

Page Segmentation for Historical Handwritten Documents Using Fully Convolutional Networks

Yue Xu^{1,2}, Wenhao He^{1,2}, Fei Yin¹, Cheng-Lin Liu^{1,2}

¹ National Laboratory of Pattern Recognition, Institute of Automation of Chinese Academy of Sciences, 95 Zhongguancun East Road, Beijing 100190, P.R. China

² University of Chinese Academy of Sciences, Beijing, P.R. China

Email: {yue.xu, wenhao.he, fyin, liucl}@nlpr.ia.ac.cn

Abstract—Page segmentation is a fundamental and challenging task in document image analysis due to the layout diversity. In this work, we propose a pixel-wise segmentation method for historical handwritten documents using fully convolutional network (FCN). The document image is segmented into different regions by classifying pixels into different categories: background, main text body, comments, and decorations. By supervised learning on document images with pixel-wise labels, the FCN can extract discriminative features and perform pixel-wise segmentation accurately. After pixel-wise classification, post-processing steps are taken to reduce noises, correct wrong segmentations and find out overlapping regions. Experimental results on the public dataset DIVA-HisDB containing challenging medieval manuscripts demonstrate the effectiveness and superiority of the proposed method, which yields pixel-level accuracy of above 99%.

Keywords—page segmentation; layout analysis; fully convolutional network;

I. INTRODUCTION

Page segmentation is a fundamental task in document image analysis and understanding. It segments the document image into different regions with uniform elements: background, texts, graphics, half-tones, decorations, etc. Page segmentation is challenging due to the large variability of layout mixing different elements in various formats, and the degradation of document images, particularly, historical documents. Fig. 1 shows a historical document image and its labeling of different regions. It is hard to segment such images using projection-based or rule-based methods.

Previous methods on page segmentation can be broadly divided into three categories: granular-based, block-based and texture-based [1]. In granular-based methods [2]–[5], pixels or connected components are considered as basic elements and merged into larger homogeneous regions. In block-based methods [6]–[8], page images are cut into small areas, and these areas are then merged or split to produce homogeneous regions. In texture-based methods [9]–[12], texture features are extracted by low-level filters and then classified into different contents by statistical models. Block-based methods fall in the top-down approach, while the other two methods adopt a bottom-up pipeline. Bottom-up approaches are superior to segment documents of irregular layout, such as non-horizontal text lines, mixing or overlapping elements. However, they are more computationally demanding because of the large number

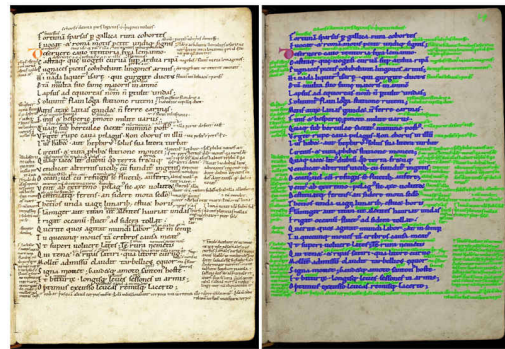


Fig. 1. Page segmentation sample. The left is the original image. The right is the segmentation ground truth. The colors: blue, green, and red are used to represent main text body, comments, and decorations respectively.

of small elements like pixels and connected components to classify. The extraction of features and design of classifier are both critical to the segmentation performance.

The recently proposed fully convolutional network (FCN) [13] has made great improvements in generic object segmentation area and this provides us a new perspective to deal with page segmentation. The FCN is an end-to-end training-testing segmentation framework, and completes feature extraction and classifier optimization simultaneously. However, to the best of our knowledge, FCN has not been applied for page segmentation problem before.

In our work, we present a FCN based framework for page segmentation of historical handwritten documents. To produce a pixel-level segmentation result as shown in Fig. 2, our framework first trains a FCN to predict the class of each pixel in document images. Pixels are classified into four classes: background, main text body, comments, and decorations. Then, coarse segmentation results are refined by reducing noises and correcting wrong segmentations through analysing connected components. Since decorations may overlap main text body or comments [14], i.e., a pixel can be both decoration and main text body/comment, finally we identify the overlap areas by analysing their sizes and surroundings.

The contributions of our work are mainly in three folds: 1) it is the first FCN based framework for page segmentation of historical handwritten document. 2) we provide a complete

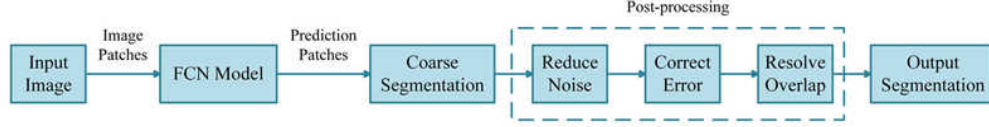


Fig. 2. Framework of the proposed method.

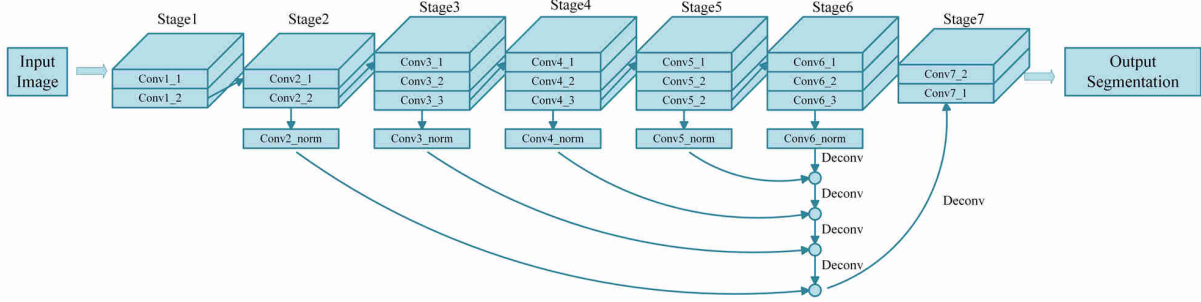


Fig. 3. Structure of the FCN model.

page segmentation framework including the strategies for large images and post refinement. 3) we achieve the state-of-the-art performance on the public page segmentation benchmark, indicating the effectiveness of our method.

The remainder is organized as follows. Section II reviews related works on page segmentation for historical handwritten document. Section III describes the details of the proposed method. Section IV presents our experimental results and analysis. Finally, concluding remarks are given in Section V.

II. RELATED WORKS

Page segmentation methods can be broadly divided into three categories: granular-based, block-based and texture-based [1].

Granular-based methods. These methods start from the smallest elements of the page, e.g., pixels or connected components, and try to agglomerate them into larger homogeneous regions. Graz et al. [4] introduced a layout analysis method for historical handwritten document. The method used the SIFT feature and suggested a part-based detection of layout entities locally. It then used a SVM for region classification. Bukhari et al. [5] introduced an approach that segments text appearing in page margins. The method extracted features of connected component level, and trained a multi-layer perception to classify connected components. A voting scheme is then applied to refine the result and produce the final classification.

Block-based methods. These methods cut page images into small regions and then merge/split them until yielding homogeneous regions. Chen et al. [15] proposed a method to segment text in complex document images. The method cut the document into blocks which are then multi-thresholded to create several layers. Then the connected components of

each layer are identified and grouped across blocks based on a predefined set of features. Ouwayed et al. [16] proposed a method to analyze the layout of historical handwritten document with multi-oriented text lines. The proposed method made clear assumptions about the document layout. It uses image meshing and projection profiles to determine the text lines progressively and locally.

Texture-based methods. These methods extract the texture features of the page images and then classify the different contents by employing statistical models. Asi et al. [1] proposed a robust framework for analysing historical manuscripts. The method segmented text into spatially coherent regions and text-lines using texture-based filters and refined this segmentation by exploiting Markov Random Fields (MRFs). Mehri et al. [17] proposed a texture based segmentation method for historical document. The method extracted textual features, used the simple linear iterative clustering (SLIC) superpixels, Gabor descriptors and classified pixels into foreground and background by using SVM.

The recent fully convolutional network (FCN) [13] extracts both low-level and high-level feature by convolutional filters and learns segmentation by an end-to-end way. The FCN also belongs to texture-based methods but has much more capability to deal with page segmentation of historical handwritten documents.

III. PROPOSED METHOD

In this section, firstly, we introduce the FCN based network for page segmentation and some crucial modifications from previous works. Secondly, we explain the edge effect in testing stage and our solution as well. Finally, steps of the post-processing like noise removal, small region correction and

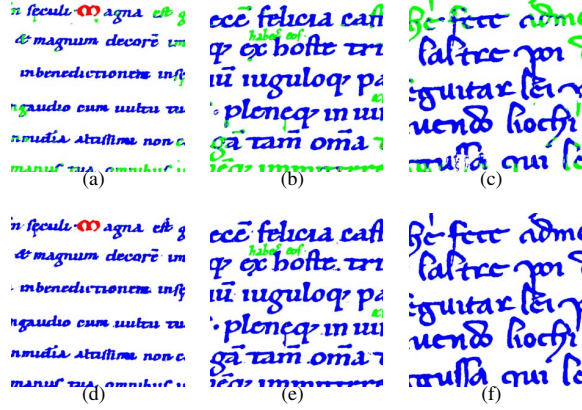


Fig. 4. (a)(b)(c) segmentation results for testing patches. (d)(e)(f) the ground truth for testing patches.

overlap refinement are illustrated.

A. Network Structure

The page segmentation network (see Fig. 3) is based on the FCN for object segmentation. The first 5 convolutional stages follow the design of VGG 16-layer net [18]. However, there are two modifications from previous works. Firstly, unlike segmentation for generic objects, page segmentation requires more accurate partitions on strokes. Consequently, we combine more low-level features (Stage 2 in Fig. 3) which could provide more information of details. Secondly, the maximum receptive field of VGG 16-layer net is around 224, which may not contain enough context if the character and line space is large. To understand the context better, we design a deeper network with larger receptive field by adding three additional 3×3 convolutional layers to the top of the Stage 5 in VGG 16-layer net.

Before feature fusion and deconvolution, we normalize the channel size of each stage to 128 with a 1×1 kernel. All the deconvolution layers have a 2×2 kernel with the stride 2. The output Stage 7 resizes the output channel to 4 which is the same as the target category amount. And we employ the *Softmax Loss* for the pixel-wise optimization.

B. Elimination for Edge Effect

Limited by the memory of GPU, the whole image can not be trained or tested directly. Therefore, we crop images into small patches. During the training stage, the input images are randomly cropped from the origin training images. All the cropped patches follow the same size of 320×320 . In this work, we generate about 120,000 training patches.

Discriminative features such as character size, stroke shape and context information are important for prediction. However, during the test stage, characters at the boundary of the patch images could be cut off and vital features would be lost a lot, which brings the edge effect. Pixels at the boundary with incomplete features could be easily misclassified and the FCN would perform the noisy output (see Fig. 4). To eliminate the edge effect, we crop the testing images into 640×640 patches

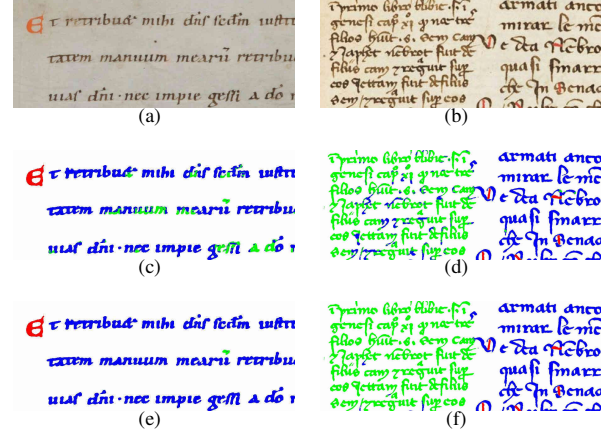


Fig. 5. (a)(b) original images. (c)(d) initial result with wrong segmentation. (e)(f) segmentation result after error correction. The colors: white, blue, green, and red represent background, main text body, comments, and decorations respectively.

with stride of 480 in a sliding window manner. And then we only use the result of the center 480×480 region and abandon the border area whose size is corresponding with the average character size.

C. Post-processing

Note that there are noises and misclassified components in the segmentation results provided by the FCN model (see Fig. 5(c) and Fig. 5(d)). To refine the prediction, we perform the post-processing with three steps.

1) *Reduce noises*: Firstly, connected components (CCs) in each foreground category (main text body, comments, and decorations) region will be extracted and then small isolated CCs whose pixel number is less than 10 are set to be background.

2) *Correct wrong segmentation*: To correct the misclassified regions, we make the following assumptions. First, the class label of small isolated regions tends to be the same as that of its surroundings. Second, the length of contacted boundary between main text body and comments is short. Suppose C_a and C_b are the adjacent CCs belonging to different classes (here we only consider main text body and comment), and their categories are predicted to be A and B respectively. L_a is the boundary length of C_a , and L_{ab} is the length of contacted boundary between C_a and C_b . If $L_{ab} \geq L_a/3$, C_a will be considered as an isolate CC surrounded by Class B. Then, pixels in 320×320 window will be counted. N_a is the pixel number of class A and N_b is the pixel number of class B . If $N_b > N_a$, A will be corrected to B (see Fig. 5). And vice versa.

3) *Refine overlapping regions*: Since decorations may be embedded in the main text body or comments, these embedded decorations will be further analyzed to extract overlapping regions. Let H_m be the mean height of the main text body CCs, h be the height of a decoration CC. A decoration CC will be identified as pure decoration if it is large enough

TABLE I
THREE TYPES OF MEDIEVAL MANUSCRIPTS.

Manuscripts Number	Training (pages)	Validation (pages)	Testing (pages)	Page size (pixels)
CSG18	20	10	10	3328 × 4992
CSG863	20	10	10	3328 × 4992
CB55	20	10	10	4872 × 6496

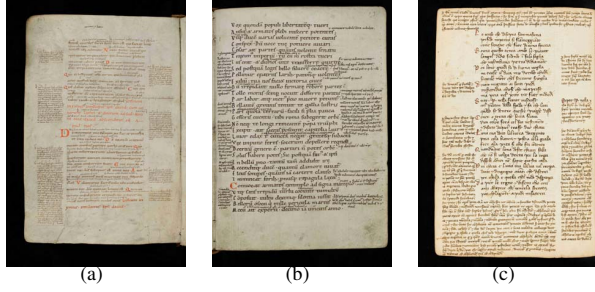


Fig. 6. Samples pages of the three types of medieval manuscripts in DIVA-HisDB

(for this work, $h > 4 \times H_m$). Otherwise it will be treated as overlapping region. Since the size of the main text body is usually larger than that of comments, a CC will be considered as the overlapping region between decoration and comment if it is small enough (for this work, $h < H_m$) and surrounded by the overlapping region between decoration and main text body.

The equation below shows the refinement for decoration regions. In Eq. (1), DE, MT, CM represent *decorations*, *main text body*, and *comments* respectively. And *srb.* is short for *surrounded by*.

$$CC_{class} = \begin{cases} DE, & h > 4 \times H_m, \\ DE \& CM, & h < H_m \text{ and } srb. CM, \\ DE \& MT, & \text{Otherwise.} \end{cases} \quad (1)$$

IV. EXPERIMENTS

A. Dataset

The proposed method is tested on DIVA-HisDB dataset [14]. DIVA-HisDB is a historical manuscript dataset that consists of three types of medieval manuscripts with complex layout elements, diverse scripts, and challenging degradations (see Fig. 6). There are total 120 annotated pages, including 60 images for training, 30 images for validation, and 30 images for testing (see Table I).

In this dataset, pixels are divided into four categories: background, main text body, comments (marginal and interlinear glosses, explanations, corrections) and decorations (characters/signs that exceed the size of a text line and written in red). It is worth noting that the dataset defines special boundaries, which surround the foreground in polygon form as shown in Fig. 7. Background pixels within the special boundaries can be classified as either foreground or background.

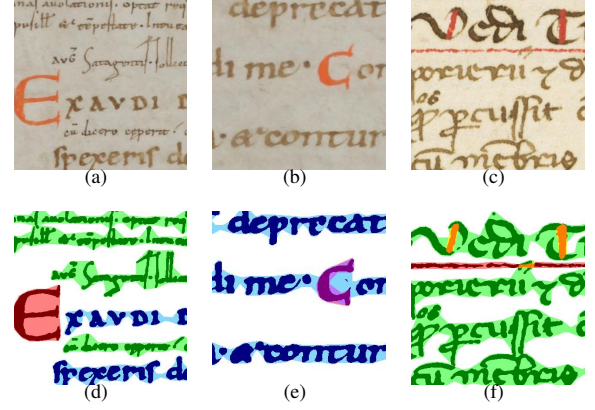


Fig. 7. Ground truth samples. (a)(d) show the main text body (blue) and its boundaries (light blue), comments (green) and its boundaries (light green), decorations (red) and its boundaries (light red). (b)(e) show decorations covering main text body (purple) and its boundaries (light purple). (c)(f) show decorations covering comments (orange) and its boundaries (light orange).



Fig. 8. Training image and label. (a) shows the original training image. (b) shows the label. The colors: white, blue, green, and red represent background, main text body, comments, and decorations respectively.

The pixels belonging to different classes are set with different RGB value. There are four different classes: main text body (RGB=0x000008), decorations (RGB=0x000004), comments (RGB=0x000002) and background (RGB=0x000001). For each multi-label pixel (overlapping regions), its value will be the sum of the corresponding classes. For example, 0x000006 means the pixel belongs to both comment and decoration, 0x00000C means the pixel belongs to both main text body and decorations.

It is worth noting that in the process of constructing the training set, pixels in overlapping regions keep only one label. For example, pixels belong to both main text body/comments and decorations are labeled as decoration. Fig. 8 shows a training image and its label.

B. Implementation Details

The network is optimized by stochastic gradient descent (SGD) with back-propagation and the maximum iteration is



Fig. 9. Segmentation results of pages in Fig. 6(a), 6(b), and 6(c). The colors: white, blue, green, red, purple and orange represent background, main text body, comments, decorations, overlap regions of main text body and decorations, overlap regions of comments and decorations respectively.

100,000. The learning rate is fixed to be 0.01 for the first 30,000 iterations and then degraded to 0.001 until the end of training. For the initialization of the network, the first 5 stages copy the parameters from the VGG 16-layer net and the rest layers are all initialized by “xavier” [19]. Weight decay is 4×10^{-4} and momentum is 0.9. The whole experiments are conducted on Caffe [20] and run on a workstation with 2.9GHz, 12-core CPU, 256G RAM GTX Titan X and Ubuntu 64-bit OS.

C. Experimental Results

Fig. 9 shows the final segmentation results of some sample images after post-processing. The performance of our method is evaluated by *Precision*, *Recall*, *Accuracy* and *F1 – Measure* on pixel level [5]. For each pixel, the proposed model will produce a set of prediction represented by a four-dimensional vector: (0, 0, 0, 1) means background (label=0), (0, 0, 1, 0) means comments (label=1), (0, 1, 0, 0) means decorations (label=2), and (1, 0, 0, 0) means main text body (label=3). The value of multi-class pixel will be the sum of the corresponding classes. For example, (0, 1, 1, 0) means both comment and decoration, (1, 1, 0, 0) means both main text body and decoration.

For each class i ($i = 0 \dots 3$), first, we count TP_i (True-Positive), FP_i (False-Positive), TN_i (True-Negative) and FN_i (False-Negative) within the whole image. Then, we calculate $TP_{total} = \sum TP_i$, $FP_{total} = \sum FP_i$, $TN_{total} = \sum TN_i$, $FN_{total} = \sum FN_i$. *Precision*, *Recall*, *Accuracy* and *F1 – Measure* for each class and total results are computed according to Eq. (2).

$$Precision = \frac{TP}{TP + FP} \quad (2a)$$

$$Recall = \frac{TP}{TP + FN} \quad (2b)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2c)$$

$$F1-Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (2d)$$

A baseline method based on Convolutional Auto-Encoder (CAE) [21] achieved around 95% accuracy. As shown in Table

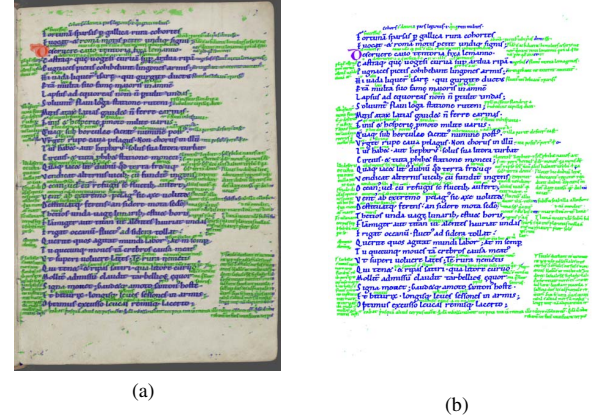


Fig. 10. Segmentation results of different methods for Fig. 1. (a) CAE [14]. (b) the proposed method.

II, the proposed method achieves the state-of-the-art performance with 99.62%, 99.61%, 99.81%, 99.62% for *Precision*, *Recall*, *Accuracy* and *F1 – Measure* respectively, which surpass the baseline method with a large margin. For the evaluation metrics above, the post processing improve 1.85%, 3.15%, 2.51%, 0.16% in main text body, 0.9%, 1.29%, 0.71%, 0.18% in comments. Since the performance without post-processing is already competitive, the improvements after post-processing should be convincing. Fig. 10 shows the segmentation results produced by the CAE method and the proposed method for the same image, indicating that the proposed methods produces more precise segmentation and less noise.

The reason why we outperform the CAE model a lot is mainly due to the multi-level fusion design in FCN. In our FCN segmentation part, features of different levels own multiple receptive field sizes, and multi-level fusion design ensures our model to see both global and local information, which produces precise classification and smooth segmentation respectively. While, in [21], the convolutional encoder part employs a shallow convolutional part to extract features, and thus it could lose much global information and produce a relatively lower accuracy.

D. Fine-Tuning Effects

In this section, we investigate the effect of fine-tuning, which is a custom training strategy. In our work, we follow the design of VGG 16 network for the first 5 convolutional stages, and we compare the loss convergence speed and accuracy of the test set between the fine-tuning training and general training (random initialization). The comparison is shown in Fig. 11, from which we could see the fine-tuning strategy gives a obvious faster convergence rate and slightly higher accuracy. The final accuracy given by fine-tuning and general training does not differ much owing to the sufficient training samples.

V. CONCLUSION

In this paper we present a pixel-wise page segmentation method for historical handwritten document using fully convolutional network (FCN). The whole framework contains two

TABLE II
PERFORMANCE OF DIFFERENT METHODS EVALUATED ON DIVA-HISDB DATASET.

	Metric	Background	Main text body	Comments	Decorations	Total
Proposed	Precision	99.76%	99.03%	98.40%	95.37%	99.62%
	Recall	99.95%	97.80%	96.07%	92.74%	99.61%
	F1-Measure	99.85%	98.41%	97.22%	94.04%	99.62%
	Accuracy	99.74%	99.80%	99.76%	99.98%	99.81%
Baseline	Accuracy	92.54%	96.24%	95.29%	98.02%	95.52%

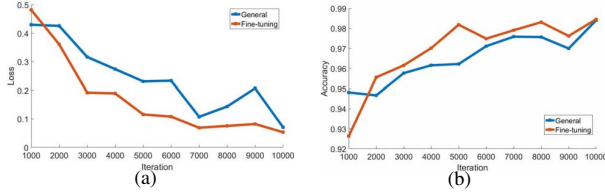


Fig. 11. Comparison between fine-tuning process (red line) and general process (blue line). (a) The curve of loss. (b) The curve of accuracy.

parts. First we use a FCN based network to classify each pixel of document images into four categories: background, main text body, comments, and decorations. And after pixel-wise segmentation, we then adopt some heuristic based post-processing to reduce noise, correct misclassified pixels and find out overlapping regions. The robust and discriminative features extracted by the FCN based network ensure the high accuracy of pixel-wise classification. On the DIVA-HisDB page segmentation benchmark, we have achieved a new state-of-the-art performance and outperformed the baseline method by a large margin.

Future works aim to design more accurate and efficient segmentation framework, as well as a more fair evaluation metric on the unbalanced segmentation classes.

VI. ACKNOWLEDGMENTS

This work has been supported by the National Natural Science Foundation of China (NSFC) grants 61573355.

REFERENCES

- [1] A. Asi, R. Cohen, K. Kedem, and J. Elsana, "Simplifying the reading of historical manuscripts," in *Proceedings of the 13th International Conference of Document Analysis and Recognition*, pp. 826–830, 2015.
- [2] M. Agrawal and D. Doermann, "Voronoi++: A dynamic page segmentation approach based on voronoi and docstrum features," in *Proceedings of the 10th International Conference on Document Analysis and Recognition*, pp. 1011–1015, 2009.
- [3] M. Baechler and R. Ingold, "Multi resolution layout analysis of medieval manuscripts using dynamic mlp," in *Proceedings of the 11th International Conference on Document Analysis and Recognition*, pp. 1185–1189, 2011.
- [4] A. Garz, R. Sablatnig, and M. Diem, "Layout analysis for historical manuscripts using sift features," in *Proceedings of the 11th International Conference on Document Analysis and Recognition*, pp. 508–512, 2011.
- [5] S. S. Bukhari, T. M. Breuel, A. Asi, and J. El-Sana, "Layout analysis for arabic historical document images using machine learning," in *Proceedings of the 13th International Conference on Frontiers in Handwriting Recognition*, pp. 639–644, 2012.
- [6] G. Nagy, M. Viswanathan, and S. Seth, "A prototype document image analysis system for technical journals," in *Proceedings of the IEEE*, vol. 25, pp. 10–22, 1992.
- [7] S. Utama, J. M. Ogier, and P. Loonis, "Top-down segmentation of ancient graphical drop caps: lettrines," in *Proceedings of the International Workshop on Graphics Recognition*, pp. 87–96, 2005.
- [8] N. Ouwayed and A. Belad, "Multi-oriented text line extraction from handwritten arabic documents," in *Proceedings of the Eighth International Workshop on Document Analysis Systems*, pp. 339–346, 2008.
- [9] N. Journet, J. Y. Ramel, R. Mullot, and V. Eglin, "Document image characterization using a multiresolution analysis of the texture: application to old documents," in *Proceedings of the International Journal on Document Analysis and Recognition*, vol. 11, pp. 9–18, 2008.
- [10] R. Cohen, A. Asi, K. Kedem, J. El-Sana, and I. Dinstein, "Robust text and drawing segmentation algorithm for historical documents," in *Proceedings of the International Workshop on Historical Document Imaging and Processing*, pp. 110–117, 2013.
- [11] A. Asi, R. Cohen, K. Kedem, and J. El-Sana, "A coarse-to-fine approach for layout analysis of ancient manuscripts," in *Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition*, pp. 140–145, 2014.
- [12] K. Chen, H. Wei, J. Hennebert, and R. Ingold, "Page segmentation for historical handwritten document images using color and texture features," in *Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition*, pp. 488–493, 2014.
- [13] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.
- [14] F. Simistira, M. Seuret, N. Eichenberger, A. Garz, M. Liwicki, and R. Ingold, "Diva-hisdb: A precisely annotated large dataset of challenging medieval manuscripts," in *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition*, pp. 471–476, 2016.
- [15] Y. L. Chen and B. F. Wu, "A multi-plane approach for text segmentation of complex document images," in *Proceedings of the Pattern Recognition*, vol. 42, pp. 1419–1444, 2009.
- [16] N. Ouwayed and A. Belad, "A general approach for multi-oriented text line extraction of handwritten documents," in *Proceedings of the International Journal on Document Analysis and Recognition*, vol. 15, pp. 1–18, 2012.
- [17] M. Mehri, N. Nayef, P. Roux, Gomez-Kr, P. Mer, and R. Mullot, "Learning texture features for enhancement and segmentation of historical document images," in *Proceedings of the International Workshop on Historical Document Imaging and Processing*, pp. 47–54, 2015.
- [18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the International Conference on Learning Representations*, 2015.
- [19] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, vol. 9, pp. 249–256, 2010.
- [20] Jia, Yangqing, Shelhamer, Evan, Donahue, Jeff, Karayev, Sergey, Long, and Jonathan, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 675–678, 2014.
- [21] M. Seuret, R. Ingold, and M. Liwicki, "N-light-n: A highly-adaptable java library for document analysis with convolutional auto-encoders and related architectures," in *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition*, pp. 459–464, 2016.