Historical Map Toponym Extraction for Efficient Information Retrieval

Ladislav Lenc[1,2], Jiří Martínek[1,2], Josef Baloun[1,2], Martin Prantl[1,2], Pavel Král[1,2]

[1] Dept. of Computer Science & Engineering
Faculty of Applied Sciences
University of West Bohemia
Plzeň, Czech Republic
[2] NTIS - New Technologies for the Information Society
Faculty of Applied Sciences
University of West Bohemia
Plzeň, Czech Republic
{llenc,jimar,balounj,perry,pkral}@kiv.zcu.cz

Abstract. The paper deals with detection, classification and recognition of toponyms in hand-drawn historical cadastral maps. Toponyms are local names of towns, villages and landscape features such as rivers, forests etc. The detected and recognized toponyms are utilized as keywords in an information retrieval system that allows intelligent and efficient searching in historical map collections. We create a novel annotated dataset that is freely available for research and educational purposes. Then, we propose a novel approach for toponym classification based on KAZE descriptor. Next we compare and evaluate several state-of-the-art methods for text and object detection on our toponym detection task. We further show the results of toponym text recognition using popular Tesseract engine.

Keywords: historical maps · toponyms · text detection · OCR · IR · FCN

1 Introduction

Information retrieval (IR) in historical documents is a challenging task. Documents occur mostly as digital images with no text layer that makes an efficient information retrieval very difficult and cumbersome. A special group of such documents are historical maps. The maps contain several interesting pictographic elements: boundary stones and milestones, scale, and other symbols determined by the map legend. In addition, they also include so-called toponyms, i.e. local names of towns, municipalities, villages, but also forests, hills, or paths. Such toponyms are crucial for IR and make up the main keywords that are mostly found in user queries in the IR systems.

We focus on toponym detection, classification and recognition in order to annotate each map sheet with a set of keywords to facilitate efficient searching and IR. The task is solved within an ongoing project that concentrates on seamless connection of cadastral maps and efficient searching in them [13]. We focus on scanned historical cadastral Austro-Hungarian maps from the 19th century provided by Czech Office for Surveying, Mapping and Cadastre (CUZK)[1].

[1] https://www.cuzk.cz/

We create a novel annotated dataset which represents the first main contribution of this work. The dataset contains 800 annotated map sheets and it is freely available for research and educational purposes. The second important contribution consists in proposition of a novel method for map toponym classification based on KAZE descriptor. We also compare and evaluate several state-of-the-art methods for text and object detection on our toponym detection task using the newly created dataset. Finally, we show the results of optical character recognition (OCR) using Tesseract system [22].

The rest of the paper is organized as follows. The next section covers the related work. Our dataset is described in Section 3. We present the approaches for text detection, toponym classification and OCR in Section 4. Experiments are described in Section 5 and Section 6 concludes the paper.

2 Related Work

The detection of regions of interest (RoI) is one of essential tasks in the computer vision field. There are many challenges, e.g.: noise, image distortion, video frames. The quality of the detection results is often crucial since it is usually followed by further steps such as OCR or classification and can negatively influence the results of the successive steps.

Many scene text detection approaches are evaluated on well-known benchmark datasets (some of them are Street View Text (SVT) [23], IIIT5k [17], ICDAR2003 [16], ICDAR2013 [12], ICDAR2015 [11]).

Traditional approaches are based on manual feature engineering such as: Stroke Width Transform [5], Maximally Stable Extremal Regions [18, 19] or others [10, 14]. Individual character recognition approaches are also employed [4, 28].

Currently, the mainstream methods in this topic are deep neural network models like Region-based Convolutional Neural Networks (R-CNNs). These networks typically use pre-trained ResNet 50 or VGG 16 backbones to extract features and region proposals. A significant approach for efficient object detection using deep convolutional networks has been proposed in [7]. R-CNN system consists of region proposal, feature extraction and classification modules. Proposed regions are separately processed and classified for presence of the desired objects. The system achieved promising results but the drawbacks are multi-stage training and the computational inefficiency due to the number of regions and thus classifications per single image.

An approach based on Spatial Pyramid Pooling has been proposed by He et al. [9]. The main strength of this approach is the capability to use a specific pooling operation that generates a fixed-length representation regardless of image size or scale.

In 2015, Ross Girshick went further in his research proposing Fast R-CNN [6] with several innovations to deal with the drawbacks and speed-up. Compared to R-CNN, the image features are calculated only once. Based on the regions, the region features are pooled into a fixed size feature space using RoI pooling (which can be understood as 1-level Spatial Pyramid Pooling).

The selective search algorithm for region proposals was still considered a drawback resulting in Faster R-CNN model [21]. Instead of selective search algorithm, the model is extended by Region Proposal Network which is faster.

Additionally, the task of instance segmentation can be solved by mask R-CNN [8]. It adopts the Faster R-CNN scenario and outputs also segmentation mask for each RoI.

You Only Look Once (YOLO) [20] is even faster than Faster R-CNN and allows real-time object detection. Briefly, as says the network name, you only look once at the image to predict the bounding boxes and the class probabilities for these boxes. It uses the grid over the image and predicts the boxes for each cell of the grid. The trade-off is the limited amount of predicted boxes per cell in the grid.

EAST: An Efficient and Accurate Scene Text Detector has been proposed by Zhou et al. [29] in 2017. In 2019, Baek et al. proposed Character Region Awareness for Text Detection (CRAFT) [2] which is able to detect characters from the scene without necessity of annotating such characters and glyphs individually.

Some approaches for text detection and recognition are focused on historical maps (e.g. Weinman [24, 25]). This task is very challenging since the maps (especially historical) are very different and heterogeneous. Moreover, the texts are often warped and rotated. Chiang and Knoblock [3] present a general semi-automatic approach that should require minimal user effort for the text recognition.

3 Dataset

The data are annotated for all three sub-tasks that we have to handle: 1) toponym detection, 2) toponym classification and 3) text recognition. We thus need to locate and extract text regions, determine the toponym category and finally recognize the text from it. There are two groups of toponyms:

1. Municipal toponyms (cities, villages, significant buildings);
2. General toponyms (forests, roads, hills, swamps, rivers and other objects).

These two groups can be separated from each other since the municipal toponyms are written with a printed-like font (capital letters) while the general toponyms are handwritten. For simplicity, in the rest of the article, we will refer toponyms as either *printed* (P) or *handwritten* (H). An example of a map sheet with detected and labeled toponyms can be seen in Figure 1.

There is also one important text element in the map sheet, so-called nomenclature. See the top right corner of Figure 1 and more detailed in Figure 2. It determines the location of the map sheet in the coordinate system. The nomenclature is also labeled in our dataset, however, we do not use it in this work.

The created dataset consists of 800 manually annotated map sheets which cover various areas across the Czech Republic[2]. In total, we marked 3700 regions indicating toponyms or nomenclatures (see Table 1). The vast majority of toponyms are thus handwritten since there are much more general objects throughout the maps. The dataset is freely available for research and educational purposes[3].

[2] The Czech Republic was a part of the Austria-Hungary Empire until 1918.

[3] `https://corpora.kiv.zcu.cz/nomenclature/`

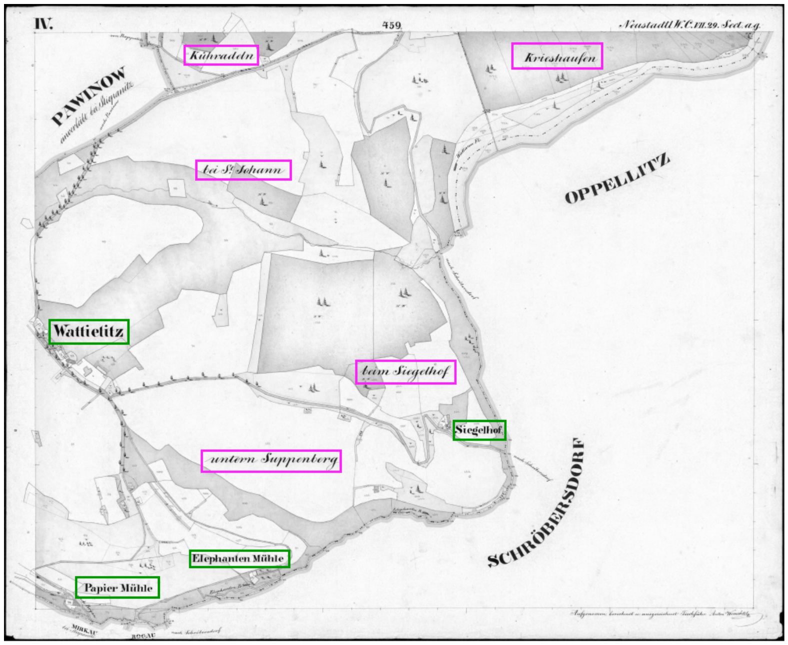

Fig. 1. Example of a map sheet with highlighted toponyms. Green-colored are municipal toponyms with printed-like font and purple-colored are general toponyms with handwritten font. Not labeled curved text specifies information about neighboring map sheet and it is not used in our case.

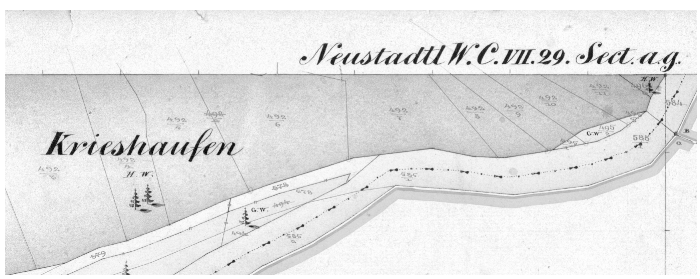

Fig. 2. Example of a nomenclature (right top corner)

Table 1. Numbers of nomenclatures, handwritten and printed toponyms within our dataset.

Dataset	Map Sheets	Nomenclatures	Handwritten Toponyms	Printed Toponyms
Train	650	650	2050	335
Test	100	100	305	41
Dev	50	50	141	28

4 Toponym Processing Approach

Within this section, we describe the approach we used in all our tasks. The overall pipeline is depicted in Figure 3. Three blue boxes represent models for particular sub-tasks: text detection, toponym classification and OCR. Note that the text detection model includes toponym classifier in the case of Faster R-CNN and YOLOv5 methods.

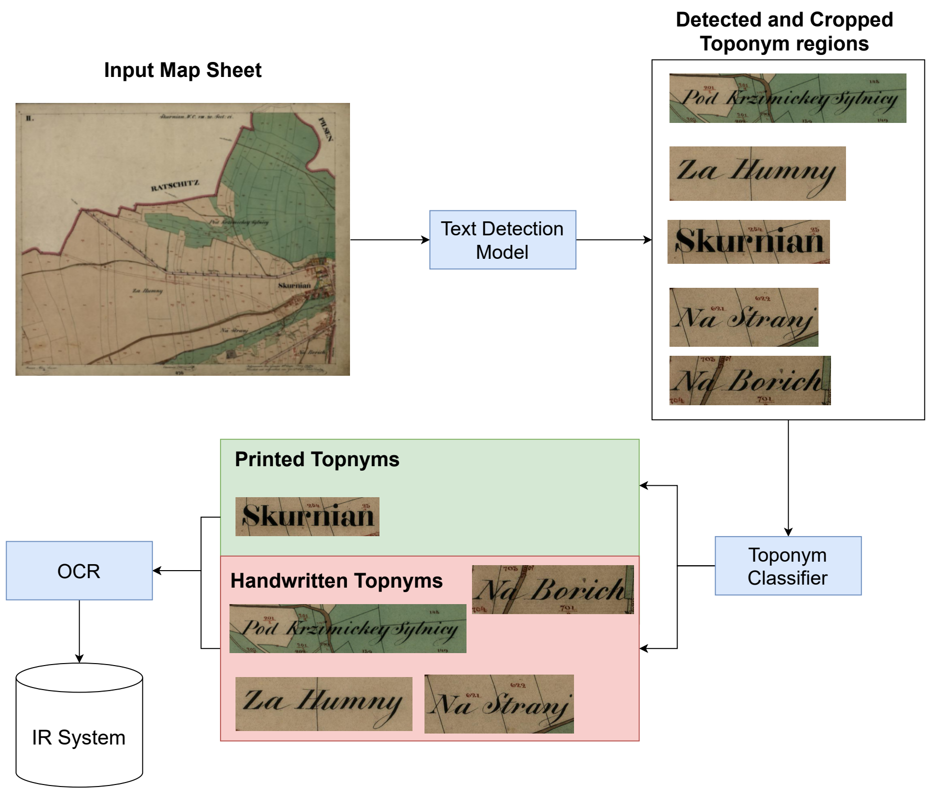

Fig. 3. Overall Processing Pipeline

4.1 Toponym Detection Methods

As a baseline we use a simple algorithm based on the connected components analysis (CCA). We first apply a median filter, binarize the image and use morphological dilation to connect single letters into one larger component. Then we use the CCA to identify large enough components that should represent the toponyms.

The more sophisticated approaches are based on neural networks and therefore, they must be trained. We use the training part of the created dataset for this task.

The following neural detectors presented in details in Related Work section are evaluated and compared:

- Faster R-CNN model with ResNet-50-FPN backbone;
- EAST: an efficient and accurate scene text detector;

- HP-FCN: High Performance Fully Convolutional Network [26];
- YOLO: You Only Look Once (we use YOLOv5 for our experiments).

YOLOv5 and Faster R-CNN are able to realize the detection and classification together. However, the other models predict only a bounding box (region of interest). In this case, we use for classification a novel algorithm based on KAZE key-point detector described below.

4.2 Toponym Classification Method

The proposed algorithm for toponym classification into printed and handwritten categories is inspired by the algorithm for writer identification based on image descriptors from [27]. The authors utilized SIFT [15] image descriptor in combination with clustering for image representation. The representations were then compared using χ^2 distance and k-nearest neighbours algorithm. They further utilized a finer identification based on contour-directional features to get the final categories.

In the proposed method, we have used faster and smaller KAZE [1] descriptor instead of the SIFT. Our comparison procedure is simplified. We have applied only a single-step comparison compared to the scheme used in the original paper. The algorithm assumes that we already have cropped toponym images obtained by a toponym detection method.

Codebook Generation The first step of the proposed algorithm is a codebook generation. It is based on a training set with known labels. Initially, all input image regions are pre-processed in order to reduce noise and to remove some unnecessary objects. We perform binarization followed by connected component analysis to preserve only components that are large enough to be part of the text content (glyphs). The minimal size of a component is based on the input resolution and degree of noise (in our case components with area smaller than 100 px are excluded).

The KAZE detector is applied on all pre-processed regions. We thus obtain a set of key-points and corresponding descriptors for each region. All descriptors are put together and the resulting set of descriptor vectors is clustered with K-means algorithm. We work with normalized descriptors (sum up to 1.0). The number of clusters K is a key parameter of this approach. Based on the recommendation from [27] we have tested several values for K and finally set $K = 100$. This process is described in Figure 4.

Image Representation Image representation is calculated based on the codebook. For each descriptor vector, we find the closest cluster using Euclidean distance. Even though the descriptor vectors have a relatively high dimension (64 or 128), this simple metric has sufficient results based on our preliminary experiments. We have also tested Cosine distance, χ^2 and Bhattacharyya distances, but the results were similar or even worse. The representation is then a histogram of size K where each bin represents how many times the given centroid was the closest to a descriptor vector. Finally, the histograms are size-normalized. This process is depicted in Figure 5.

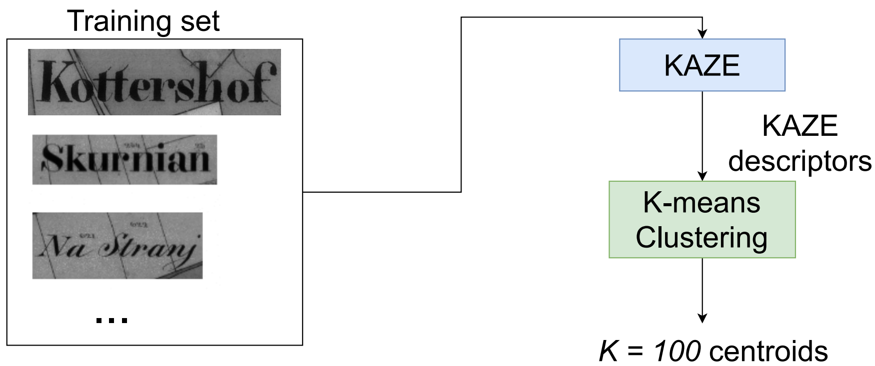

Fig. 4. Codebook generation process

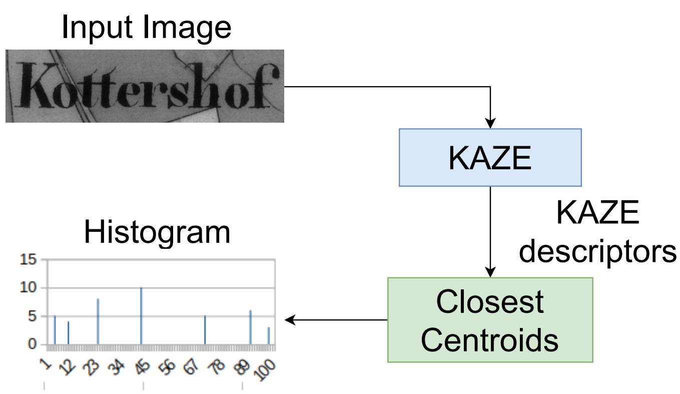

Fig. 5. Image representation creation

Prediction The prediction of an unknown text region is based on comparison of its representation with representations of known samples (training set). We first pre-process the image and obtain its histogram representation. Then we find N most similar histograms from the training set using Bhattacharyya distance. Based on our experiments, this distance is more suitable for the comparison of histograms than other traditional distance measures. The predicted class is determined as the majority class occurring in the N most similar histograms. Figure 6 shows the prediction phase of our approach.

The biggest advantage of this algorithm is the fact that only a small amount of training examples is sufficient for reasonable results (comparing to the neural network models).

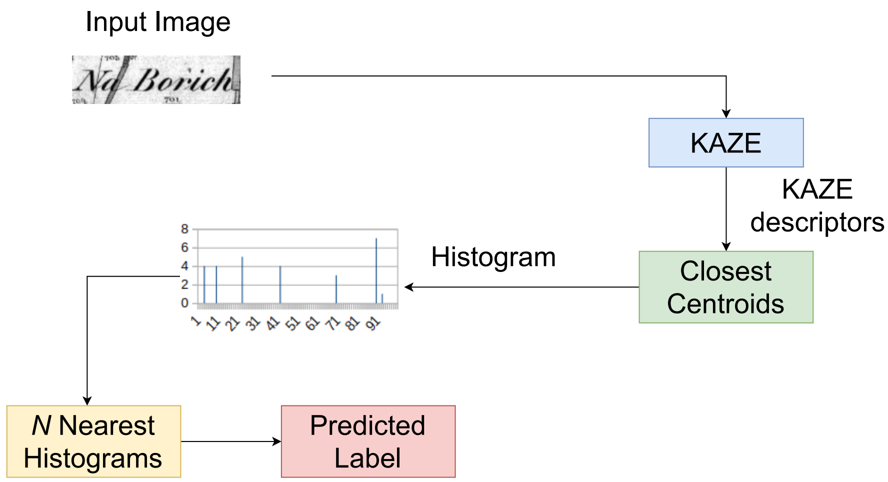

Fig. 6. Test image prediction process

4.3 OCR Models

Tesseract 4.0 Tesseract[4] [22] is a very popular and powerful open-source OCR engine. The important advantages are high recognition score and availability of many different language supports. It employs a powerful LSTM based OCR engine and integrates models for 116 additional languages. It is also possible to carry out training of the Tesseract engine and create a custom *tessdata* file which is fine-tuned on particular font and data.

5 Experiments

The experiments are divided into three sections. The first part deals with the toponym detection sub-task. The second experiment measures the success rate of toponym classification. The last set of experiments shows the results of OCR using Tesseract engine.

5.1 Toponym Detection

We evaluate the four different neural network models and compare it with the baseline algorithm based on connected components analysis, see Section 4.1. We use the evaluation protocol for the COCO dataset. We calculate precision, recall and F1 score for two intersection over union (IoU) levels (IoU above 0.5 and 0.75). Moreover, we calculate also average precision (AP, area under precision / recall curve). Finally, we provide the average AP at IoU interval 0.5 – 0.95 with 0.05 step. The overall text detection results are summarized in Table 2.

As we can see, the best results have been obtained by EAST Detector. The Faster R-CNN obtained also very good results. Its advantage compared to the EAST detector is the ability to predict also the label together with the bounding box.

[4] `https://github.com/tesseract-ocr/`

Table 2. Evaluation of the toponym detection algorithms on 0.5 and 0.75 IoU levels

Model	IoU@50				IoU@75				
	Prec.	**Rec.**	**F1**	**AP**	**Prec.**	**Rec.**	**F1**	**AP**	**Avg AP**
CCA (baseline)	19.5	60.4	29.5	11.3	10.7	33.1	16.2	0.27	2.78
EAST Detector	84.5	**89.9**	**87.1**	**77.8**	77.5	**82.4**	**79.9**	**51.3**	**46.7**
HP-FCN	65.4	75.4	70.1	44.4	53.9	62.2	57.8	17.1	20.6
YOLOv5	84.6	79.2	81.8	67.1	76.4	71.7	73.9	39.7	37.1
Faster R-CNN	**87.2**	80.9	83.9	71.2	**80.6**	75.0	77.7	45.4	41.8

Figure 7 extends the results reported in Table 2 and shows the precision of the utilized models based on the different IoU levels. This figure shows that the best precision is obtained by Faster R-CNN at all IoU levels. Moreover, EAST detector is only slightly worse.

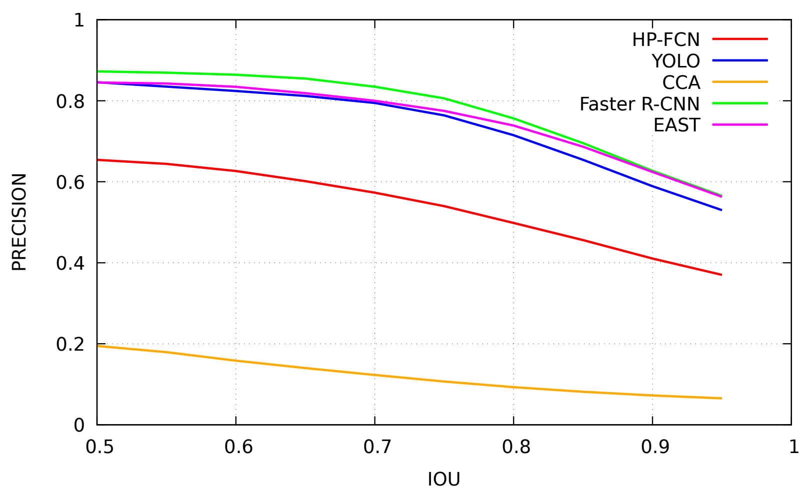

Fig. 7. Precision results based on IoU level

5.2 Toponym Classification

This experiment compares two approaches for toponym classification. The first one is the proposed toponym classification algorithm already described in Section 4.2. The classification is done for all toponym detection models. We use only correctly detected toponym regions for evaluation (IoU between predicted and ground truth regions at least 0.5). In such a case, we ran our algorithm and predicted a toponym label (P or H).

The second part of this experiment measures the quality of YOLOv5 and Faster-RCNN models and their built-in classification heads.

We show classification accuracies in Table 3. This table shows that using the proposed classification approach brings comparable results for all tested detection methods which indicates that it is robust against different sizes of the detected regions. It is also superior to the toponym classification accuracy of the YOLOv5 and Faster-RCNN models. The best results are obtained using HP-FCN together with our proposed classification algorithm.

Table 3. Toponym classification results; accuracy (ACC) in %

Detection Approach	**Classification Approach**	**ACC**
CCA (baseline)	Proposed	98.7%
EAST	Proposed	99.1%
HP-FCN	Proposed	**99.2**%
YOLOv5	Proposed	98.8%
Faster R-CNN	Proposed	98.8%
YOLOv5	YOLOv5	97.6 %
Faster R-CNN	Faster R-CNN	98.2 %

5.3 OCR Results

The last experiment measures the quality of the OCR using popular Tesseract engine with three different configurations. As a baseline OCR, we used Tesseract with standard ENG trained data that is available as a default configuration.

Since we have two types of toponyms that are visually different, we trained a separate Tesseract model for each of them. As a result, we created **Tesseract**$_P$ and **Tesseract**$_H$ for printed and hand-written toponyms respectively. During the evaluation step (on 346 bounding boxes in the test set), we measured the character error rate (CER). First, we split all test toponyms (bounding boxes) into printed and handwritten, and we evaluated each Tesseract configuration. Table 4 shows the results.

Table 4. OCR Results with Tesseract

	Printed Toponyms	**Handwritten Toponyms**	**All Toponyms**
Number of Toponyms	41	305	346
Tesseract ENG (baseline)	0.153	0.477	0.437
Tesseract$_P$ **(trained)**	**0.061**	0.512	0.459
Tesseract$_H$ **(trained)**	0.076	**0.185**	0.185
Combined Tesseract	–	–	**0.171**

Naturally, **Tesseract**$_P$ obtained excellent results for printed toponyms and **Tesseract**$_H$ was very successful in handwritten toponyms and vice versa. The best results on the full test set (combined printed and handwritten toponyms) were obtained by **Tesseract**$_H$.

Basically, the reason is that the vast majority of toponyms are handwritten. We also employed a combined Tesseract approach where we pick a Tesseract configuration according to the predicted class (e.g. if a bounding box of a toponym has been previously marked as P, **Tesseract**$_P$ is used for OCR and similarly for the label H). With such a configuration we obtained the best results (see the last row in Table 4) reaching CER value of 0.171. This result is sufficient for the target application.

6 Conclusions and Future Work

The main goal of this work was to detect, classify and recognize toponyms occurring in historical maps to enable an efficient information retrieval.

We have evaluated several text detection neural network models. We have conducted detection experiments on a newly published dataset and presented the results. In terms of the average precision, the best text detection model is EAST. Its drawback, though, is that it is not able to predict a toponym label per se.

We have also implemented a novel toponym classification algorithm that can be used for regions detected by arbitrary text detector. It is based on KAZE descriptor and the idea behind it comes from writer identification approaches. We have shown that this approach reaches excellent classification results. The drawback of training YOLOv5 and/or Faster R-CNN is the necessity of having a relatively high number of training samples. Our method does not require such a large number of samples and works very well even if only less than 200 toponym images (detected text-regions) are used for training.

Although Faster R-CNN and YOLOv5 obtained very good detection and classification results, the best strategy seems to be in separated training: 1) focus on the best possible text detection and 2) use a separate classification algorithm that can be easily adapted for classification into more classes. Finally, we have conducted the OCR experiments which are crucial for the success rate of the following IR system.

In the future work, we would like to concentrate on the further improvement of the processing pipeline, especially on the OCR part which is very important. We would also like to try a finer categorization of the toponyms (distinguish between rivers and hills etc.) We further plan to utilize our methods on map sheets from a different era and different geographical area.

Acknowledgements

This work has been partly supported from ERDF ”Research and Development of Intelligent Components of Advanced Technologies for the Pilsen Metropolitan Area (InteCom)” (no.: CZ.02.1.01/0.0/0.0/17_048/0007267) and by Grant No. SGS-2022-016 ”Advanced methods of data processing and analysis”.

References

1. Alcantarilla, P.F., Bartoli, A., Davison, A.J.: Kaze features. In: Computer Vision – ECCV 2012. pp. 214–227. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)

2. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9365–9374 (2019)
3. Chiang, Y.Y., Knoblock, C.A.: Recognizing text in raster maps. GeoInformatica **19**(1), 1–27 (2015)
4. De Campos, T.E., Babu, B.R., Varma, M., et al.: Character recognition in natural images. VISAPP (2) **7** (2009)
5. Epshtein, B., Ofek, E., Wexler, Y.: Detecting text in natural scenes with stroke width transform. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 2963–2970. IEEE (2010)
6. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (December 2015)
7. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
8. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
9. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE transactions on pattern analysis and machine intelligence **37**(9), 1904–1916 (2015)
10. Jung, K., Kim, K.I., Jain, A.K.: Text information extraction in images and video: a survey. Pattern recognition **37**(5), 977–997 (2004)
11. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR). pp. 1156–1160. IEEE (2015)
12. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: Icdar 2013 robust reading competition. In: 2013 12th International Conference on Document Analysis and Recognition. pp. 1484–1493. IEEE (2013)
13. Lenc, L., Prantl, M., Martínek, J., Král, P.: Border detection for seamless connection of historical cadastral maps. In: Barney Smith, E.H., Pal, U. (eds.) Document Analysis and Recognition – ICDAR 2021 Workshops. pp. 43–58. Springer International Publishing, Cham (2021)
14. Liang, J., Doermann, D., Li, H.: Camera-based analysis of text and documents: a survey. International Journal of Document Analysis and Recognition (IJDAR) **7**(2), 84–104 (2005)
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision **60**(2), 91–110 (Nov 2004). https://doi.org/10.1023/B:VISI.0000029664.99615.94, `http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94`
16. Lucas, S.M., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R., Ashida, K., Nagai, H., Okamoto, M., Yamamoto, H., et al.: Icdar 2003 robust reading competitions: entries, results, and future directions. International Journal of Document Analysis and Recognition (IJDAR) **7**(2-3), 105–122 (2005)
17. Mishra, A., Alahari, K., Jawahar, C.: Scene text recognition using higher order language priors. In: BMVC-British Machine Vision Conference. BMVA (2012)
18. Neumann, L., Matas, J.: A method for text localization and recognition in real-world images. In: Asian conference on computer vision. pp. 770–783. Springer (2010)
19. Neumann, L., Matas, J.: Real-time scene text localization and recognition. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3538–3545. IEEE (2012)

20. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
21. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28**, 91–99 (2015)
22. Smith, R.: An overview of the tesseract ocr engine. In: Ninth international conference on document analysis and recognition (ICDAR 2007). vol. 2, pp. 629–633. IEEE (2007)
23. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: 2011 International Conference on Computer Vision. pp. 1457–1464. IEEE (2011)
24. Weinman, J.: Geographic and style models for historical map alignment and toponym recognition. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 1, pp. 957–964. IEEE (2017)
25. Weinman, J., Chen, Z., Gafford, B., Gifford, N., Lamsal, A., Niehus-Staab, L.: Deep neural networks for text detection and recognition in historical maps. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 902–909. IEEE (2019)
26. Wick, C., Puppe, F.: Fully convolutional neural networks for page segmentation of historical document images. In: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS). pp. 287–292. IEEE (2018)
27. Xiong, Y.J., Wen, Y., Wang, P.S.P., Lu, Y.: Text-independent writer identification using sift descriptor and contour-directional feature. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR). pp. 91–95 (2015). https://doi.org/10.1109/ICDAR.2015.7333732
28. Yokobayashi, M., Wakahara, T.: Segmentation and recognition of characters in scene images using selective binarization in color space and gat correlation. In: Eighth International Conference on Document Analysis and Recognition (ICDAR'05). pp. 167–171. IEEE (2005)
29. Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J.: East: an efficient and accurate scene text detector. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 5551–5560 (2017)