognit. 45 (7) (2012) 2598–2609.
[116] Z. Shi, V. Govindaraju, Historical document image enhancement using background light intensity normalization, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004, pp. 473–476.
[117] H. Cao, V. Govindaraju, Handwritten carbon form preprocessing based on Markov random field, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 1–7.
[118] B. Gatos, I. Pratikakis, S. Perantonis, Adaptive degraded document image binarization, Pattern Recognit. 39 (3) (2006) 317–327.
[119] N. Howe, A Laplacian energy for document binarization, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 6–10.
[120] S. Cao, V. Govindaraju, Template-free word spotting in low-quality manuscripts, in: Proceedings of the 6th International Conference on Advances in Pattern Recognition (ICAPR), 2007, pp. 45–53.
[121] Y. Leydier, F.L. Bourgeois, H. Emptoz, Text search for medieval manuscript images, Pattern Recognit. 40 (12) (2007) 3552–3567.
[122] M. Rusiñol, D. Aldavert, R. Toledo, J. Lladós, Efficient segmentation-free keyword spotting in historical document collections, Pattern Recognit. 48 (2) (2015) 545–555.
[123] X. Zhang, C.L. Tan, Handwritten word image matching based on heat kernel signature, Pattern Recognit. 48 (11) (2015) 3346–3356.
[124] V. Papavassiliou, T. Stafylakis, V. Katsouros, G. Carayannis, Handwritten document image segmentation into text lines and words, Pattern Recognit. 43 (1) (2010) 369–377.
[125] N. Nikolaou, M. Makridis, B. Gatos, N. Stamatopoulos, N. Papamarkos, Segmentation of historical machine-printed documents using adaptive run length smoothing and skeleton segmentation paths, Image Vis. Comput. 28 (4) (2010) 590–604.
[126] M. Feldbach, K. Tönnies, Robust Line Detection in Historical Church Registers, in: Proceedings of the 23rd DAGM Symposium on Pattern Recognition, Springer, Berlin Heidelberg, 2001, pp. 140–147.
[127] G. Louloudis, B. Gatos, I. Pratikakis, C. Halatsis, Text line and word segmentation of handwritten documents, Pattern Recognit. 42 (12) (2009) 3169–3183.
[128] G. Seni, E. Cohen, External word segmentation of off-line handwritten text lines, Pattern Recognit. 27 (1) (1994) 41–52.
[129] T. Varga, H. Bunke, Tree structure for word extraction from handwritten text lines, in: Proceedings of the 8th International Conference on Document Analysis and Recognition (ICDAR), 2005, pp. 352–356.
[130] S. Banerjee, G. Harit, S. Chaudhury, Word image based latent semantic indexing for conceptual querying in document image databases, in: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), 2, 2007, pp. 1208–1212.
[131] A. Bhardwaj, D. Jose, V. Govindaraju, Script independent word spotting in multilingual documents, in: Proceedings of the 2nd Workshop on Cross Lingual Information Access (CLIA), 2008, pp. 48–54.
[132] P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Lladós, A. Fornés, A coarse-to-fine word spotting approach for historical handwritten documents based on graph embedding and graph edit distance, in: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR), 2014, pp. 3074–3079.
[133] Z. Shi, S. Setlur, V. Govindaraju, A steerable directional local profile technique for extraction of handwritten Arabic text lines, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 176–180.
[134] F.M. Wahl, K.Y. Wong, R.G. Casey, Block segmentation and text extraction in mixed text/image documents, Comput. Graphics Image Process. 20 (4) (1982) 375–390.
[135] M. Kassis, J. El-Sana, Word spotting using radial descriptor, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 387–392.
[136] M. Khayyat, L. Lam, C.Y. Suen, F. Yin, C.L. Liu, Arabic handwritten text line extraction by applying an adaptive mask to morphological dilation, in: Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (DAS), 2012, pp. 100–104.
[137] F. Yin, C.-L. Liu, Handwritten Chinese text line segmentation by clustering with distance metric learning, Pattern Recognit. 42 (12) (2009) 3146–3157.

[138] C.-L. Liu, M. Koga, H. Fujisawa, Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading, IEEE Trans. Pattern Anal. Mach. Intell. 24 (11) (2002) 1425–1437.
[139] P. Wang, V. Eglin, C. Garcia, C. Largeron, A. McKenna, A comprehensive representation model for handwriting dedicated to word spotting, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 450–454.
[140] A. Fischer, A. Keller, V. Frinken, H. Bunke, HMM-based word spotting in handwritten documents using subword models, in: Proceedings of the 20th International Conference on Pattern Recognition (ICPR), 2010, pp. 3416–3419.
[141] A. Kumar, C. Jawahar, R. Manmatha, Efficient search in document image collections, in: Proceedings of the 9th Asian Conference on Computer Vision (ACCV), 4843, 2007, pp. 586–595.
[142] J.A. Rodríguez-Serrano, F. Perronnin, Synthesizing queries for handwritten word image retrieval, Pattern Recognit. 45 (9) (2012) 3270–3276.
[143] J.A. Rodríguez-Serrano, F. Perronnin, Local gradient histogram features for word spotting in unconstrained handwritten documents, in: Proceedings of the 11th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2008.
[144] J.A. Rodríguez-Serrano, F. Perronnin, Score normalization for HMM-based word spotting using a universal background model, in: Proceedings of the 11th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2008, pp. 5–8.
[145] G. Sfikas, G. Retsinas, B. Gatos, Zoning aggregated hypercolumns for keyword spotting, in: Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 283–288.
[146] J. Almazán, A. Gordo, A. Fornés, E. Valveny, Handwritten word spotting with corrected attributes, in: Proceedings of the 14th IEEE International Conference on Computer Vision (ICCV), 2013, pp. 1017–1024.
[147] S. Ghosh, E. Valveny, A sliding window framework for word spotting based on word attributes, in: Proceedings of the 7th Iberian Conference on Pattern Recognition and Image Analysis (PRAI), 9117, 2015, pp. 652–661.
[148] B. Gatos, A. Kesidis, A. Papandreou, Adaptive zoning features for character and word recognition, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 1160–1164.
[149] D. Fernández, J. Lladós, A. Fornés, Handwritten word spotting in old manuscript images using a pseudo-structural descriptor organized in a hash structure, in: Proceedings of the 5th Iberian Conference on Pattern Recognition and Image Analysis (PRIA), 6669, 2011, pp. 628–635.
[150] J. Almazán, A. Gordo, A. Fornés, E. Valveny, Efficient exemplar word spotting, in: Proceedings of the 23rd British Machine Vision Conference (BMVC), 2012, pp. 67.1–67.11.
[151] T. Mondal, N. Ragot, J.-Y. Ramel, U. Pal, Exemplary sequence cardinality: an effective application for word spotting, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 1146–1150.
[152] R. Jain, C.V. Jawahar, Towards more effective distance functions for word image matching, in: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS), 2010, pp. 363–370.
[153] R. Saabni, Efficient word image retrieval using earth movers distance embedded to wavelets coefficients domain, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 314–318.
[154] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 580–587.
[155] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[156] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: C. Schmid, S. Soatto, C. Tomasi (Eds.), Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2, 2005, pp. 886–893.
[157] J. Almazán, A. Fornés, E. Valveny, Deformable HOG-based shape descriptor, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 1022–1026.
[158] T.M. Rath, R. Manmatha, Word image matching using dynamic time warping, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2, 2003, pp. 521–527.
[159] V. Frinken, A. Fischer, H. Bunke, R. Manmatha, Adapting BLSTM neural network based keyword spotting trained on modern data to historical documents, in: Proceedings of the 12th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE Computer Society, Washington, DC, USA, 2010, pp. 352–357.
[160] A. Toselli, E. Vidal, Fast HMM-Filler approach for key word spotting in handwritten documents, in: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 501–505.
[161] T. Ahonen, A. Hadid, M. Pietikainen, Face description with local binary patterns: application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 28 (12) (2006) 2037–2041.
[162] A.K. Jain, N.K. Ratha, S. Lakshmanan, Object detection using Gabor filters, Pattern Recognit. 30 (2) (1997) 295–309.
[163] J.T. Favata, G. Srikantan, A multiple feature/resolution approach to handprinted digit and character recognition, Int. J. Imaging Syst. Technol. 7 (4) (1996) 304–311.
[164] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: Workshop on Statistical Learning in Computer Vision, 2004, pp. 1–22.
[165] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2006, pp. 2169–2178.
[166] F. Perronnin, J. Sanchez, T. Mensink, Improving the Fisher kernel for large-scale image classification, in: Proceedings of the 11th European Conference on Computer Vision (ECCV), 6314, Springer-Verlag, 2010, pp. 143–156.
[167] J. Sanchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the Fisher vector: theory and practice, Int. J. Comput. Vis. 105 (3) (2013) 222–245.
[168] K. Simonyan, O.M. Parkhi, A. Vedaldi, A. Zisserman, Fisher vector faces in the wild, in: Proceedings of the 24th British Machine Vision Conference (BMVC), 1, 2013, p. 7.
[169] V. Athitsos, J. Alon, S. Sclaroff, G. Kollios, Boostmap: a method for efficient approximate similarity rankings, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2004, pp. 268–275.
[170] T. Adamek, N.E. O'Connor, A.F. Smeaton, Word matching using single closed contours for indexing handwritten historical documents, Int. J. Doc. Anal. Recognit. 9 (2) (2007) 153–165.
[171] S. Colutto, B. Gatos, Efficient word recognition using a pixel-based dissimilarity measure, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR), 2011, pp. 1110–1114.
[172] H.A. Glucksman, Classification of mixed-font alphabets by characteristic loci, in: Proceedings of the IEEE Computer Society Conference, 1967, p. 138141.
[173] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 3360–3367.
[174] S. Sudholt, G.A. Fink, A modified isomap approach to manifold learning in word spotting, in: German Conference on Pattern Recognition, 2015, pp. 529–539.
[175] L.R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE Comput. Soc. 77 (2) (1989) 257–286.
[176] F. Moreno-Noguer, Deformation and illumination invariant feature point descriptor, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 1593–1600.
[177] H. Jegou, M. Douze, C. Schmid, Product quantization for nearest neighbor search, IEEE Trans. Pattern Anal. Mach. Intell. 33 (1) (2011) 117–128.
[178] K. Riesen, H. Bunke, Approximate graph edit distance computation by means of bipartite graph matching, Image Vis. Comput. 27 (7) (2009) 950–959.
[179] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, Y. Pan, A multimedia retrieval framework based on semi-supervised ranking and relevance feedback, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 723–742.
[180] B. He, Rocchios formula, Encyclopedia of Database Systems, Springer US, 2009. 2447–2447
[181] G. Giacinto, F. Roli, Instance-based relevance feedback in image retrieval using dissimilarity spaces, in: Case-Based Reasoning on Images and Signals, in: Studies in Computational Intelligence, 73, Springer Berlin Heidelberg, 2008, pp. 419–436.
[182] R. Shekhar, C. Jawahar, Word image retrieval using bag of visual words, in: Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (DAS), 2012, pp. 297–301.
[183] R. Nuray, F. Can, Automatic ranking of information retrieval systems using data fusion, Inf. Process. Manage. 42 (3) (2006) 595–614.
[184] I. Pratikakis, K. Zagoris, B. Gatos, G. Louloudis, N. Stamatopoulos, ICFHR 2014 competition on handwritten keyword spotting (H-KWS 2014), in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 814–819.
[185] J. Puigcerver, A. Toselli, E. Vidal, ICDAR2015 competition on keyword spotting for handwritten documents, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 1176–1180.
[186] U.V. Marti, H. Bunke, The IAM-database: an English sentence database for offline handwriting recognition, Int. J. Doc. Anal. Recognit. 5 (1) (2002) 39–46.
[187] V. Lavrenko, T.M. Rath, R. Manmatha, Holistic word recognition for handwritten historical documents, in: Proceedings of the 1st International Workshop on Document Image Analysis for Libraries, 2004, pp. 278–287.
[188] W. Pantke, M. Dennhardt, D. Fecker, V. Märgner, T. Fingscheidt, An historical handwritten Arabic dataset for segmentation-free word spotting - HADARA80P, in: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICHFR), 2014, pp. 15–20.
[189] S. Yao, Y. Wen, Y. Lu, HoG based two-directional dynamic time warping for handwritten word spotting, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 161–165.

Angelos P. Giotis Received his B.Sc. and M.Sc. degrees in Computer Science from the Department of Computer Science and Engineering, University of Ioannina, Greece in 2010 and 2012, respectively. He is a Ph.D. student at the same department. He is currently working as a Research Associate at the Institute of Informatics and Telecommunications of the National Center for Scientific Research "Demokritos" in Athens, Greece. His research interests lie on Text Understanding, Information Retrieval and Object Detection.

Giorgos Sfikas Received his B.Sc. and M.Sc. degrees in Computer Science from the Department of Computer Science, University of Ioannina, Greece in 2004 and 2007, respectively, and his Ph.D. degree in Image Processing and Computer Vision from the University of Strasbourg, France in 2012. His research interests include statistical image processing, medical imaging, document image processing, machine learning and computer vision. He is currently working as a Research Associate at the Institute of Informatics and Telecommunications of the National Center for Scientific Research "Demokritos" in Athens, Greece.

Basilis Gatos Received his Electrical Engineering Diploma in 1992 and his Ph.D. degree in 1998, both from the Electrical and Computer Engineering Department of Democritus University of Thrace, Xanthi, Greece. He worked as Director of the Research Division in the field of digital preservation of old newspapers at Lambrakis Press Archives and as Managing Director of R&D Division in the field of document management and recognition at BSI S.A. in Greece. He is currently working as a Researcher at the Institute of Informatics and Telecommunications of the National Center for Scientific Research "Demokritos" in Athens, Greece. His main research interests are in Image Processing and Document Image Analysis, OCR and Pattern Recognition. He has more than 150 publications in journals and international conference proceedings and has participated in several research programs funded by the European community. He is a member of the Editorial Board of the International Journal on Document Analysis and Recognition (IJDAR) and program committee member of several international Conferences and Workshops. He is co-organizer of the International Conference of Frontiers in Handwriting Recognition (ICFHR) in 2014 and of the International Workshop on Document Analysis Systems (DAS 2016).

Christophoros Nikou Received the Diploma in electrical engineering from the Aristotle University of Thessaloniki, Greece, in 1994 and the DEA and Ph.D. degrees in image processing and computer vision from Louis Pasteur University, Strasbourg, France, in 1995 and 1999, respectively. He was a Senior Researcher with the Department of Informatics, Aristotle University of Thessaloniki in 2001. From 2002 to 2004, he was a Research Engineer and Project Manager with Compucon S.A., Thessaloniki, Greece. He was a Lecturer (2004–2009) and an Assistant Professor (2009–2013) with the Department of Computer Science and Engineering, University of Ioannina, Ioannina, Greece, where he has been an Associate Professor, since 2013. During the academic year 2015-2016 he has been a visiting Associate Professor at the Department of Computer Science, Univiresity of Houston, USA. His research interests mainly include image processing and analysis, computer vision and pattern recognition and their application to medical imaging. He is a member of EURASIP and an IEEE Senior Member.