Word Spotting in Handwritten Historical Documents by N-gram Retrieval

Giuseppe De Gregorio[1,*], Angelo Marcelli[1]

[1]*Department of Information and Electrical Engineering and Applied Mathematics - DIEM, University of Salerno, Via Giovanni Paolo II 132, Fisciano (SA), 84084, Italy*

Abstract

We address the problem of handwritten word retrieval in documents belonging to small collections of historical interest. Word retrieval consists of finding the images of a given query word, and one of the techniques commonly used to solve this problem is Keyword Spotting (KWS), which promises to retrieve images of words without requiring explicit handwriting recognition. KWS systems, however, are limited by the problem of so called out-of-vocabulary (OOV) words, i.e. words not included in the training set, that cannot be retrieved. To overcome this limitation, we propose a KWS system that focuses the search on character sequences, referred to as N-grams, instead of whole words, thus aiming to make OOV words searchable. The system is based on a Siamese Network that searches for all the N-gram images of the training set that corresponds to the N-gram of the query word and outputs a ranked list of possible images of the searched word extracted from the collection. The results show that the system is able to retrieve words indiscriminately from the set of In-Vocabulary and Out-Of-Vocabulary words, showing similar performance in both cases, suggesting that focusing the search on N-grams may provide a valid solution to the OOV word search problem.

Keywords

Keyword Spotting, Word Retrieval, Historical Document, Handwritten

1. Introduction

The digitization of paper documents has shown several advantages, and for this reason, it constantly captures the attention of researchers. Digitising early written and printed documents can make it possible to preserve a digital version even if the original document is destroyed or damaged. Moreover, typically historical documents are stored in libraries and archives and this may limit to access them, so digitisation makes it easier for researchers to access archival collections by publishing images of the collections online and enabling new ways of access and interaction [1]. The digital format also allows the application of data mining, information retrieval and document analysis techniques to handwritten paper documents using modern tools based on computer vision, document analysis and machine learning [2].

In this paper, we focus on handwritten word retrieval, which consists of finding images of a given query word from a collection of documents. We put our attention on small collections,

1st Italian Workshop on Artificial Intelligence for Cultural Heritage (AI4CH22), co-located with the 21st International Conference of the Italian Association for Artificial Intelligence (AIxIA 2022). 28 November 2022, Udine, Italy.

*Corresponding author.

gdegregorio@unisa.it (G. De Gregorio); amarcelli@unisa.it (A. Marcelli)

0000-0002-8195-4118 (G. De Gregorio); 0000-0002-2019-2826 (A. Marcelli)

comprising 50 pages or less of handwritten documents of historical interest, as they present typical and unique features. Collections with these characteristics can make word retrieval demanding, as handwriting recognition techniques can produce unsatisfactory results. To get around the problem, the KeyWord Spotting (KWS) technique promises to retrieve words without the need for explicit recognition [3]. KWS based on a Query by Example (QbE) paradigm requires the construction of a reference dictionary consisting of transcription/image pairs of words that can be searched. This effectively limits the search to the words in the dictionary and creates the problem of Out of Vocabulary (OOV) words for which no search can be performed. One solution followed by OCR-oriented systems is to concentrate the search on individual characters and to assemble the searched words only at a later stage. It is indeed possible to create an almost complete dictionary of examples of characters, even starting from a few reference pages. However, applying this idea to cursive writing is difficult. The main issue is that segmentation at the character level of the handwritten script is a hard-to-solve problem because of the spatial and stylistic variability typical of cursive handwriting. On the other hand, segmentation into sequences of a few characters, which we will henceforth call N-grams, is certainly a simpler process than character segmentation, and the dictionary of N-grams that can be derived from a few pages of reference would cover a larger percentage of the text present in the whole collection compared to the word-based dictionary, thus reducing the limit of OOV word recovery.

Focusing our attention on short sequences rather than whole words or individual characters can rediscover a motivation related to the fine motor skills acquired by an individual during the learning phase of writing. Studies on motor behaviour have shown that writing is the result of precise motor actions that can be automated [4, 5]. In general, during the learning phase, an individual tends to develop motor programs associated with motor actions with a high frequency of execution. This leads to the development of motor primitives associated with motor actions that are frequently performed and to the definition of motor programs that encode the execution of the movement itself and that are stored in the brain and activated each time that movement is performed. In the field of cursive handwriting, it is plausible to assume that the motor programs are developed not in terms of single characters or whole words, the first being too short and the second too long and complex, but in sequences of a few characters. This would mean that every time a subject writes an N-gram to which a motor program is assigned, he produces an ink trace that is always compatible with and similar to all the others. The repeated similarity in the execution of the same movements for the N-grams could make the N-grams recognisable, thus making them ideal candidates for the recognition of cursive handwriting. With this work, we propose a KWS model capable of recognising words in a small collection of handwritten documents, using N-grams extracted from a subset of the collection as recognition primitives. The KWS follows the Query by String (QbS) paradigm, as it receives as input the string of characters representing the query word, but the core of the system is an N-grams spotting system based on the Query by Example paradigm.

Below, a brief overview of the state of the art is presented in the section 2, the method is then presented in the section 3 and the experimental results are presented in the section 4- Finally the section 5 presents the conclusions.

2. State of the Art

Recognition-free retrieval, also known in the literature as word spotting or keyword spotting (KWS), was developed as an alternative to recognition-based information retrieval. Its purpose is to find all instances of a query in a set by evaluating the similarity between elements and returning as output the results that appear most similar with respect to a common representation [3]. KWS was first applied to the field of handwritten documents of historical interest by Manmatha et al. in [6]. Since then, various solutions have been proposed and different paradigms have been defined. The first important distinction for KWS systems is defined by the query mode of the system. The system can be queried by providing an image of a word to search for similar words in an entire document. This approach is called Query by Example (QbE) [7, 8]. Alternatively, it is possible to query the system with a text query and expect images that contain the word, in which case called Query by String (QbS) [9] approaches. Another important classification distinguishes between segmentation-based and segmentation-free approaches. The former start from the word segmentation of the data collection [6], and the latter [8] apply directly to the entire page or lines of the document. These approaches are less used, although they avoid the problems of poor segmentation. An important distinction concerns lexical-based and lexical-free approaches. The difference lies in the presence of a reference lexicon that collects all the words that the system can recognise and thus recover. Lexical-based approaches prove to be more effective than lexicon-free methods [10, 11]. The main limitation of lexicon-based systems is the problem of OOV (Out Of Vocabulary) words, i.e. words for which the system knows no representation and which therefore cannot be recovered.

Some solutions have been presented in the literature to alleviate the problem of OOV. A naive solution to the OOV problem is to expand the reference dictionary. However, expanding the dictionary involves a loss of execution time. Rabiner et al. [12] try to solve the problem by introducing different systems with small and complementary dictionaries. However, the application of solutions aimed at increasing the size of the lexicon is only feasible if it is possible to expand it. If the initial data of the dictionary is limited, such approaches are not feasible. Puigcerver et al. [10] propose a solution that does not involve expanding the dictionary by introducing a similarity metric between words that can also be applied to OOV. Brakensiek et al. in [13] present a recognition system that uses Markov models together with language models based on N-grams. In all these solutions, the performance of OOVs improves slightly but remains highly dependent on the size of the dictionary and the ability to train a language model. For these reasons, we believe that the problem of OOV words remains an open problem.

3. Method

3.1. Overview

The general idea behind the system is to search for a query word within the image of a document page by decomposing the word into the N-grams for which the system can perform a search. Figure 1 shows the general system workflow. Once the set of query N-grams has been determined, the N-gram spotting phase allows the identification of the positions of the different N-grams within the document, which are finally analysed to determine the position of the entire query

word. In the next sections we will describe in detail the different phases of the process workflow, starting with the deconstruction of the query word and the definition of the set of query N-grams, then the analysis of the N-gram spotting phase, and finally the process of combining the identified N-grams to recover the original query word.

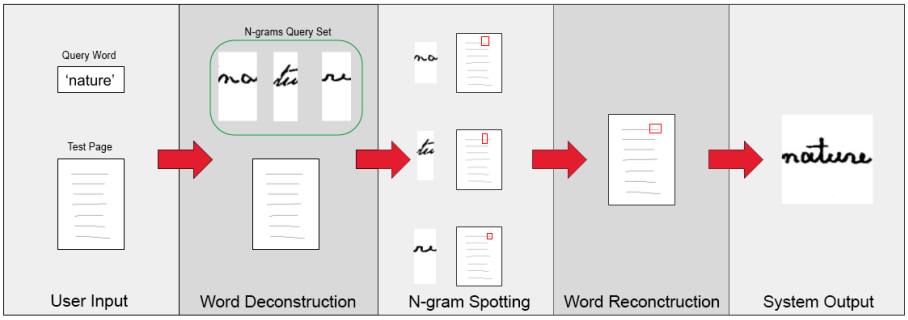

Figure 1: System overview: the system receives the query word "nature" and a page on which to search. The query word is decomposed into the query N-grams that are searched for. At the end of the process, the N-grams found are combined to provide the position of the word within the document page.

3.2. Word Deconstruction

The entire system receives as input an image of a page of a document and a string of the word to search. Since the examination scheme is to search for the N-grams and not the whole word, it is crucial to define the *N-grams query set*. This set is composed of all the N-grams of the query word, defining the maximum depth N, where N represents the maximum number of characters to be considered. When the next step of searching for the N-grams is performed with a system based on a dictionary, it is necessary to purge the set of N-grams of the query of all possible N-grams that do not fit into the reference dictionary.

3.3. N-gram Spotting

The problem reduces to searching the image of the page for the areas containing the N-grams of the query set, performing the N-gram spotting operation. For this purpose, we have defined an N-gram spotting system based on the QbE paradigm. The system, therefore, requires the definition of a reference dictionary consisting of images of N-grams to be searched. The search thus consists of identifying the areas of the image that appear most similar to the images of the N-grams of the reference dictionary. For this to be possible, a measure of similarity between the images must be defined. The measure of image similarity can be learned using neural networks structured according to the paradigm of Siamese architectures. The network allows obtaining a measure of similarity between at least two inputs, which in our case may consist of two N-grams images[14]. The architecture of one branch of the network is kept quite simple and consists of a convolutional backbone for extracting features from the image, followed by a fully connected layer for refining the encoding of the final embedding. Figure 2 shows the network architecture in the top right; images of two N-grams are fed into the network and a measure of the similarity

between them is provided, which is smaller the more similar the input images are to each other. The two arms of the network share the weights, as per the Siamese network paradigm. The convolutional network used to extract the features from the images is a PHOCNet network that has shown good performance in identifying handwritten words [15].

The system then assumes that the document page is segmented into lines of text. The search process is performed on the text lines by sliding a window that crops out part of the image to be compared with one of the N-grams of the query set, and a similarity score between the two images is provided by the network. In the end, the scoring trend of the similarity on the entire line is computed, as shown in Figure 2. Analyzing the trend, the minimum peaks should correspond to an instance of the N-gram we are looking for. The N-gram spotting phase then is repeated for all the N-gram classes contained in the N-grams query set.

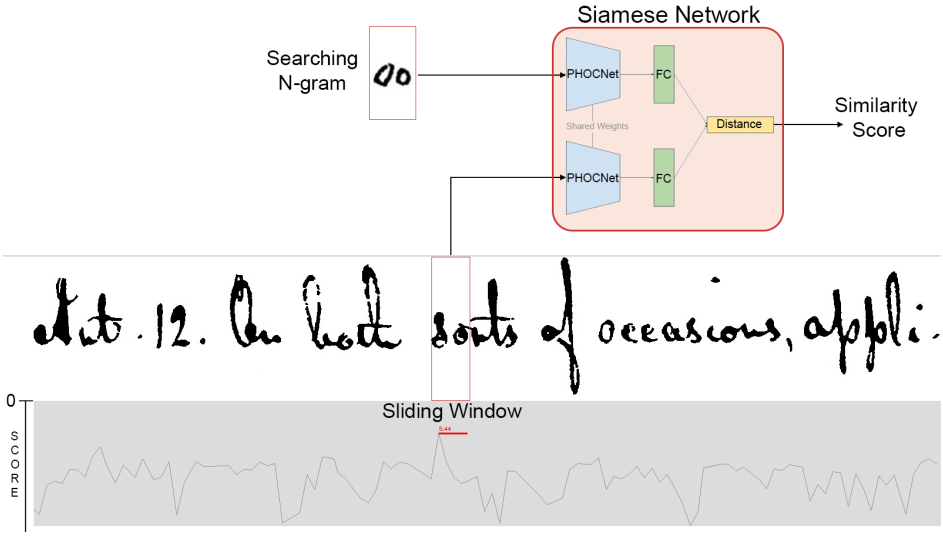

Figure 2: Sliding Window Architecture for N-gram potting:A sliding window crops the portions of images of a text line. The crop is fed into the Siamese Network together with the image of the reference N-gram to obtain the similarity score.

The system provides the option to choose the number α of samples for each class of N-grams to use for the search. If the search cardinality α is greater than 1, we could find multiple peaks in the same region from searches with different items of the same class of N-grams. To obtain a single similarity trend for each class of N-gram, all the two overlapping solutions s_1 and s_2, i.e. solutions with peaks in the same region of the row obtained with two different instances of N-grams of the same class, are merged by reshaping the score according to the following expression:

$$s = \min(s_1, s_2) - \gamma$$

where γ is the reward attributed to the new solution and is equal to:

$$\gamma = (1 - sigmoid(|s_1 - s_2|))(3/4)^{min(s_1,s_2)}$$

In this way, similarity scores are rewarded with a low difference between them and have values close to zero.

3.4. Word Reconstruction

The ultimate goal is to determine the position of a whole query word, starting from the analysis of the results of the N-gram spotting phase. To this end, we look for "*high-density areas*", i.e. areas of the text line where there are overlaps of searched N-grams. If the N-gram spotting phase was successful for all classes of N-grams, a density area could correspond to an area where the searched word occurs. However, it is important to note that the detection of density areas alone is not sufficient to confirm that these zones correspond to the position of the searched words. This is because the density area does not take into account the position of the N-grams and could therefore contain different anagrams of the query word. Moreover, the N-gram spotting phase is not error-free and could yield density zones that do not contain all the N-grams of the word. Therefore, a density area evaluation phase is required in which a confidence measure is assigned to each area. A confidence measure has been defined with a value in the range $(0, 100)$, where 100 represents maximum confidence estimated considering three criteria: *1) number of retrieved N-grams*; *2) mean similarity score of the retrieved N-grams*; *3) position of the retrieved N-grams*.

As for the first criterion, the more N-grams are detected in the density area, the greater the confidence measure. For a maximum confidence measure, the density area must have exactly the number of expected N-grams. More generally, the confidence level varies linearly with the number of detected N-grams. For example, if the number of detected N-grams equals half of the expected number, the confidence measure is halved.

In applying the second criterion, note that each N-gram was recognised with a similarity score. The lower the similarity score, the more reliable the prediction. The confidence measure can then be reshaped by subtracting from it the average value of the similarity scores of all N-grams belonging to the density area. In the optimal case, each detected N-gram has a similarity value of zero, resulting in a maximum probability for each N-gram. In this case, the confidence would not change because every N-gram in the area is safe.

To evaluate the position of the N-grams in the density area, we can calculate the pyramidal decomposition of the query word and consider the different sets of N-grams at the different levels of the representation. To calculate the representation, the word for each level must be divided into different parts corresponding to the depth of the level. In other words, level two of the representation consists of the query word divided into two parts, the third level consists of the word divided into three parts, and so on. We can assign the set of N-grams that make up every single sequence of the representation, building the pyramidal N-gram sets representation of the query word. Similarly, the pyramidal representation of the ordered N-grams of the density area can be calculated. The density area is divided into an increasing number of contiguous zones from time to time and the sets of N-grams belonging to the different zones are constructed. If the density area is consistent with the query word, consistency must be maintained between both the pyramidal representations at all levels. To assess this consistency, the number of N-grams of the pyramidal representation of the density area that does not match the pyramidal representation of the word query is counted. An N-gram of the density area is inconsistent with the word query representation if it is present in a set of N-grams at a particular level of the density zone representation but is not present in the relative set of the word query representation at the same level. The confidence value can then be reshaped based

on the ratio between the inconsistent and consistent N-grams. If all N-grams from level 2 of the representation are inconsistent, the confidence value is reduced by 100%. If, on the other hand, all N-grams are consistent, the restructuring does not affect the confidence value.

At this point, each detected density area is assigned a confidence measure, the higher the more likely it is to contain an instance of the searched word. In this way, once the system receives a query word, it can return a list of all the areas that may contain the word, each with its confidence measure.

4. Results

4.1. Dataset

The KWS system was tested on a selection of 20 pages from the Bentham collection [16]. The pages of the dataset are binarized using the Sauvola method [17] and then segmented into text lines. The set is divided into a subset of 5 pages, from which the reference alphabet of N-grams is created, and a test subset consisting of the remaining 15 pages. From the first 5 pages, together with their transcription, all N-grams with N equal to 2 and 3 are extracted. The decision to limit N to 3 is because considering character sequences consisting of more than 3 characters would, in our opinion, run counter to the premises of this work. We want to use N-grams as recognition primitives, keeping the sequences large enough to facilitate segmentation, but small enough to remain effective recognition primitives. Following the process of extracting the N-grams from the training set, we obtain a set of N-grams consisting of 1044 distinct classes. However, the dataset is highly imbalanced as the cardinality of each class ranges from a minimum of 1 to a maximum of 117, with 615 classes consisting of less than 3 items. To reduce the imbalance, the minimum cardinality of the classes was raised to 3 by simple image transformations and noise addition. In this way, the training set consists of 1044 classes totalling 5440 samples.

4.2. Training of the Siamese Network

The core of the N-gram spotting system is a Siamese Neural Network that uses a PHOCnet convolutional framework to extract features from images. For the experiments, the PHOCNet backbone was pre-trained with the IAM handwritten dataset [18]. The whole branch of the Siamese network, i.e. the PHOCNet backbone and the downstream fully connected layers, were fine-tuned according to a triplet loss [19] using the images of the N-grams. For this purpose, a training set and a validation set were created starting from the set of all N-grams extracted from the first 5 pages of the collection. For the training set, at most 10 elements were selected for each class and a training triplet (A, P, N) was defined for each of these elements. For each anchor element A, the element belonging to the same class is selected as the positive element P which provides a PHOCNet embedding that is more distant than the embedding of the anchor element. To define the negative element N, the 10 classes closest to the anchor class are selected based on the Levenshtein distance calculated on the N-gram labels. The element N is then randomly selected from this set with a probability of 80%, otherwise, it is randomly selected from the entire data set. In this way, '*hard*' triples are obtained that try to maximise the distance

between A-P and minimise the distance between A-N. Any images that are not in the training set at the end of the process are added to the validation set.

4.3. Experimental Results

By definition, the system returns in response to a query the confidence-ordered list of k images that supposedly contain instances of the searched word. As the number of instances of the words in the collection is unknown, setting the value of k may lead to underestimating the performance in terms of both Recall and Precision: given the value of k, it will lead to under estimating the Recall whenever there are more than k instances of the query word in the collections, while it will lead to underestimating the Precision in case of words with less than k instances. Figure 3 reports the results in terms of recall for different values of k (r@k) for two groups of words, the first of which consists only of in-vocabulary (IV) words and the second of OOV words that can be constructed from the reference dictionary of N-grams.

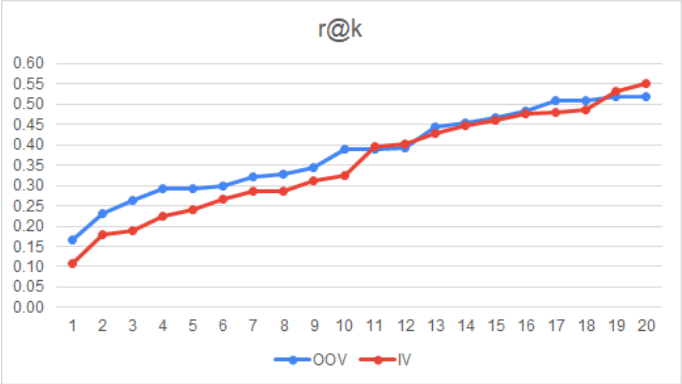

Figure 3: r@k trend: the recall increase whit the dimension of the output list k.

5. Discussion and Conclusion

The results show that the system can recover words from the set of IV words as well as from the set of OOV words with a similar recall rate, independently of the value of k. Indeed, the N-gram spotting phase takes place in the N-gram search space, and as long as all N-grams are available to search for the query word, it makes no difference to the system whether it is an IV word or an OOV word.

An important feature of the proposed system is that it is segmentation-free, i.e. it avoids both character and word-level segmentation, which are anything but easy in cursive handwriting. The price to pay is that the system also retrieves parts of words that are similar to the searched word. In other words: if the query word is part of a longer word the system can retrieve these parts, for instance in the case of the query word "perform", the system can recover part of the word "performed", as shown in Figure 4, with a fairly high confidence value since actually, the transcription of the selected crop is the same, or very similar, to the query word. Similarly, there

are also cases when not all the necessary N-grams of the query word can be spotted, but most of them are, as in the case of the query word "appellate" and the instance of the word "Appeal" shown in Figure 4.

The experimental results, eventually, show that the Siamese Network does not perform satisfactorily, as it is shown by the case on top of the figure, where the image of the N-gram "mo" is matched with instances of the N-gram "na" because of their similarity, as well as the case on the bottom, where the images of the N-gram "lla" is matched with instances of the N-gram "int" despite they do not similar. This may depend mostly on the small size of the training set, which is unavoidable due to the size of the collections we are dealing with, as already mentioned. A possible way to overcome this limitation is to apply data augmentation techniques to a larger extent than we have done in this study.

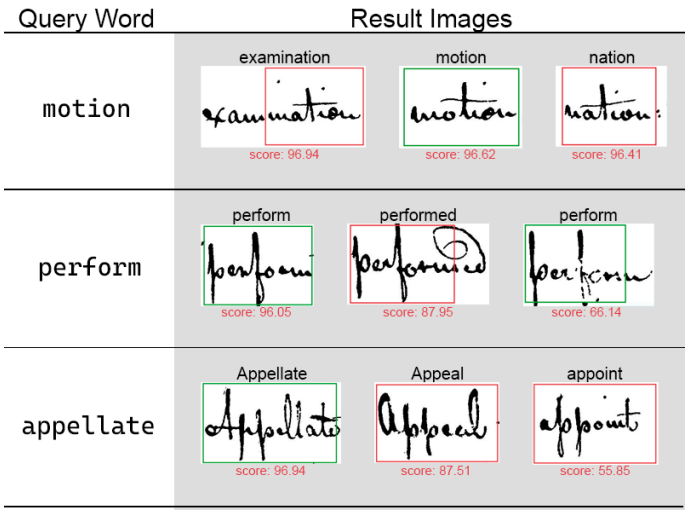

Figure 4: Example of system results: Examples of images retrieved by the system for a few query words. The correct outputs are highlighted with green boxes, while the incorrect ones are in red

In this paper, we have presented a KWS system that makes it possible to retrieve OOV words, thus circumventing the limitation of the dictionary-based approaches, by searching for N-gram images within the text line, thus avoiding both character and word-level segmentation. The experimental results have shown that the system can spot OOV words with similar performance to the case of IV words. They also show that the overall performance is very encouraging, considering the small size of the training set, with many classes and a few samples per class. This certainly hampers the performance of the Siamese Network, and thus we are currently investigating data augmentation techniques as a possible solution to achieve better performance.

References

[1] A. Sulaiman, K. Omar, M. F. Nasrudin, Degraded historical document binarization: A review on issues, challenges, techniques, and future directions, Journal of Imaging 5 (2019) 48.

[2] J. P. Philips, N. Tabrizi, Historical document processing: historical document processing: a survey of techniques, tools, and trends, arXiv preprint arXiv:2002.06300 (2020).
[3] A. P. Giotis, G. Sfikas, B. Gatos, C. Nikou, A survey of document image word spotting techniques, Pattern recognition 68 (2017) 310–332.
[4] A. Marcelli, A. Parziale, R. Senatore, Some observations on handwriting from a motor learning perspective., in: AFHA, volume 1022, Citeseer, 2013, pp. 6–10.
[5] A. M. Wing, Motor control: Mechanisms of motor equivalence in handwriting, Current biology 10 (2000) R245–R248.
[6] R. Manmatha, C. Han, E. M. Riseman, Word spotting: A new approach to indexing handwriting, in: Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 1996, pp. 631–637.
[7] T. M. Rath, R. Manmatha, Word spotting for historical documents, International Journal of Document Analysis and Recognition (IJDAR) 9 (2007) 139–152.
[8] T. Konidaris, A. L. Kesidis, B. Gatos, A segmentation-free word spotting method for historical printed documents, Pattern analysis and applications (2016).
[9] J. Almazán, A. Gordo, A. Fornés, E. Valveny, Word spotting and recognition with embedded attributes, IEEE TPAMI (2014).
[10] J. Puigcerver, A. H. Toselli, E. Vidal, Querying out-of-vocabulary words in lexicon-based keyword spotting, Neural Computing and Applications 28 (2017) 2373–2382.
[11] A. H. Toselli, E. Vidal, V. Romero, V. Frinken, Hmm word graph based keyword spotting in handwritten document images, Information Sciences 370 (2016) 497–518.
[12] L. R. Rabiner, C.-H. Lee, B.-H. Juang, J. G. Wilpon, Hmm clustering for connected word recognition, in: International Conference on Acoustics, Speech, and Signal Processing,, IEEE, 1989, pp. 405–408.
[13] A. Brakensiek, J. Rottland, G. Rigoll, Handwritten address recognition with open vocabulary using character n-grams, in: Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition, IEEE, 2002, pp. 357–362.
[14] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, U. Pal, Signet: Convolutional siamese network for writer independent offline signature verification, arXiv preprint arXiv:1707.02131 (2017).
[15] S. Sudholt, G. A. Fink, Phocnet: A deep convolutional neural network for word spotting in handwritten documents, in: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE, 2016, pp. 277–282.
[16] J. A. Sanchez, A. H. Toselli, V. Romero, E. Vidal, Icdar 2015 competition htrts: Handwritten text recognition on the transcriptorium dataset, in: ICDAR, 2015.
[17] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, Pattern recognition 33 (2000) 225–236.
[18] U.-V. Marti, H. Bunke, The iam-database: an english sentence database for offline handwriting recognition, International Journal on Document Analysis and Recognition 5 (2002) 39–46.
[19] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815–823.