How to Segment Handwritten Historical Chronicles Using Fully Convolutional Networks?

Josef Baloun[1,2][0000−0003−1923−5355], Pavel Král[1,2][0000−0002−3096−675X], and Ladislav Lenc[1,2][0000−0002−1066−7269]

[1] Department of Computer Science and Engineering, University of West Bohemia,
Univerzitní, Pilsen, Czech Republic
[2] NTIS - New Technologies for the Information Society, University of West Bohemia,
Univerzitní, Pilsen, Czech Republic
{balounj, pkral, llenc}@kiv.zcu.cz

Abstract. This paper deals with historical document image segmentation with focus on chronicles available in the Porta fontium portal. We build on our previously published database that has precise pixel-level annotations in PAGE format but also utilise other datasets for transfer learning in order to improve the results. We discuss a series of experiments that evaluate possibilities how to train a neural model for image, text and background segmentation. The outcome, in a form of segmentation method with relatively low computational costs and great results, is integrated into the Porta fontium portal to improve its possibilities of searching and publication of the documents.

Keywords: Page Segmentation · Chronicle · Historical Document · Text · Image · Background · Fully Convolutional Neural Network · Pixel Labeling · Artificial Page

1 Introduction

Archival documents such as old periodicals and chronicles are a valuable source of information. Preserve it and make it available for researchers and for general public is thus of a great importance. Current standard is to scan the materials and publish it in digital form through various portals and databases. However, scanning is only the first step in the digitisation process. Modern technologies allow to further process the document images and can provide many ways of intelligent search, classification and visualisation which is a great benefit for people working with the documents.

Project Porta fontium[3] is an example of efforts to provide the researchers and other interested persons with an efficient search in archival materials. It covers the Czech-Bavarian border area which has a common history before the World War II. After the war, the regions on both sides of the border where separated. It is thus a logical step to re-connect it and provide the related documents at one place.

[3] http://www.portafontium.cz/

In our work we concentrate on the development of efficient methods how to index and search in the vast data collections. A very important part of the processing pipeline is segmentation of the document images. Especially the issue of text localisation is crucial. Text segmentation has a long history dating back to the late 1970s when Optical Character Recognition (OCR) was addressed and it was necessary to extract single characters. "In order to let character recognition work, it is mandatory to apply layout analysis including page segmentation." [12] Today, most approaches tend to extract text lines instead of characters, but the importance of this step remains the same. There is also a need for extracting images that can be further processed and allow image search.

We focus on chronicles and their segmentation into text, image and background areas. These segments are crucial for further processing. For example OCR engines require a text input, but it could behave unpredictably when the input is an image or a graphic element. In such a case, the usability of the result could be harmed due to the produced noise. We can also provide image search or provide only pages that contain images.

To be able to train segmentation models that are usually based on deep learning neural architectures, we have to provide the model with a sufficient amount of training data. In our previous work [2], we have created a novel segmentation dataset that is designed for model training as well as for benchmarking purposes. We have utilised the dataset for initial experimentation with a Fully Convolutional Network (FCN) architecture and we have achieved promising results on real chronicle data. We have also presented an approach to automatically create artificial pages that can be used for data augmentation. In this paper, we go further and try to find efficient ways how to train a segmentation model with decent portion of data and evaluate several ways that can improve the model performance. Focusing mainly on the experiments and discussion, we provide more experiments on the input resolution, balancing the classes, post-processing the output and also examine the use of different training data including transfer learning and manual extension. Finally, we discuss the integration and usage on real data in the Porta fontium.

2 Related Work

This section first summarises recent methods for page segmentation and then it provides a short overview of available datasets.

2.1 Methods

There are many methods that were designed for the task of page segmentation which can be categorised into top-down and bottom-up categories. Historically, the segmentation problem was usually solved by conservative approaches based on simple image operations and on connected component analysis. Recent trend is to use neural networks for this task.

A page segmentation method using connected components and a bottom-up approach is presented in [5]. This method includes digitisation, rotation correction, segmentation and classification into text or graphics classes. Another method based on background thinning that is independent of page rotation is presented in [11]. These standard computer vision methods usually fail on handwritten document images, because it is difficult to binarise pages due to significantly low quality. It is also hard to extract characters since they are usually connected.

The above mentioned issues are successfully solved by approaches using Convolutional Neural Networks (CNN) that brought a significant improvement in many machine learning tasks including computer vision. An example of a CNN for historical document page segmentation is presented in [4]. Super-pixels (groups of pixels with similar characteristics) are identified in the image and they are classified using the network that takes 28×28 pixels as an input. The result of the classification is then assigned to the whole super-pixel.

Alternatively, every pixel can be classified separately using a sliding window. The problem of this approach is computational inefficiency because a large amount of computation is repeated as the window moves pixel by pixel. This problem is solved by Fully Convolutional Networks (FCNs) where one of the most efficient topology is U-Net [14]. This network was initially used for biomedical image segmentation but can be used in many other segmentation tasks including page segmentation.

Another architecture is presented by Wick & Puppe in [17]. This network is proposed for page segmentation of historical document images. In contrast to U-Net, it does not use skip-connections and uses transposed convolutional layer instead of up-sampling layer followed by convolutional layer. The speed improvement is achieved mainly due to the small input size (260×390 pixels).

In order to achieve the best results in competitions, there were proposed also networks like the one presented by Xu et al. in [18]. This network uses the original resolution of images and provides many more details in the output.

There are many architectures that solve the segmentation problem very well. However, the main issue of this task consists in the availability of appropriate training data because the relevant data is the key point of approaches based on neural networks.

2.2 Datasets

There are several datasets for a wide range of tasks. Unfortunately, a significant number of datasets are inappropriate for our task, because the documents differ significantly.

ChronSeg [2] dataset focuses on the segmentation of handwritten historical chronicles and it is available through website[4]. It contains training, validation,

[4] https://corpora.kiv.zcu.cz/segmentation/

testing and experimental parts containing totally 58 (double-sided) pages with precise ground-truth for text, image and graphic regions in PAGE format. There are five different chronicles present in the dataset.

Diva-hisdb [16] is a publicly available dataset with detailed ground-truth for text, comments and decorations. It consists of three manuscripts and 50 high-resolution pages for each manuscript. These manuscripts have similar layout features. The first two manuscripts come from the 11th century. They are written in Latin language using the Carolingian minuscule script. The third manuscript is from the 14th century and shows a chancery script in Italian and Latin. Unfortunately, the pages contain no images.

IAM-HistDB [7] contains handwritten historical manuscript images together with ground-truth for handwriting recognition systems. Currently, it includes three datasets: Saint Gall, Parzival and Washington.

Saint Gall database [6] contains 60 page images of a handwritten historical manuscript from 9th century. It is written in Latin language and Carolingian script.

Parzival database [8] is composed of 47 page images of handwritten historical manuscript from 13th century. The manuscript is written in Medieval German language and Gothic script.

Washington database [8] is created from the George Washington Papers. There are word and text line images with transcriptions. The provided ground-truth is not intended for page segmentation, but Saint Gall Database contains line locations that can be used for text segmentation.

Layout Analysis Dataset [1] is precisely annotated for page layout analysis and contains suitable regions for our task. The dataset contains a huge amount of page images of different document types. There is a mixture of simple and complex page layouts with varying font sizes. The problem is that the documents are printed and consist mostly of modern magazines and technical journals so the page layout is totally different to our chronicles in most cases.

Competition Datasets at PRImA Website There are also competition datasets at PRImA website[5]. These datasets has to be requested first and consist mainly of newspapers, books and typewritten notes. The number of annotated pages is usually around ten per dataset. Similarly as in the Layout Analysis Dataset, the text is printed and the page layout is different to our chronicle images.

[5] https://www.primaresearch.org/datasets

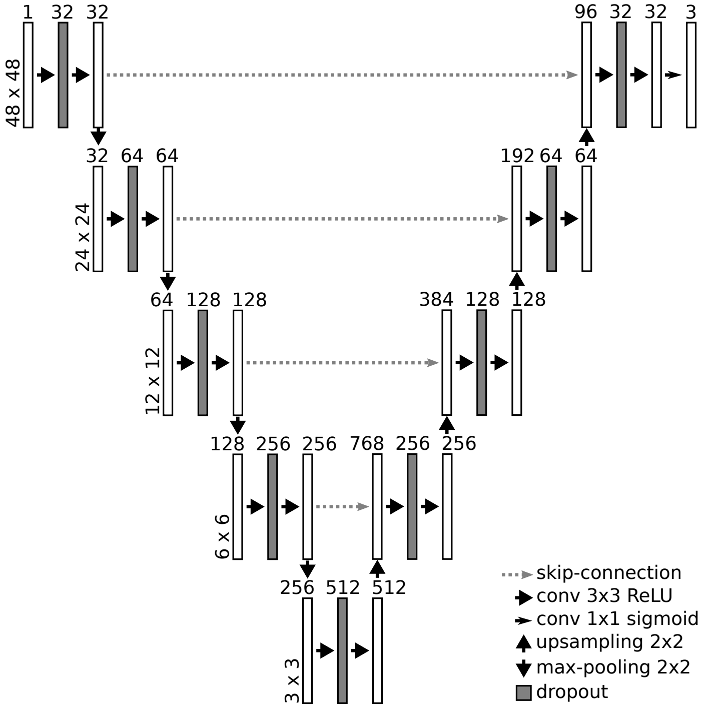

Fig. 1. FCN model architecture; values for48×48 input: Boxes represent feature maps (dimensions are denoted on the left side, the number of channels is indicated above the box) [2]

3 FCN Architecture

We utilise a model that is based on U-Net [14], see Figure 1 for the details . The architecture is designed to segment the entire input page at once. It uses padding in the convolutional layers, so there is no need for input image padding which could be problematic if the region with the padding colour is present in the image. Then this region could be understood by the network as a position at the borders and result in wrong predictions. There is usually a lot of noise at the borders of scanned document pages that should be suppressed. The padding in the convolutional layers allows that and also preserves dimensions so that the input resolution matches the output resolution.

Shared parameters in the convolutional layers allow variable input dimensions. In order to prevent skip-connection dimension inconsistency, the model input dimension has to be multiple of $2^4 = 16$ (given by four 2×2 max-pooling layers).

If there is a high-resolution input, the memory limitations appear. Then, there is again the need to trade-off between localisation accuracy and the use of context as discussed in [14]. The high resolution input can be processed in the sliding window manner using small context of the page or it can be down-sampled and processed with less details, bigger context but worse localisation accuracy. To reduce computational costs, the input image size is limited to 512×512 pixels. This setup has been identified based on our preliminary experiments and it is also supported by the work of Wick and Puppe [17] where the authors used input of 260×390 pixels.

ReLU activation function is used in all but the output layer where sigmoid is utilized in order to produce three binary (segmentation) masks. The Binary Cross-Entropy loss function is used to allow the classification of the pixel into more classes since there could appear regions that correspond to more classes (e.g. image overlaid with text).

4 Experimental Setup

According to the ChronSeg paper [2], we utilise the same input images and GT masks for the experiments. The dataset contains training, validation, testing and experimental parts. Thus the experiments are evaluated on the validation part and the combined setup also on the test part of the dataset.

For evaluation, the classification metrics *accuracy*, *precision*, *recall* and *F1 score* are used, since the task is a pixel-labelling problem. Each pixel is binary classified for each channel so that True Positive (TP), True Negative (TN), False Positive (FP) or False Negative (FN) sets can be identified. Further, the pixel modification of Intersection over Union (IoU) (see Equations 1) and Foreground Pixel Accuracy ($FgPA$) [17] are used. FgPA is practically an accuracy calculated only over foreground pixels that are estimated using binarisation [15] in this work. For the *combined* setup we calculate also the Panoptic metric [10] which handles semantic segmentation and instance segmentation. The Panoptic Quality (PQ), Segmentation Quality (SQ) and Recognition Quality (RQ) are obtained accordingly to [3].

$$IoU = \frac{TP}{TP + FP + FN} \tag{1}$$

If not stated differently in the experiment, the model is trained only on the training part containing 6 page images that contain pictures. As depicted in Figure 2, the grey level input image is first down-sampled to the target resolution,

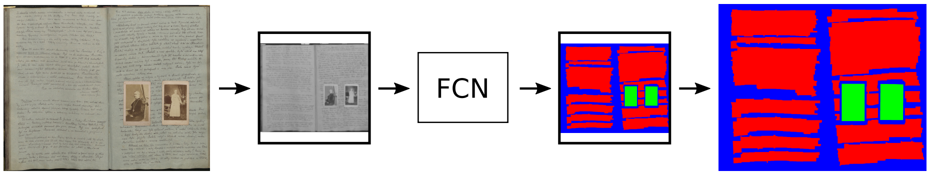

Fig. 2. Input image segmentation process: The input limit of 512×512 pixels is represented by squares before and after FCN box. [2]

predicted and then resized back. The target resolution is obtained as the nearest correction of the resolution that fits the 512×512 input and has the same aspect ratio as input. The input resolutions can be 512×400 and 512×416 for example.

For the training, dropout rate of 0.2, Adam optimiser [9] and the early stopping technique are used. The training is stopped if the average IoU on the validation part is not improving. The best model (highest IoU) is then used for evaluation.

5 Experimental Results

We have designed a set of experiments for techniques that are used to enhance the recognition results if only small amount of data are available. Namely, we experiment with extending the training data, transfer learning and loss function weighting. The automatic creation of artificial pages from the existing ones is also presented as a data augmentation approach that deals with the problem of class imbalances and brings significant improvements. We also evaluate the influence of input resolution and a post-processing step. The results are reported in Table 1 and compared to the *baseline* setup which represents the model trained only on the 6 pages of chronicles that contains the image. Based on the experiments the *combined* setup of the model is made.

Table 1. Average results (in %) of the experiments on the validation part: *Baseline* is a referential setup with 512×512 input limit and the model is trained only on 6 pages that contain images. Baseline setup modifications are presented in next three blocks (different input size, loss function weighting, training data). Based on the experiments, the *combined* setup is reported in the next block. The last block contains modifications to the combined setup using post-processing, transfer learning and extended training data. All the modifications are closely described in Sections 5 and 6.

	Accuracy	**Precision**	**Recall**	**F1 score**	**IoU**	**FgPA**
Baseline	95.3	91.8	92.6	92.0	85.5	98.5
128×128 input	86.6	79.9	82.7	80.7	68.0	93.4
256×256 input	93.9	89.9	91.8	90.7	83.1	98.4
1024×1024 input	95.5	94.1	91.6	92.6	86.5	98.8
Weighted sep. areas	95.3	94.6	90.7	92.3	85.9	99.0
Weighted classes	94.9	92.5	91.4	91.8	85.0	98.3
Augmentation	95.5	93.2	92.7	92.8	86.7	98.4
Artificial pages	96.1	94.0	94.3	94.0	88.9	99.2
Printed pages	95.5	94.2	92.1	93.0	87.1	98.8
Transfer learning	94.8	94.0	89.8	91.6	84.6	98.5
Combined	96.4	94.5	94.3	94.2	89.2	99.2
Post-process	95.9	93.4	94.6	93.9	88.6	99.0
Pre-trained	82.4	73.6	73.8	65.8	52.0	90.1
Fine-tuned	95.8	94.7	91.9	93.1	87.2	99.0
Extended	96.3	95.1	93.3	94.1	89.0	99.4

5.1 Input Resolution

The input resolution of the image is important since there are usually memory limitations and the compromise between computational costs, amount of details and available context for the prediction has to be made. Hypothetically, the neighbourhood of the pixel can be more important than local pixel details. Therefore the model is trained with 128, 256, 512 and 1024 input size limit. For a human, the limits lower than 512 results in images that are hard to read. On the other hand, 1024 limit is comfortable for reading the text. The limit of 512 is somewhere between.

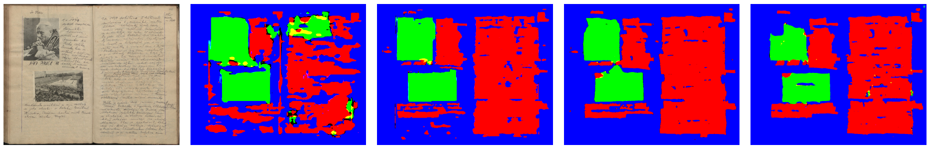

Fig. 3. Example predictions with different input limits (from left: input image, 128×128, 256×256, 512×512 and 1024×1024 input limits)

The results are reported in Table 1 and the example predictions are illustrated in Figure 3. The 128 limit seems too low and results in a lot of noise and a significant drop in IoU. The 256 limit is applicable but noise is still present. The limits of 512 and 1024 are visually comparable. For the 1024 limit, there is an improvement in the presented metrics at the cost of higher computational demands. With different setup using artificial pages and augmentation, this difference is vanishing as can be seen in Table 2.

Alternatively, the network can be trained on smaller crops (e.g. 512×512) and then the predictions can be made for the whole page at once thanks to the shared parameters. So no composing of sub-results is necessary. Together with sliding window approach, this scenario is not appropriate for the architectures that use padding in the convolutional layers, because the padding provides a lot of information for predictions in border areas. In such a case, the model tends to amplify the noise at the borders of predicted samples as presented in Figure 4. On the other hand, that information can be used for the noise suppression. For example, if there is mainly background class at the borders of the training samples, the model will more likely predict the border as background.

Table 2. Average results (in %) of different input limits with artificial pages and image augmentation setup [2]

	Accuracy	Precision	Recall	F1 score	IoU	FgPA
512×512	96.1	94.6	93.8	94.1	88.9	99.1
1024×1024	96.6	94.7	94.0	94.2	89.2	99.2

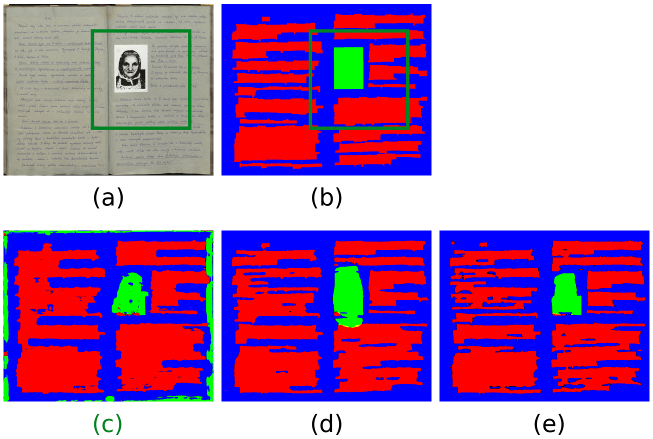

Fig. 4. Prediction examples: (a) input image, (b) ground-truth, (c) prediction of the model trained on crops (example training sample in green box of (a) and (b)), (d) *baseline* model prediction, (e) model prediction with weighted loss function for separating area [2]

5.2 Loss Function Weighting

Weighting of the loss function is used to improve the separation of the components as proposed in the U-Net paper [14]. The idea is that the component separating areas are more important for training thus the loss function has more weight as illustrated in Figure 5.

$$w(x) = w_0 \cdot \exp\left(-\frac{(d_1(x) + d_2(x))^2}{2\sigma^2}\right) \cdot (1 - \mathrm{gt}(x)) + 1 \qquad (2)$$

Weights are calculated for text and image channels according to Equation 2, where x is the pixel position, $d_1(x)$ and $d_2(x)$ stands for the distance to two nearest components. The ground-truth value of the pixel is denoted as $\mathrm{gt}(x)$. Parameter w_0 is set to 10 according to the U-Net paper and increased $\sigma = 10$ is used because of wider gaps between components.

The results of *weighted sep. areas* in Table 1 slightly improved. At the same time, the component separation is visually much better as illustrated in Figure 4.

The weighting of the loss function can be used also for weighting classes to deal with class imbalances. For the given channel of the training sample, the weight of the binary class is edited based on its area. For example, the image class has usually bigger weight than no-image class. The weights should be also limited, otherwise the high values can cause problems during training. The best achieved results with *weighted classes* setup is presented in Table 1 and does not lead to an improvement. It increases the amount of noise in the output and the training can be problematic. Thus it is not very useful, since the training samples seem already good balanced.

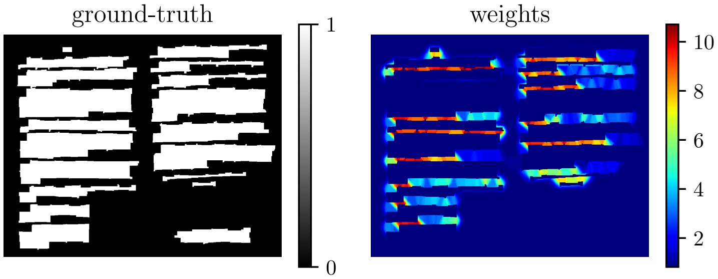

Fig. 5. Calculated weights for loss function weighting to improve the separation of the components. [2]

5.3 Training Data Extension

Even though the *baseline* setup trained on 6 pages achieved promising results, better results can be expected when providing more training data. The image augmentation is a good approach for automatic extension of training data. The same transformations are applied on the input image and corresponding ground-truth as illustrated in Figure 6. For the *augmentation* improvement in Table 1, the skew, slight rotations and grid based random distortions are used. As discussed previously, it is good to have the background class at the input borders to suppress noise. Since the grid based random distortion preserves borders, it is the most suitable.

The dataset contains also experimental part with annotated pages without images. These *no-image* pages are problematic for training because of class imbalances. On the other hand, they contain specialities like different writing styles and decorations that are useful for training. To be able to use them, the images are added randomly into a no-image page as depicted in Figure 7 with reason-

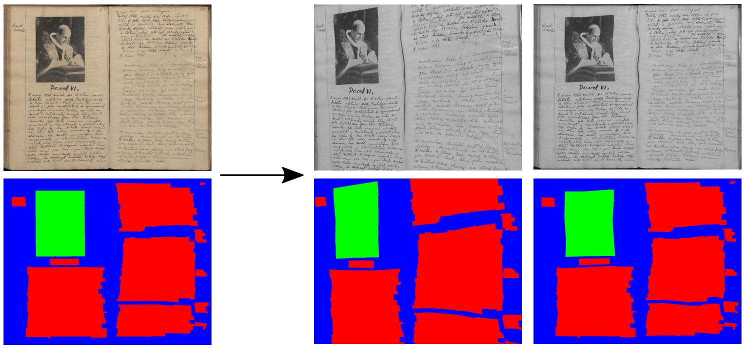

Fig. 6. From left: (1) input image and its ground-truth, (2) augmented image, (3) image augmented only with random_distortion. [2]

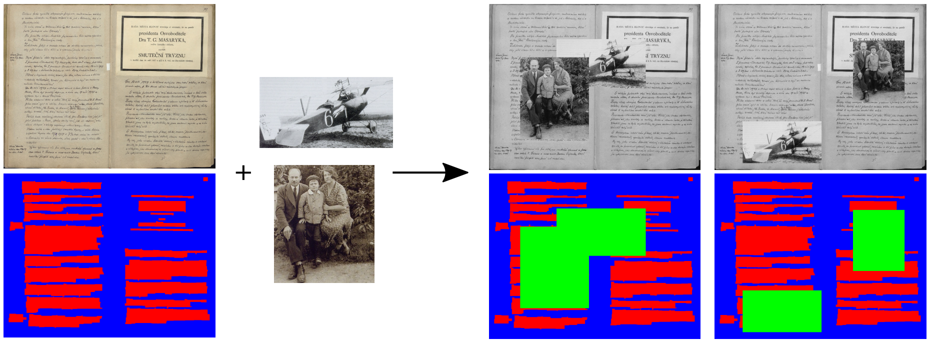

Fig. 7. Creation of artificial pages: Images are added randomly into document page [2]

able size and position restrictions. The image size ranges from 10×10 pixels up to 60 % of the page dimensions and the image can not touch the borders. These pages can be easily used for the training (see Figure 8). Table 1 shows remarkable improvement for generated *artificial pages*.

Printed pages setup experiments with training the model on the training set extended by printed documents from the experimental part. According to Table 1, the features learned from the printed pages can be useful and improve the results especially if there is small amount of training samples. On the other hand, this approach leads to predictions with more noise and does not work well if combined together with other approaches.

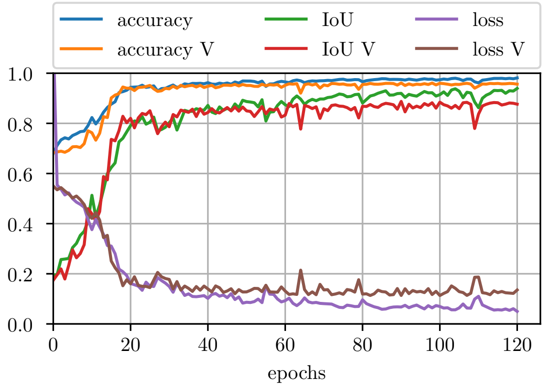

Fig. 8. Training process with using artificial pages for training does not indicate any problems caused by the generated samples. (V denotes validation set.) [2]

Table 3. Combined setup model evaluation on the test part of the dataset (in %)

	Accuracy	Precision	Recall	F1 score	IoU	FgPA	PQ	SQ	RQ
Text	96.3	95.8	92.1	93.8	88.4	98.7	51.6	80.2	63.8
Image	99.1	93.7	98.0	95.7	91.9	98.7	56.6	93.1	60.1
Background	96.1	96.5	96.4	96.4	93.1	99.0	22.9	93.0	24.3
Average	97.2	95.4	95.5	95.3	91.2	98.8	43.7	88.8	49.4

5.4 Combined Setup

The best results (see Table 1) were achieved with the *combined* training setup that weights the loss function for separating area ($w_0 = 5$ and $\sigma = 10$), artificial pages creation, grid based random distortions for image augmentation and input is limited to 512×512 pixels.

The results on the testing part of the dataset are provided in Table 3 and one example prediction can be seen in Figure 11.

5.5 Post-Processing

The idea for this experiment is to post-process the prediction to imitate the page text annotation process that is illustrated in Figure 9. To do that, the binarised image is masked with each separate component in the predicted text mask. This step imitates the noise removal and manual annotation. To shrink the region, the result is dilated with kernel k and the contour of the component is filled. After that, the erosion with the same kernel k is applied according to Figure 10. The setting of kernel k is problematic since the annotation is done manually and for each page different shrinking setup can be used. Experimentally, the $k = (38, 45)$ is used. The small components for text and image channels are discarded and finally, the background is edited.

Although the comparison of Figures 11 and 12 seems promising, the basic *post-process* does not improve the IoU results as could be seen in Table 1, since it tends to fill the separating area in some predicted regions (see Figure 10). On the other hand compared to the Table 3, the RQ improves significantly while SQ remains comparable resulting in better PQ as presented in Table 4.

Table 4. Post-processed combined setup model evaluation on the test part of the dataset (in %)

	Accuracy	Precision	Recall	F1 score	IoU	FgPA	PQ	SQ	RQ
Text	95.9	93.7	93.0	93.3	87.4	98.8	59.4	79.2	75.1
Image	99.1	93.8	97.9	95.6	91.8	99.0	80.1	93.1	85.8
Background	95.5	95.5	96.1	95.8	92.0	99.0	52.8	91.9	56.9
Average	96.8	94.3	95.7	94.9	90.4	98.9	64.1	88.1	72.6

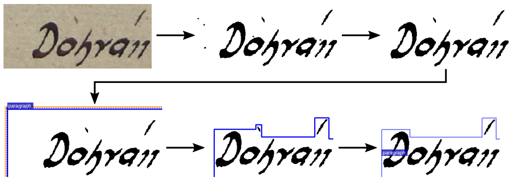

Fig. 9. The process of page annotation consists of image binarisation, noise removal, manual annotation and shrinking of the region. [2]

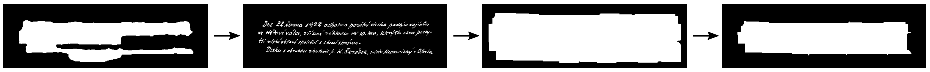

Fig. 10. Post-process example (from left: component from text mask prediction, masked binarised image, dilation, erosion)

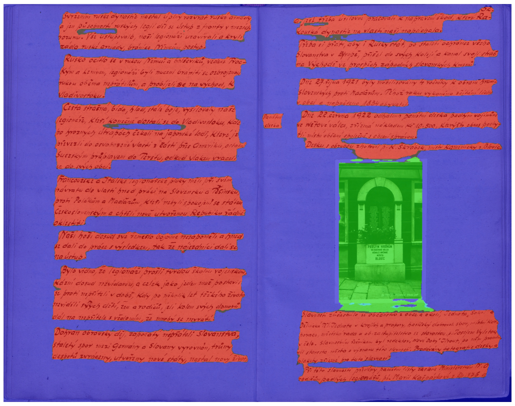

Fig. 11. Combined setup model example prediction of the page from test set [2]

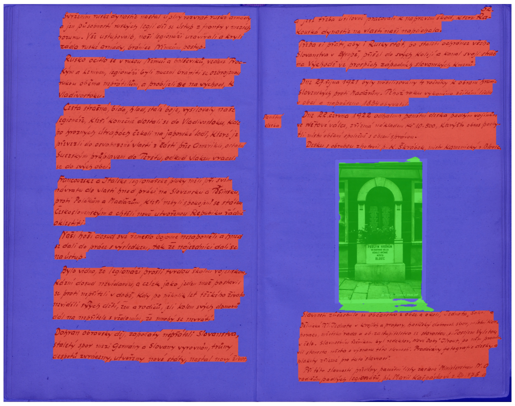

Fig. 12. Post-processed combined setup model prediction example

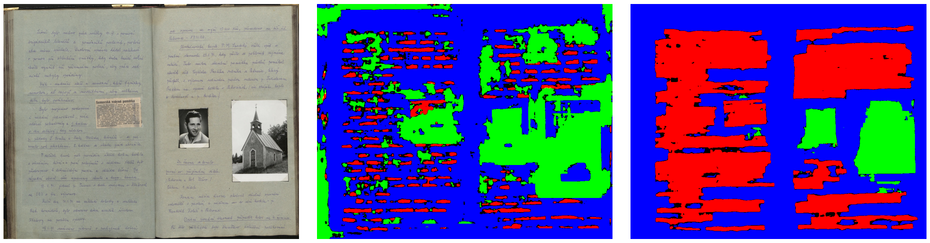

Fig. 13. From left: the input image, prediction of pre-trained model, prediction after fine-tuning (23 epochs)

5.6 Transfer Learning

Transfer learning in Table 1 is pre-trained on the printed documents from Layout Analysis Dataset and then fine-tuned as the *baseline* setup. The results are slightly worse and the model mispredicts the handwritten text as image more likely. On the other hand, it works better for the glued printed text blocks. This is probably due to the learned features for printed documents during pre-training. The fine-tuning is very fast and takes about 20 epochs compared to 160 epochs for the *baseline* setup. If the model is trained further for roughly the same number of epochs as *baseline*, the results are comparable.

The combined setup is also used for the transfer learning and results are reported for *pre-trained* model and *fine-tuned* model separately. For the fine-tuned model, the characteristics are the same as in the previous case. The predictions of pre-trained model are not usable as could be seen in Figure 13.

6 Porta Fontium Integration and Method Tuning

The model allows automatic text, image and background segmentation of the chronicles with relatively low computational costs. These days, the result allows to filter the chronicles or pages that contain images as illustrated in Figure 14. This can help the researchers studying the arts for example. It has also a great potential to further utilise the output for image search and handwritten text recognition to improve search options.

When testing the system on a large variety of different chronicles and thus previously unseen samples, the problems of mispredictions showed up. Using prediction masks, the two main problematic regions were identified. The first is image regions that were previously unseen in the training set. The second is text region with different writing style like decorative headings etc. The reason was determined as not enough training samples. There is a very limited set of images used for artificial pages during training. Also the decorative headings can be very diverse and not present among training samples.

To deal with this a huge amount of highly variable images from OpenImages [13] was used for artificial pages creation. The training set was also extended

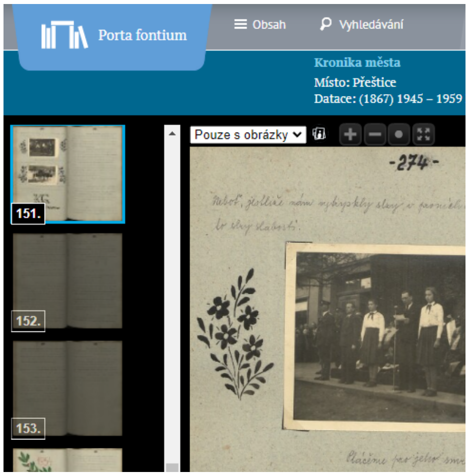

Fig. 14. Porta fontium extended functionality: The pages containing images are highlighted on the left side and can be browsed separately.

by newly annotated problematic pages containing decorative fonts, sketches etc. With *extended* data, a more general model is made while the results are still comparable to the specialised one as presented in Table 1. The results without post-processing presented in Table 5 are comparable to the combined setup results. The test part predictions are worse for the text but much better for the image class.

Table 5. Porta fontium model with extended data evaluation on the test part of the dataset (in %)

	Accuracy	Precision	Recall	F1 score	IoU	FgPA	PQ	SQ	RQ
Text	95.7	95.5	90.8	92.7	86.9	99.3	36.0	79.5	45.0
Image	99.2	96.0	98.0	96.9	94.1	99.3	63.1	94.0	66.2
Background	95.6	96.1	96.3	96.2	92.6	99.3	13.2	92.6	14.2
Average	96.8	95.9	95.0	95.3	91.2	99.3	37.5	88.7	41.8

7 Conclusions and Future Work

This paper presents an approach to segment historical handwritten chronicles into text, image and background classes together with a series of experiments. These experiments are very useful for the final model integrated into Porta fontium to improve the search options.

Based on the experiments, we can say that high resolution is not crucial for the chronicle segmentation into text, image and background. FCN model can generalise well on the documents that are similar but it is hard to create one generalised FCN model that can segment pages of different types and characteristics (e.g. modern printed magazines and historical handwritten documents). In such a case, the model tends to output more noise than the specialised one. This makes the real usage difficult. In that case, the transfer learning can help in creation of specialised models allowing fast fine-tuning. As shown in the experiments, a small amount of the data can be sufficient and the results can be further improved with data augmentation approaches. Also extending the dataset for verified segmented samples and the iterative training could help significantly and reduce the costs of manual annotations.

We plan further studying the possibilities to normalise the different types of document images that could allow the usage of one generalised model. The idea is to pre-process the image to normalise the pixel representation since it is very different in terms of pixel values from which the predictions are made.

Acknowledgments This work has been partly supported by Cross-border Cooperation Program Czech Republic - Free State of Bavaria ETS Objective 2014-2020(project no. 211) and by Grant No. SGS-2019-018 Processing of heterogeneous data and its specialized applications.

References

1. Antonacopoulos, A., Bridson, D., Papadopoulos, C., Pletschacher, S.: A realistic dataset for performance evaluation of document layout analysis. In: 2009 10th International Conference on Document Analysis and Recognition. pp. 296–300 (July 2009). https://doi.org/10.1109/ICDAR.2009.271
2. Baloun., J., Král., P., Lenc., L.: Chronseg: Novel dataset for segmentation of handwritten historical chronicles. In: Proceedings of the 13th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART,. pp. 314–322. INSTICC, SciTePress (2021). https://doi.org/10.5220/0010317203140322
3. Chazalon, J., Carlinet, E.: Revisiting the Coco panoptic metric to enable visual and qualitative analysis of historical map instance segmentation. In: Proceedings of the 16th International Conference on Document Analysis and Recognition (ICDAR'21). Lecture Notes in Computer Science, vol. 12824, pp. 367–382. Springer, Cham, Lausanne, Switzerland (Sep 2021). https://doi.org/10.1007/978-3-030-86337-1_25
4. Chen, K., Seuret, M., Hennebert, J., Ingold, R.: Convolutional neural networks for page segmentation of historical document images. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 1, pp. 965–970. IEEE (2017)
5. Drivas, D., Amin, A.: Page segmentation and classification utilising a bottom-up approach. In: Proceedings of 3rd International Conference on Document Analysis and Recognition. vol. 2, pp. 610–614 vol.2 (Aug 1995). https://doi.org/10.1109/ICDAR.1995.601970
6. Fischer, A., Frinken, V., Fornés, A., Bunke, H.: Transcription alignment of latin manuscripts using hidden markov models. In: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing. pp. 29–36 (2011)

7. Fischer, A., Indermühle, E., Bunke, H., Viehhauser, G., Stolz, M.: Ground truth creation for handwriting recognition in historical documents. In: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems. pp. 3–10 (2010)
8. Fischer, A., Keller, A., Frinken, V., Bunke, H.: Lexicon-free handwritten word spotting using character hmms. Pattern Recognition Letters **33**(7), 934–942 (2012)
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9404–9413 (2019)
11. Kise, K., Yanagida, O., Takamatsu, S.: Page segmentation based on thinning of background. In: Proceedings of 13th International Conference on Pattern Recognition. vol. 3, pp. 788–792 vol.3 (1996)
12. Kise, K.: Page Segmentation Techniques in Document Analysis, pp. 135–175. Springer London, London (2014). https://doi.org/10.1007/978-0-85729-859-1_5, https://doi.org/10.1007/978-0-85729-859-1_5
13. Krasin, I., Duerig, T., Alldrin, N., Ferrari, V., Abu-El-Haija, S., Kuznetsova, A., Rom, H., Uijlings, J., Popov, S., Kamali, S., Malloci, M., Pont-Tuset, J., Veit, A., Belongie, S., Gomes, V., Gupta, A., Sun, C., Chechik, G., Cai, D., Feng, Z., Narayanan, D., Murphy, K.: Openimages: A public dataset for large-scale multi-label and multi-class image classification. Dataset available from https://storage.googleapis.com/openimages/web/index.html (2017)
14. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. pp. 234–241. Springer International Publishing, Cham (2015)
15. Sauvola, J., Pietikinen, M.: Adaptive document image binarization. Pattern Recognition **33**(2), 225 – 236 (2000). https://doi.org/https://doi.org/10.1016/S0031-3203(99)00055-2, http://www.sciencedirect.com/science/article/pii/S0031320399000552
16. Simistira, F., Seuret, M., Eichenberger, N., Garz, A., Liwicki, M., Ingold, R.: Diva-hisdb: A precisely annotated large dataset of challenging medieval manuscripts. In: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 471–476. IEEE (2016)
17. 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 (April 2018). https://doi.org/10.1109/DAS.2018.39
18. Xu, Y., He, W., Yin, F., Liu, C.L.: Page segmentation for historical handwritten documents using fully convolutional networks. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 1, pp. 541–546. IEEE (2017)