Document Layout Analysis with an Enhanced Object Detector

Mohammad Minouei*†, Mohammad Reza Soheili‡, Didier Stricker*†
*TU Kaiserslautern, †German Research Center for Artificial Intelligence (DFKI), Germany
Email: †firstname.lastname@dfki.de
‡Department of Electrical and Computer Engineering, Kharazmi University, Iran
Email: ‡soheili@khu.ac.ir

***Abstract*—Digital images of documents contain a rich set of information. To automate their extraction, computers are programmed to analyse the content of document images. Document layout analysis is vital in that respect and can enhance the optical character recognition. The boundaries of different document regions, i.e. paragraphs, figures, or tables can be estimated using the convolutional neural networks. In this paper, we present a deep neural network that is inspired by natural scene object detectors. The network is trained and tested using the labeled samples from a large public dataset. Results demonstrate the potential of using object detectors for layout analysis. An implementation of the method will be available at: https://github.com/minoueikl/layout-detection.**

***Index Terms*—document layout analysis, object detection, optical character recognition, deep learning**

I. Introduction

Automatic recognition of texts is precious for retrieving the desired information from document images. Documents are comprised of many sections (segments), such as tables, figures, lists, paragraphs, etc. Appropriate optical character recognition (OCR) of a document image demands proper treatment for each of these regions. For example, we need to preserve the relation between entries of a table or the sequence of list items, instead of just recognizing their characters.

Therefore, it is necessary to detect the elements of a document in advance. This is the problem of document layout analysis (DLA) or page segmentation, in which different elements in a document must be recognized and localized in a digital image. Considering the variety of possible layouts for a sample, this task is very challenging even with modern techniques.

Early methods were challenged to distinguish text lines from figures [1]. Approaches such as run-length streaming algorithm [2], projection profile [3] and white space analysis [4], were taken by these early methods. Although traditional algorithms were not limited to text lines and figures [5], descriptions from convolutional neural networks (CNNs) have outperformed the traditional hand-crafted features for page segmentation [6].

The underlying idea for applying CNNs to page segmentation is the analogy between objects in a scene and elements in a document. The same architecture for object detection is employed in [7] for localizing text lines, equations, figures, and tables. Li et al. proposed a hybrid method for page object detection [8]. First, they extracted candidate regions based on connected components and projection profiles. These regions are then classified and clustered using a model based on conditional random fields. Finally, to further enhance the classification accuracy of the large regions, a CNN is employed to get the final labels.

Schreiber et al. employed a network inspired by the R-CNN object detector [9] to localize tables [10]. The same architecture is trained for detecting tables, equations, and figures [11]. In [12], authors considered the neighbor regions when classifying document elements.

Many of the mentioned algorithms are experimented by limited sized, or custom labeled data. Despite the datasets proposed by ICDAR competitions [13]–[16], the need for large amounts of training samples for deep CNNs was not satisfied until the release of PubLayNet [17]. In 2019, Zhong et al. proposed this dataset of 358 thousands labeled documents with five classes, namely, figure, text, title, table, and list. With this dataset, deep learning methods can have a common criterion for evaluation and comparison.

In [18], Li et al. trained a network on PubLayNet and argued that natural scenes are different with documents, where regions are guaranteed to have spatial relations. Graph neural networks are employed to model this feature of document images.

In this paper we develop an architecture for detecting various document entities. The suitable properties of object detectors are combined and the results on PubLayNet demonstrate the accuracy of the proposed layout analyser. In the next section our architecture is explained. Section III details the experiments for training and comparison the network's performance. The paper will be concluded in section IV.

II. The Proposed Method

The problem of identifying the layout in document images has a conceptual resemblance to natural object detection. Similar to a natural scene, in a typical document there are various entities that occupy certain regions of the image. Locating and classifying these regions can be done with the same conventions of a general object detector. Here, we will develop a detector for locating and classifying the elements of a document.

Fig. 1 shows an overview of our proposed architecture. The input image is given to a deep CNN that summarizes the image

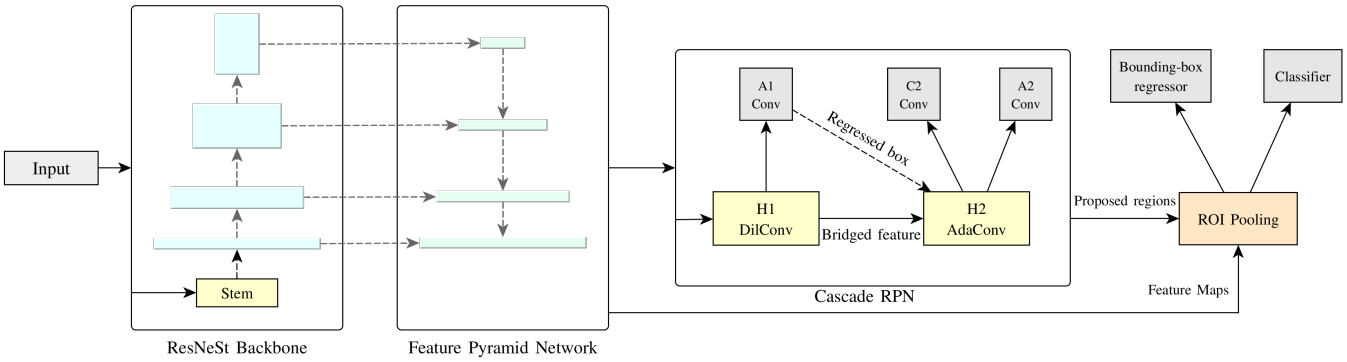


Fig. 1: A representation of the proposed method. First, the input image is fed into a CNN. Second, a feature pyramid network fuses the feature maps from the previous step. Third, a cascade RPN is used to find the potential object regions. At last, the ROI-Pooling layer is used to downsize the feature maps and feeds them to both the classifier and bounding box regressor. 'H', 'C', and 'A' designate the head, classifier, and anchor regressor of the cascade RPN.

into feature maps. The last four of these maps, are employed for further processes. The second box, fuses these maps and the regions are estimated based on these features. The rest of the network refines the proposed bounding boxes and performs the classification. In the following we will describe each part in detail.

Deep CNNs showed a great performance in extracting meaningful features from images; Therefore, they are widely used in image classification and object detection as the backbone feature extractor. Due to its superior performance in image classification challenges, we used ResNeSt [19] as the backbone of our architecture. It is a residual CNN based on ResNet [20] that has a multiple path design [21], employing a channel-attention mechanism [22] in each path. This design inherits all the benefits from its predecessors and resulted in more accurate predictions. Fig. 2 shows a ResNeSt block that have multiple paths with channel attention in each of them. The split attention mechanism learns to weight the channels based on their importance in determining the objects' category.

Since objects may appear in different sizes in images, it is straight-forward to perform image analysis in multiple scales. Although this is an effective attempt for increasing the accuracy [23], the computational demand leads to other heuristics for exploiting the information in multiple scales. With feature pyramid networks (FPN) [24], this is achieved by fusing multiple feature maps instead of multiple resolutions of the image. As shown Fig. 1, the feature maps from the backbone stages are fed into the network and then they get fused in a top-down pathway. Since high-resolution maps have more detailed features and low-resolution maps represent more semantic features, the fused features are more strong and scale invariant.

Locating objects in an image is challenging. Traditional methods used heuristics such as selective search [25] to generate a set of candidate regions; However, the region proposal network (RPN), which was introduced in Faster R-CNN [9], suggests a set of potential regions directly from the feature maps. In the RPN, authors introduced the concept of anchor boxes to support various scales and aspect ratios. For each pixel of a feature map, a set of anchor boxes with different scales and ratios are generated. Then these boxes are classified to foreground and background depending on their intersection of union (IOU) with ground truth. This process results in hundreds of proposals and the non-maximum suppression (NMS) is used to filter the duplicates.

Despite RPN's superiority to the previous methods, it has the shortcoming of being dependent on heuristically defined scale and aspect ratio values for the anchors. In contrast to anchor-based methods, in anchor-free methods only a single anchor is used that represents the center of an object [26]. This approach is faster yet can achieve inferior accuracy [27].

Cascade RPN [28] is a multi-staged region proposal network that exploits both anchor-based and anchor-free techniques to achieve higher performance. In its first stage, a set of anchors are uniformly initialized over the image using only a single anchor per location. In the second stage, the anchor-based metric is used to refine the outputs of the first stage. This is done using an adaptive convolution kernel that expands the sampling area from feature maps according to the proposed bounding box.

These characteristics of Cascade RPN would greatly benefit the document layout detection because documents mostly contain non-overlapping rectangular regions and a single anchor with the help of adaptive convolution is enough to detect page elements.

After the RPN proposed a set of bounding boxes, these boxes need to be classified to output categories. Fully-connected layers of perceptron generate the labels for the extracted regions. Since regions may have different sizes, a region of interest (ROI) pooling stage is performed to unify the size of feature vectors for the subsequent fully connected layers.

Our network generates the coordinates and dimensions of the detected bounding boxes along with a class probability

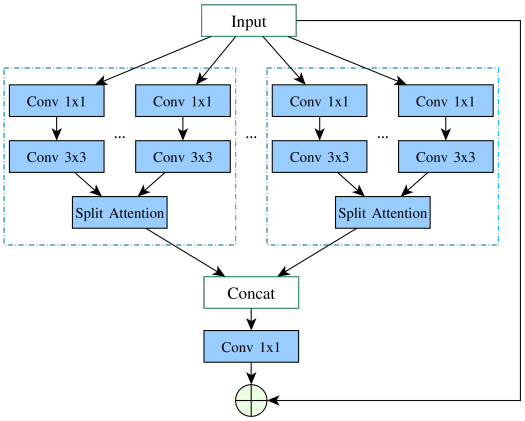


Fig. 2: A representation of a ResNeSt block. The input is fed into multiple paths of CNNs. In each path, channel attention mechanism is used. The final output is achieved by concatenating the path's outputs and the input.

for each region. It is end-to-end trainable and optimized for document images. In the next section the method is trained, tested, and compared using the PubLayNet dataset.

III. Experiments and Results

There are common conventions for evaluating the performance of object detection algorithms. In this section we will introduce our experimental methodology and analyse the predictions of our network.

A. Dataset and Evaluation Protocol

There are 358353 document images in PubLayNet dataset. The contents cover a wide range of scientific papers with both single-column and two-columns format. As mentioned before, the document elements (analogous to objects) in the pages are: figure, table, list, title, and text. Images are divided into three sets for train, development and test with 335703, 11245, and 11405 samples, respectively. The ground-truth for the first two sets is publicly available.

The ground-truth for each sample is the correct bounding boxes along with correct class probabilities. For assessing the output of a network, the intersection of union (IOU) between the network's bounding boxes and the ground-truth will be computed. The mean average precision (MAP) is computed for IOUs $\in [0.5, 0.95]$. This is the convention of COCO challenge [29] which is also adopted by the methods in [17]. Hence, the performance of the algorithm will be compared with these methods.

B. Implementation Details

The method is implemented using the MMDetection codebase [30]. The input to the network has 704 pixels in its largest dimension. The ResNeSt-50 that is used as the backbone, is not pre-trained and the entire network is trained on PubLayNet with 8 GPUs (4 images per GPU) and a mini batch size of 32. Synchronized batch normalization (SyncBN) [31] is incorporated with stochastic gradient descend (SGD) for 12 epochs in total. The learning rate decreases at the 6th and 9th epochs from 0.02 with a factor of 0.1. The values of weight decay and momentum are set to 0.0001 and 0.9, respectively[1].

C. Results

Table I compares our method's performance with compatible results of prior works. In [17] authors reported the performance of two base line methods: Faster R-CNN and Mask R-CNN [32].

Our proposed method achieved a higher accuracy in four of the categories. From the MAP indexes, it can be seen that titles are more challenging. This is due to visual similarities and class imbalance in the dataset. By its nature, a large area of a document is occupied by text, and since titles and lists also contain texts, they are commonly miss classified (Fig. 3). Nevertheless, the predictions made by our network are more accurate by a significant margin.

IV. Conclusion

Document layout analysis is an important pre-processing step for OCR. With the prevalence of smartphones that are capable of scanning the documents and also performing computations, the need for intelligent document analysis is increasing. In this work we proposed an enhanced method for detecting various elements in document images. With this method, it would be possible to obtain a more accurate OCR of a document paper. We leveraged the expressive features of ResNeSt as the backbone and Cascade RPN. By optimizing an object detector for document images, we could achieve an acceptable accuracy for the task of DLA. Performance of our proposed method is evaluated on the PubLayNet dataset that shows a significant improvement compared to the baseline.

References

[1] K. Y. Wong, R. G. Casey, and F. M. Wahl, "Document analysis system," *IBM journal of research and development*, vol. 26, no. 6, pp. 647–656, 1982.

[2] F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Computer graphics and image processing*, vol. 20, no. 4, pp. 375–390, 1982.

[3] G. Nagy and S. Seth, "HIERARCHICAL REPRESENTATION OF OPTICALLY SCANNED DOCUMENTS," *CSE Conference and Workshop Papers*, jan 1984. [Online]. Available: https://digitalcommons.unl.edu/cseconfwork/292

[4] K. Kise, O. Yanagida, and S. Takamatsu, "Page segmentation based on thinning of background," in *Proceedings of 13th International Conference on Pattern Recognition*, vol. 3. IEEE, 1996, pp. 788–792.

[5] O. Okun, D. Dœrmann, and M. Pietikainen, "Page segmentation and zone classification: the state of the art," OULU UNIV (FINLAND) DEPT OF ELECTRICAL ENGINEERING, Tech. Rep., 1999.

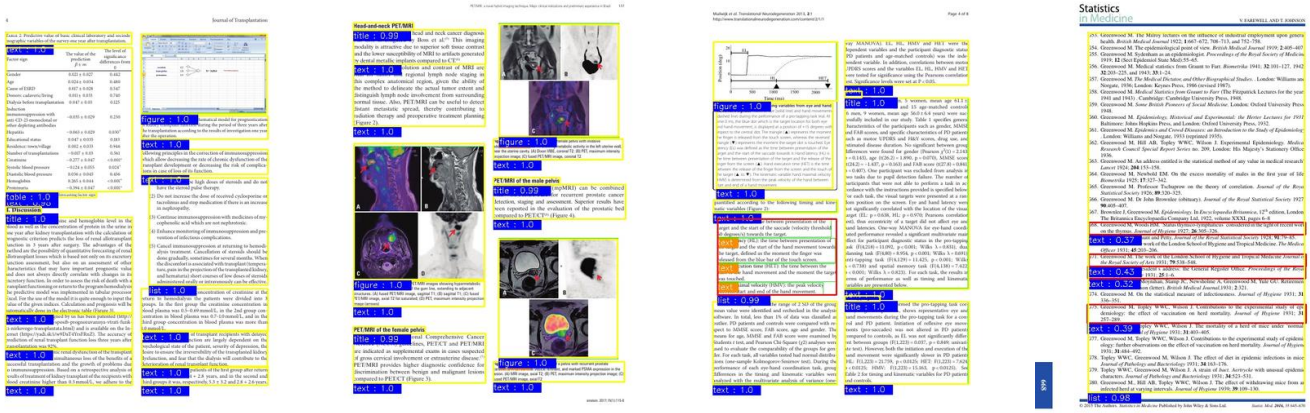
[6] G. M. Binmakhashen and S. A. Mahmoud, "Document layout analysis: A comprehensive survey," *ACM Comput. Surv.*, vol. 52, no. 6, Oct. 2019. [Online]. Available: https://doi.org/10.1145/3355610

[7] X. Yi, L. Gao, Y. Liao, X. Zhang, R. Liu, and Z. Jiang, "Cnn based page object detection in document images," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 01, 2017, pp. 230–235.

[1] An implementation of the method can be found at: https://github.com/minouei-kl/layout-detection

TABLE I: results on the development set of PubLayNet

Method	AP					Macro average
	Text	Title	List	Table	Figure	
F-RCNN [17]	0.91	0.826	0.883	0.954	0.937	0.902
M-RCNN [17]	0.916	0.84	0.886	0.96	0.949	0.91
CBM [18]	0.886	0.527	0.8683	0.9761	0.8376	0.819
MBC [18]	0.888	0.5279	0.8811	**0.9766**	0.8398	0.8227
Ours	**0.944**	**0.908**	**0.94**	0.974	**0.966**	**0.946**



(a) (b) (c) (d)

Fig. 3: Detection results with corresponding labels. (a) and (b) are true positive while (c) and (d) have false negatives and false positives

[8] X.-H. Li, F. Yin, and C.-L. Liu, "Page object detection from pdf document images by deep structured prediction and supervised clustering," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 3627–3632.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," 2015.

[10] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed, "Deepdesrt: Deep learning for detection and structure recognition of tables in document images," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 01, 2017, pp. 1162–1167.

[11] R. Saha, A. Mondal, and C. Jawahar, "Graphical object detection in document images," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2019, pp. 51–58.

[12] A. Goswami, J. McGrath, S. Peters, and T. Rekatsinas, "Fine-grained object detection over scientific document images with region embeddings," *arXiv preprint arXiv:1910.12462*, 2019.

[13] A. Antonacopoulos, B. Gatos, and D. Karatzas, "Icdar2003 page segmentation competition," 2003.

[14] A. Antonacopoulos, S. Pletschacher, D. Bridson, and C. Papadopoulos, "Icdar 2009 page segmentation competition," in *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009, pp. 1370–1374.

[15] A. Antonacopoulos, C. Clausner, C. Papadopoulos, and S. Pletschacher, "Icdar2015 competition on recognition of documents with complex layouts-rdcl2015," in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2015, pp. 1151–1155.

[16] L. Gao, X. Yi, Z. Jiang, L. Hao, and Z. Tang, "Icdar2017 competition on page object detection," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 01, 2017, pp. 1417–1422.

[17] X. Zhong, J. Tang, and A. J. Yepes, "Publaynet: largest dataset ever for document layout analysis," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2019, pp. 1015–1022.

[18] X.-H. Li, F. Yin, and C.-L. Liu, "Page segmentation using convolutional neural network and graphical model," in *International Workshop on Document Analysis Systems*. Springer, 2020, pp. 231–245.

[19] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, Z. Zhang, H. Lin, Y. Sun, T. He, J. Mueller, R. Manmatha *et al.*, "Resnest: Split-attention networks," *arXiv preprint arXiv:2004.08955*, 2020.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[22] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[25] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[27] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128 837–128 868, 2019.

[28] T. Vu, H. Jang, T. X. Pham, and C. D. Yoo, "Cascade rpn: Delving into high-quality region proposal network with adaptive convolution," 2019.

[29] "Coco - common objects in context." [Online]. Available: https://cocodataset.org

[30] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[31] C. Peng, T. Xiao, Z. Li, Y. Jiang, X. Zhang, K. Jia, G. Yu, and J. Sun, "Megdet: A large mini-batch object detector," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6181–6189.

[32] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.