Here is the Title of our Work

Eugen Rusakov, Sebastian Sudholt, Fabian Wolf and Gernot A. Fink
Department of Computer Science
TU Dortmund University
44227 Dortmund, Germany
Email: {firstname.lastname}@tu-dortmund.de

***Abstract*—The generation of word hypotheses for segmentation-free word spotting on document level is usually subject to heuristic expert design. This involves strong assumptions about the visual appearance of text in the document images. In this paper we propose to generate hypotheses with text detectors. In order to do so, we present three detectors that are based on SIFT contrast scores, CNN region classification scores and attribute activation maps. The uncertainty in the detector scores is modeled with the extremal regions method. Retrieving word hypotheses is based on PHOC representations which we compute with the TPP-PHOCNet. We evaluate our method on the George Washington dataset and the ICFHR 2016 KWS competition benchmarks. In the evaluation we show that high word detection rates can be achieved. This is a prerequisite for high retrieval performance that is competitive with the state-of-the-art.**

I. Introduction

Word spotting is an efficient method for making document images searchable. Therefore, it provides an essential functionality for working with large document image collections. The approach is efficient since the search functionality is directly implemented and not a by-product of a more complex task, typically transcription. Most commonly, the search query is either given as a word image in query-by-example scenarios or as text in query-by-string scenarios. All word spotting methods need to either explicitly (segmentation-based) or implicitly (segmentation-free) segment the document collections into word image hypotheses. State-of-the-art methods project the word images into an embedded attribute space [**?**] using *Convolutional Neural Networks (CNN)* [**?**], [**?**]. In this space, word spotting can then be accomplished through a simple nearest neighbor search. For historic documents, automatic segmentation is especially challenging due to high variability in writing style, document layout, visual appearance of ink and paper, as well as aging artifacts.

Segmentation methods that have been successful in modern document images, such as projection profiles or connected components, are likely to fail for historic documents. Instead, these methods have to be manually tuned to the document collection's specificities. Interesting segmentation methods have been presented in [**?**] and [**?**]. Within the scale space approach in [**?**], some parameters can automatically be derived from data. The approach in [**?**] uses a CNN for classifying segmentation hypotheses. The visual word appearance is, therefore, learned from annotated sample data. However, methods addressing solely segmentation need to detect words without recognizing them or, in case of word spotting, without taking relevance to the query into account. Therefore, these methods have to rely on discriminative characteristics of the document collections considered. In the challenging scenario of historic document images, it remains questionable, if suitable characteristics can automatically be extracted. This aspect can potentially limit the generalization capability.

In order to be more robust with respect to word size variability, our segmentation-free word spotting method is inspired by approaches using local text detectors. In many cases text detectors are solely built on connected components, e.g. [**?**], [**?**], [**?**]. This has two important drawbacks. First, the detectors are dependent on document image binarization. In historic document images binarization is difficult due to fading ink, low contrast and inhomogeneous backgrounds. This makes detections imprecise. Second, it can be difficult to derive word hypotheses from connected components. Since connected components can represent parts of words, single words or multiple words, heuristic strategies for combining connected components are required.

For these reasons, we propose to generate word hypotheses based on higher-level feature representations that indicate word occurrences. First, we predict scores for certain document image regions. These scores reflect whether the respective region contains text or not. The uncertainty of these scores is then explicitely modeled with extremal regions (ERs) [**?**] that have been very successful for text detection in natural scene images, cf. [**?**]. The ER approach generates hypotheses of word bounding boxes. For these, PHOCs are predicted using a TPP-PHOCNet [**?**]. This is essentially a *Region-based CNN (R-CNN)* [**?**] framework. After predicting the PHOCs, word spotting can be performed through a nearest neighbor search.

Generating the local text scores is a critical part of our method. Here, we consider three different approaches: SIFT contrast scores, local region classification scores generated with a CNN and local word region scores obtained with an extension of CNN class activation maps [**?**].

Sec. II presents segmentation-free word spotting methods and briefly reviews extremal regions. Our segmentation-free word spotting approach and its evaluation are presented in Sec. III and Sec. IV. Finally, conclusions are drawn in Sec. V.

II. Related Work

Word spotting methods that are addressing segmentation and retrieval jointly are referred to as segmentation-free. In order

to address the segmentation problem at document level, mainly two different approaches can be identified. Based on local text detectors, different competing word hypotheses are obtained, cf. [**?**], [**?**], [**?**], [**?**], [**?**]. In contrast, patch-based approaches densely sample word hypotheses from the document images, cf. e.g., [**?**], [**?**], [**?**], [**?**], [**?**]. By searching the full document, patch-based approaches do not rely on heuristic detectors. However, they limit the search to a single patch size per query, thus assuming that the size variability is relatively low. Finally, in both approaches word hypotheses are ranked according to similarity with the query and overlapping hypotheses are suppressed if they obtained a non-optimal score. Segmentation-free methods, therefore derive the segmentation during the retrieval process and do not rely on a given segmentation that is assumed to be correct.

Segmentation-free word spotting based on PHOC representations, cf. [**?**], has been presented in [**?**] and [**?**] for the first time. Here, the document is divided into a number of blocks and a PHOC is predicted for each block. For efficient patch-based retrieval, an integral image over the block-wise PHOC vectors is computed. In order to improve the results, a regression is learned which projects PHOCs and predictions into a common subspace. At query time, the query PHOC is projected into this subspace. The similarity between the query and the patches is then determined through a dot product. While all patches are considered in [**?**], the approach presented in [**?**] adds an indexing stage in order to efficiently detect regions of interest. In this stage, connected components in close proximity to each other are combined in order to obtain word hypotheses. For retrieval, candidate word regions, obtained from the index, define the document image search area for the patch-based framework presented in [**?**]. For query-by-example [**?**] the patch size equals to the size of the query word image and for query-by-string [**?**] the patch size is estimated from training word images.

Very recently, a method for proposing regions of interest and representing them with word string embeddings in an integrated manner has been presented [**?**]. The authors train a Region Proposal Network in order to predict bounding boxes. Furthermore, the predicted bounding boxes are augmented with a set of heuristically generated region proposals. A word string embedding is computed for each region. Regions are retrieved according to cosine distance with the query.

Related to word segmentation is text detection in natural scene images. These methods need to cope with large variability in the visual appearance of text. While this problem domain may seem to be less constrained compared to word segmentation, it has to be noted that the reliable detection of word boundaries in historic document images requires to correctly recognize the text in the document images first. In order to avoid recognition in our segmentation-free word spotting method, we are inspired by *extremal regions*.

Extremal regions are part of the maximally stable extremal region (MSER) blob detection method [**?**]. The key idea is to derive blobs based on connected components in thresholded images which are referred to as extremal regions (ER). In order to avoid the selection of a single threshold, MSERs are detected within an ER scale space. This scale space is obtained by thresholding the image at all image intensity values.

Building on the MSER approach, a method for text detection in natural scene images is presented in [**?**]. The method consists of different stages where character candidates are first detected, grouped into triplets and finally merged into line regions. For this purpose, ERs are extracted from color image channels. In contrast to the MSER blob detection [**?**], the ER stability is defined on probabilistic character class scores obtained with a boosted decision tree [**?**]. The final decision whether an MSER becomes a character candidate is determined with an SVM classifier.

In order to avoid the limitations of a basic connected component-based word detection, cf. e.g., [**?**], or patch-based frameworks, cf. e.g., [**?**], we propose to build ERs on top of pixel-wise text detector scores. This way, we avoid the need for classifying ERs into words and non-words which would require a word recognizer. The main advantage over a word recognizer is that the detector is applied on the entire document image and not limited to document image regions that have been heuristically selected. This way ERs model different variants for word candidates, particularly in document image regions where the detector scores are ambiguous. In order to do so, we carefully adapt the ER selection strategy. Furthermore, the integration and combination of different text detection approaches is straight forward.

To the best of our knowledge this is the first time that ERs are extracted based on detector scores. ERs have not been used in the context of segmentation-free word spotting in historic document images, before.

Similar to the ResNet architecture, Dense Convolutional Networks (DenseNets) [**?**] also utilize identity connections. Each layer generates a new set of feature-maps, which are concatenated with the feature-maps provided by the skip-connection. Following this connectivity scheme, each layer receives all preceeding feature-maps as an input. The number of feature-maps added by each layer is refered to as the network's growth rate. Already small growth rates are sufficient to achieve state-of-the-art results, resulting in very narrow layers. To avoid the concatenation of differently sized feature-maps, DenseNet are organized into densely connected blocks. Consequently, pooling layers are only used outside the dense blocks. The model compactness of DenseNets is further improved by the introduction of compression layers. A compression layer corresponds to a convolutional layer with kernel size one. DenseNets tend to make stronger use of high-level features learned at the end of a dense block. Therefore, the convolutional layer reduces the number of features maps by a compression factor. The DenseNet architecture outperformed other networks such as ResNets in various state-of-the-art benchmark experiments in the field of image classification. It has been shown, that the dense architecture is especially parameter efficient and achieves competitive results, with significantly reduced numbers of parameters.

III. Method

A. PHOCnet Architecture

1) ResNet:

2) DenseNet: For our experiments, we use a DenseNet with two densely connected blocks. Before entering the first block, a convolutional layer with 32 output channels and a $2x2$ average pooling layer are applied. Following the dense connectivity pattern, the first block consists of 30 convolutional layers with kernel sizes $3x3$. For the second block 60 densely connected convolutional layers are used. The transition layer between both blocks uses a convolutional layer with kernel size $1x1$ and a $2x2$ average pooling layer. The convolutional layer compresses the number of feature maps by a factor of 0.5 Analogue to the *TPP-PHOCNet* architecture, our DenseNet makes use of a 5-level TPP layer in combination with a Multilayer Perceptron.

B. Loss Function

IV. Experiments

For the experiments with the PHOCnet architectures we used three benchmark datasets described in Sec. IV-A and a evaluation protocol (Sec. IV-B) for segmentation-based wordspotting commonly used in the literature. In Sec. IV-C we describe the training setup with all network hyper-parameter used to train the networks. Afterwards we discuss the retrieval results achieved by our methods and compare the architectures in Sec. IV-D.

A. Datasets

We evaluate our method on three publicly available data sets. The first is the **George Washington (GW) data set**. It consists of 20 pages that are containing 4,860 annotated words. The pages originate from a letterbook and are quite homogeneous in their visual appearance. However, particularly for smaller words the annotation is very sloppy. As the GW data set does not have an official partitioning into training and test pages, we follow the common approach and perform a four-fold cross validation. Thus, the data set is split into batches of five consecutive documents each.

The second benchmark is the large **IAM off-line dataset** comprising 1,539 pages of modern handwritten English text containing 115,320 word images, written by 657 different writers. We used the official partition available for writer independent text line recognition. We combined the training and validation set to 64,XXX word images for training, and used the 13,XXX word images in the test set for evaluation. We exclude the stop words as queries, as this is common in this benchmark.

Botany in British India (Botany) is the third benchmark introduced in the Handwritting Keyword Spotting Competition, held during the 2016 International Conference on Frontiers in Handwriting Recognition. The training data of Botany was partitioned into three different training sets from smaller to larger (*Train I* 1684, *Train II* 5295, and *Train III* 21981 images)

B. Evaluation Protocol

We evaluate the two CNN architectures for the data sets GW and IAM in the segmentation-based word spotting standard protocol proposed in [**?**]. For the *QbE* scenario all test images are considered which occure at least twice in the test set. For *QbS* only unique string are used as queries. The PHOCnet predictes a attribute representation for each given query, afterwards a nearest neighbor search is performed by comparing the attribute vectors with the *cosine similarity*. The Retieval list is created by sorting the computed distances from nearest to farthest. Whereas in the QbE scenario only predicted attribute representation are considered for a nearest neighbor search, the QbS scenario takes also the direct computed string embedding from the transcription (query)

C. Training Setup

D. Results & Discussion

The QbE results achieved with our word hypothesis methods are listed in Tab. I. The DRs for all datasets show that we obtain very accurate results. High DR is a prerequisite for high retrieval performance. Given a query, only the hypotheses can be retrieved that have been detected, beforehand.

An important result is that DR and retrieval performance can be improved when word hypothesis heights are quantized to values in $[h_{min}, h_{min} + 5, \cdots h_{max}]$. These parameters are estimated such that h_{max} is the maximum word height in the training set and h_{min} is set to the typical line height in the training set. On the GW dataset h_{min} is set to 70 pixels, to 150 pixels on Konzilsprotokolle and to 120 pixels on Botany. In Tab. I these experiments are denoted with *quant*. The positive effect has mainly three reasons. First, quantization is required on GW due to the sloppy annotation of smaller words that are arbitrarily padded with white space, cf. [**?**]. Accurate word hypotheses will, therefore, not be considered as relevant. Second, the TPP-PHOCNet tends to favor bounding boxes that fit the text core areas. Thus, h_{min} defines a lower bound for all word hypotheses. Third, retrieval speed can be improved by suppressing similar hypotheses.

Regarding the text detectors, we evaluate the heuristic SIFT and the learned LRC and AAM methods. Further, we use linear combinations of SIFT and LRC or AAM scores, as denoted with LRC+SIFT and AAM+SIFT in Tab. I. While accurate results can be achieved with SIFT, detection and retrieval results can be improved by adding the learning-based methods. The best DRs are obtained with detectors including LRC. This is due to the explicit modeling of the visual appearance of word boundaries. Consequently, this mostly applies to retrieval performance as well. An exception can be observed on GW where the training annotations for the LRC-CNN can be considered as noisy (see above). In contrast, the AAM detector learns the visual appearance of text. The results for the AAM detectors show that the TPP-PHOCNet focusses on text core areas the most. Therefore, word hypothesis bounding boxes tend to fit closely to the words in the document.

In Tab. II we consider QbE and QbS scenarios with 50% and 25% region overlap for the segmentation-free scenario. With

Table I
COMPARISON OF THE DIFFERENT TEXT DETECTION METHODS FOR THE QUERY-BY-EXAMPLE EXPERIMENTS [%]

Architecture	George Washington		IAM		Botany?	
	mR	mAP	mR	mAP	mR	mAP
ResNet Config 1	73.2	64.8	75.9	66.3	89.9	86.2
ResNet Config 2	88.4	80.7	77.9	68.9	91.6	87.1
DenseNet Config 1	81.8	77.0	80.5	71.6	81.6	76.1
DenseNet Config 2	86.3	80.1	82.2	73.0	90.1	86.1
Random Attributes Config 1 ?	35.4	31.0	63.6	53.9	77.8	70.3
Random Attributes Config 2 ?	67.8	59.9	68.2	59.1	89.5	83.5

respect to our best performing text detector configurations, the trend in retrieval accuracy that we observed for QbE, can also be confirmed for QbS. A closer look at the results for 25% region overlap reveals that our word detections are often tighter than the original bounding box annotations. Word hypotheses that are ranked high in the retrieval list, have not been considered as relevant when using 50% region overlap.

To obtain a feasible number of word hypotheses we adjusted the number of ER-thresholds to 50 for all detectors. In our best configuration on GW (c.f. Tab. II) around 10 000 hypotheses per page were computed. After applying the aspect ratio filter approximately 5 400 regions per query and page are left for scoring. This low number of filtered hypotheses leads to an average query time of 60ms per page.

In comparison with the state-of-the-art our results compare very favourably. We outperform the previous results on Botany and Konzilsprotokolle by a large margin. On GW only the very recently presented Region Proposal CNNs [**?**] achieve better results. However, the authors use an additional CNN combined with brute-force hypotheses generation in order to cope with the inaccurate word annotations.

V. CONCLUSION

We have presented a method for segmentation-free word spotting which combines a novel ER-framework with a TPP-PHOCNet in an R-CNN framework. The ER method generates word hypotheses for which PHOCs are predicted. We proposed three different detectors in order to predict local text scores. This way, we avoid using a patch-based framework as well generating large amounts of region hypotheses blindly. In the experimental evaluation we achieve results that are competitive with the state-of-the-art.

Table II
STATE OF THE ART COMPARISON (RESULTS ARE GIVEN IN MAP [%] AT DIFFERENT OVERLAP THRESHOLDS)

Method	George Washington				Botany				Konzilsprotokolle			
	QbE		QbS		QbE		QbS		QbE		QbS	
	50%	25%	50%	25%	50%	25%	50%	25%	50%	25%	50%	25%
SIFT	64.8	71.1	70.7	76.5	66.3	75.2	68.9	79.0	86.2	91.1	84.6	91.5
$SIFT_{quant}$	80.7	90.6	82.5	89.1	68.9	76.4	72.0	80.2	87.1	94.0	87.4	92.9
LRC+SIFT	78.3	89.3	81.2	88.9	73.0	79.9	76.2	83.6	88.4	94.9	86.6	95.6
LRC+$SIFT_{quant}$	81.0	92.0	83.6	90.5	**74.5**	**80.4**	**78.8**	**85.3**	**91.1**	95.6	**89.9**	95.3
AAM+SIFT	62.3	69.0	70.0	76.1	67.8	75.4	71.0	80.1	84.2	94.9	81.9	95.2
AAM+$SIFT_{quant}$	81.6	92.0	84.6	90.6	69.4	75.9	74.0	80.3	89.6	**96.2**	88.9	**96.0**
BoF-HMM [**?**]	–	–	76.5	80.1	–	–	–	–	–	–	–	–
Ctrl-F-Net [**?**]	**90.9**	**97.0**	**91.0**	**95.2**	–	–	–	–	–	–	–	–
TAU [**?**]	–	–	–	–	37.48	–	–	–	61.78	–	–	–
Attribute-SVMs+RR [**?**]	–	–	73.7	–	–	–	–	–	–	–	–	–