FFD: Figure and Formula Detection from Document Images

Junaid Younas*†, Syed Tahseen Raza Rizvi*†, Muhammad Imran Malik‡, Faisal Shafait‡,
Paul Lukowicz*†, Sheraz Ahmed†
*Kaiserslautern University of Technology, Germany
†German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
Email: firstname.lastname@dfki.de
‡National Centre of Artificial Intelligence, NCAI-NUST, Islamabad H-12, Pakistan
Email: firstname.lastname@seecs.edu.pk

***Abstract*—In this work, we present a novel and generic approach, Figure and Formula Detector (FFD) to detect the formulas and figures from document images. Our proposed method employs traditional computer vision approaches in addition to deep models. We transform input images by applying connected component analysis (CC), distance transform, and colour transform, which are stacked together to generate an input image for the network. The best results produced by FFD for figure and formula detection are with F1-score of 0.906 and 0.905, respectively. We also propose a new dataset for figures and formulas detection to aid future research in this direction. The obtained results advocate that enhancing the input representation can simplify the subsequent optimization problem resulting in significant gains over their conventional counterparts.**

***Keywords*—** deep-learning, computer vision, transfer learning, mask RCNN, figure detection, formula detection,.

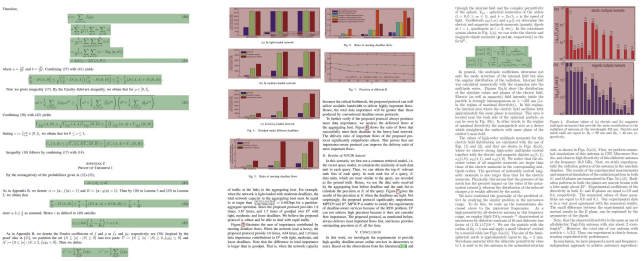


Fig. 1: Examples of annotated document images from the FFD dataset; green colour annotates formulas, brown colour represents figures

I. Introduction

Figures and formulas are an integral part of scientific documents. Figures are the simplest and one of the most effective ways to communicate complex ideas in a concise form. Similarly, mathematical formulas are also equally important to describe the relations between concepts and objects concretely and effectively. As a diverse set of tools are used to create documents, information presented in them may appear in varying forms. Information present in documents may include the document title, author information, text, figures, formulas, and tables along with several other entities. This diverse set of document layouts makes figure and formula detection a very demanding task.

Automated and reliable information extraction in document images has been a core focus of the document analysis community for a long time [1, 2, 3, 4]. Automated document processing has applications in several key areas which includes litigation, intelligence analysis, and knowledge management both for commercial and non-commercial entities. Figure and formula detection is an essential ingredient in automated document processing, which can not only help in the digitization of records but can also enable easy and remote access to data on demand. Digital document images owe several advantages over their traditional counterparts, which include easy retrieval, search, copies and transmission.

Figures and formula detection from document images is not only a challenging task but also a foundation for systems used to generate transcription of a given document image. Figure detection enables the system to signify textual and non-textual regions in document images. Figures include natural scene images, graphs, charts, layout designs, block diagrams or maps. We don't consider decorative graphics i.e., long lines and rules as figures in current work. Mathematical formula detection is equally important and significant in page segmentation and page object detection. Formulas may visually appear similar to text but they are different in structure, as they are represented in a 2-D arrangement. Since the conventional text processing pipeline fails in the analysis of formulas, it is essential to detect and ignore them during Optical Character Recognition (OCR).

Figures and formulas detection along with tables is also an important and critical step in automatic document generation e.g., presentation slides, posters, and/or technical reports, etc. [5, 6]. Figures and formulas may appear on varying locations in document images depending on document format, layout, orientation, aspect ratio, and other factors. Few examples of annotated figures and formulas in document images are shown in Figure 1. Therefore, it is not easy to directly identify figures and formulas from document images. This might be the reason for why existing commercial or open-source systems lack support for this functionality.

Page object segmentation and detection from document images remains an important subject of research in document-analysis community. Initial work in figure detection was focused on low-level analysis of geometric features, analysis of connected components and mathematical morphology [1, 2, 7]. As these methods rely heavily on hand-crafted heuristics and thresholds, they might perform well in a specific scenario but fail to generalize in other cases. With the introduction of deep-learning methods, major developments have been made for page objects segmentation and detection [4, 8, 9, 10]. All these methods involve either pre-processing, post-processing or both based on heuristics before generating the final results. Gilani et al. [11] used distance transforms to process input image and processed

them with Faster-RCNN. However, a combination of connected components, distance transform along with the raw input to be fed to the deep-learning model hasn't be explored in the past to the best of the authors' knowledge.

In this paper, we present a novel and generic end-to-end method to detect figures and formulas in document images. Our approach leverages the potential of traditional computer vision techniques to further strengthen the performance of deep learning models. We employed faster-RCNN [12] and mask RCNN [13] as deep models in our approach to detect figures and formulas occurring at different locations, scales, orientation and aspect ratios. We also leverage transfer learning in order to circumvent the need for having a large amount of labeled data. We tested our method on the publicly available ICDAR-2017 POD [14] along with a newly proposed *FFD* dataset[1]. Major contributions of presented work are:

- Merging of conventional computer vision techniques with deep neural networks to aid detection of page objects, figures and formulas in document images.
- Adaptation of deep object detection models for heterogeneous objects detection from document images.
- Curation of a new dataset for figures and formulas detection to benchmark the proposed approach.

The rest of the paper is organized as follows. Section-II covers the related work done in the field of figure and formula detection along with recent developments and existing state-of-the-art systems. Section-III provides a detailed overview of the FFD approach along with its components and methodology. Section-IV covers the details about the proposed dataset, evaluation protocol and network parameters for the training and testing phase. Section-V presents the results and their analysis; it also covers the major strengths and weaknesses of the proposed approach. Lastly, section-VI concludes the paper with possible future extensions.

II. Related Work

Page object detection got the attention of the document-analysis community in the recent past. Significant efforts have been invested in the domain of table detection [11] and table structure detection [8]. Figure and formula detection somehow remain unexplored to an extent. Existing work in this domain is based on heuristics, which includes colour-based features, shape-based features, and/or geometric features, etc. [1, 2, 15]. Deep learning approaches include conventional neural networks, region proposal networks and/or deformable neural networks [9, 16, 17]. Object detection can also be achieved by applying statistical methods i.e., conditional random fields (CRFs) or graph trees in combination with convolutional neural networks [10, 18].

Shih et al. [1] presented a graphics primitives algorithm from images of paper-based line drawings. They applied the MSM algorithm followed by extensive post-processing to generate a list of graphics primitives and their attributes. Ha et al. [2] presented a top-down page segmentation technique based on the recursive X-Y algorithm. They use colour transform on input image followed by connected component analysis to generate the bounding boxes. Projection profiles were generated from bounding boxes recursively to generate the segmentation results.

Cronje et al. [19] presented a solution to extract figures, captions and part labels from patents. They applied conventional computer vision techniques i.e., colour transform, connected component analysis and character recognition along with predefined heuristics. As a first step, they segmented the textual and non-textual regions from a patent image. Non-textual image parts were further post-processed based on predefined heuristics to generate bounding boxes around the figures in the input image.

[1] *FFD* dataset will be made publicly available

Iwatsuki et al. [18] presented preliminary results to detect mathematical expressions in scientific documents. They used PDF documents to construct manually annotated corpus, which were then processed by CRFs for the identification of math zones. Math zones were identified by using both layout features and linguistic features.

Kamola et al. [20] presented a solution to recognize the structure of textual and graphical regions in digital document images. Their approach is two-fold i.e. segmentation followed by recognition. During the segmentation process, masks for graphical regions in digital document images are generated using traditional computer vision methods. Textual regions are then extracted using connected component analysis followed by character recognition methods. A major limitation of their proposed approach is a set of requirements for input images to be sufficed i.e., image source, quality, background, etc.

Yi et al. [9] presented a method to detect text-lines, formulas, figures, and tables from document images. The proposed three step classification method starts with a component-based region proposal method to generate the region of interests from an input image which were classified by CNN. At the last stage, a dynamic post-processing algorithm is used to generate a final classification result. The authors also claimed to introduce an open-access dataset of $12,000$ document images, but no information or link to access the dataset has been provided.

Li et al. [10] presented a method to detect page objects i.e., figures, tables, and formulas from PDF document images. They used a hybrid model, which combined deep structured convolutional neural network (CNN) predictions with supervised clustering. Their presented method is a combination of conventional computer vision techniques, deep neural networks, and statistical models. The input image is segmented into row and column regions by applying traditional computer vision techniques. These row regions are classified by CNN into objects, which were clustered by conditional random fields (CRFs) followed by post-processing to assign a final label. They establish the utility of their method on the widely used ICDAR-2017 POD dataset.

Vo et al. [21] presented one of the recent works for page object detection (POD) which includes figures, formulas and tables. Their approach is based on deep neural networks which is an ensemble of fast-RCNN, and faster-RCNN. They combine the region proposals from Fast-RCNN and Faster-RCNN before applying bounding box regression to boost performance. They benchmark their approach on the ICDAR2107-POD dataset.

One thing which is common in all these methods is the use of pre-processing and/or post-processing. Pre-processing and post-processing are mainly carried out based on hand-defined heuristics, which puts a question mark on the generality of existing systems. In contrast, our proposed method doesn't involve any heuristic-based pre and/or post-processing, which is a major advantage over existing methods.

III. FFD: The proposed approach

The proposed approach takes the benefit of traditional computer vision techniques in combination with deep learning models. Deep learning models are known for their state-of-the-art performance for object classification, detection, and segmentation in natural scene images. Document images are sparse signals where most of the information is blank. Therefore, a memory module is usually required to capture these dependencies. This is partly the reason why LSTMs have been particularly famous in the document analysis community. This sparse nature is in contrast to natural images which are dense in terms of content that the CNN can effectively exploit. In order to enable the system to capture long-term dependencies without the requirement of having an explicit memory module, we augment the input signal itself with contextual information.

Connected component analysis is one of the most common techniques used to conserve the relationship of details present in

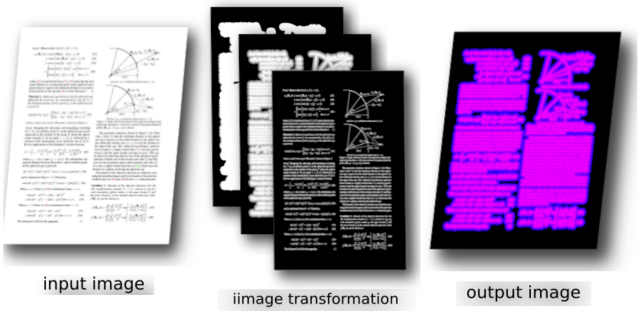


Fig. 2: Image transformation process with intermediate steps

document images [2, 7, 19, 22]. Another technique used to transform input image is the distance transform [11] to preserve the precise distance between page entities. So, we take advantage of both connected components and inverse distance transform to process input image, but also preserve original image information as a grey-scale image. We stack these transformed images together, each in single-channel before feeding it to the network. The resulting image contains contextual information, precise distance information and most importantly, the original information of input document image. The process of image transformation from input image to output with intermediate steps is shown in Figure 2.

We adhere to faster-RCNN [12] and mask-RCNN [13] in our approach built upon the pre-trained ResNet-50 [23] on the ImageNet[2] dataset. Using a pre-trained network enables our approach of domain adaptation from natural scene images to document images by taking advantage of transfer learning. Deep neural networks are data-driven and require extensive resources in the training phase; to which transfer learning is a remedy. Transfer learning is an important aspect of deep neural networks, as it avoids over-fitting along with better resource utilization because of a useful initialization point.

Faster-RCNN has been successfully used for page object detection from document images in the recent past [8, 11, 21], but mask-RCNN is used for the very first time to detect objects from document images to the best of the authors' knowledge. Faster-RCNN is a combination of three networks: a feature extraction backbone, a region proposal network (RPN) to generate bounding boxes for potential candidates present in an input image and a classification network with bounding box regression to classify the region of interest. Mask RCNN adopts an additional Fully Convolutional Network (FCN) [24] along with region proposal detection and object classification. With the help of the FCN, mask RCNN additionally segments the detected object by generating binary masks in parallel to bounding boxes and classification scores. FFD along with its complete pipeline is shown in Figure 3.

In FFD, the transformed input image is fed to the feature extraction backbone, which not only generates the feature map but also preserves the shape and structure of the original image. Using pre-trained weights from a state-of-the-art image classification network with final layers sheared off is a common practice to overcome the large dataset requirements as training on these large-scale datasets transforms the inital layers of the network into a generic feature extractor. Pre-trained ResNet-50 [25] up to final convolutional layer of 4^{th}-stage is used as the feature extractor in FFD.

Region proposal network (RPN) predicts bounding boxes of all possible candidate regions commonly termed as anchors and their possibility of being foreground or background based on overlap. It also refines the anchors. Input to RPN is a feature map, the output of the feature extraction backbone. RPN is a small convolutional network, which transforms $x \times x$ spatial input into a lower-dimensional feature. This feature is used for bounding box regression and classification. In FFD, we used four different anchor scales along with three aspect ratios, resulting in a total of 12 anchors. Multiple anchors help the network in overcoming variability in terms of size present in real-world objects.

Region proposal networks are followed by a detection or classification network, usually known as RCNN. RCNN takes the input from both the feature network and RPN to generate the final class label and bounding box offsets for every input region. By doing so, detection network crops the features from feature network using bounding boxes fed from RPN to classify the object present inside the bounding box. Both faster-RCNN and mask RCNN share the same pipeline to this step.

Mask RCCN implements an additional module by using the fully convolutional network to generate pixel-level binary masks for every region of interest (RoI). Input objects are encoded in spatial-layout by mask representation. It also implements RoI alignment to preserve explicit per-pixel spatial correspondence of input RoI features. We refer readers to [12, 13] for comprehensive details of faster-RCNN and mask RCNN.

[2]http://www.image-net.org/

Split	document images	Figures	formulas
Train	480	681	1,212
Test	200	308	708
Total	680	989	1,929

TABLE I: Dataset content details including numbers of objects present in training and test set

IV. Dataset and Evaluation Protocol

A. Dataset

ICDAR-2017 POD competition dataset [26] is the largest publicly available dataset for page object detection to the best of the authors' knowledge. There is a real need for a publicly available dataset for cross-evaluation and to achieve generalization. Therefore, we collected and manually annotated a dataset named *FFD* particularly targeted towards formulas and figure detection. The dataset consist of 680 document images from 100 scientific papers in English language available at *arXiv*[3]. The collected document images are from different disciplines and cover a variety of page formats, layouts, and styles. Page objects present in every document image also show diversity and variability.

We manually annotated only two classes i.e., figures and formulas. Example document images from FFD dataset are visualized in figure 1. There are a total number of $1,929$ formulas and 993 figures present in *FFD* dataset. Every document image carries a corresponding *XML* with annotated ground-truth information in *PASCAL-VOC* format. Out of 680 document images, 70% are used in the training set and the remaining 30% are placed in the test set. FFD dataset will be made publicly available for the research community to aid research in this direction[4].

B. Network Parameters

We train and test faster-RCNN and mask RCNN on the presented dataset. Input images are rescaled to the size of $1,000 \times 1,200$ before feeding them to the network. A single image per batch is used. We used the Detectron implementation [25] of both faster-RCNN and mask RCNN including pre-trained weights of ResNet-50. Extracted features till the final 4^{th}-stage convolutional layer of pre-trained ResNet-50 are used as backbone in both models. 4 different anchor scales of $[32 \times 32, 64 \times 64, 128 \times 128, 256 \times 256]$ with 3 aspect ratios of [1:2, 1:1, 2:1] are used in this implementation. All

[3]https://arxiv.org/

[4]http://bit.ly/2lNvWfL

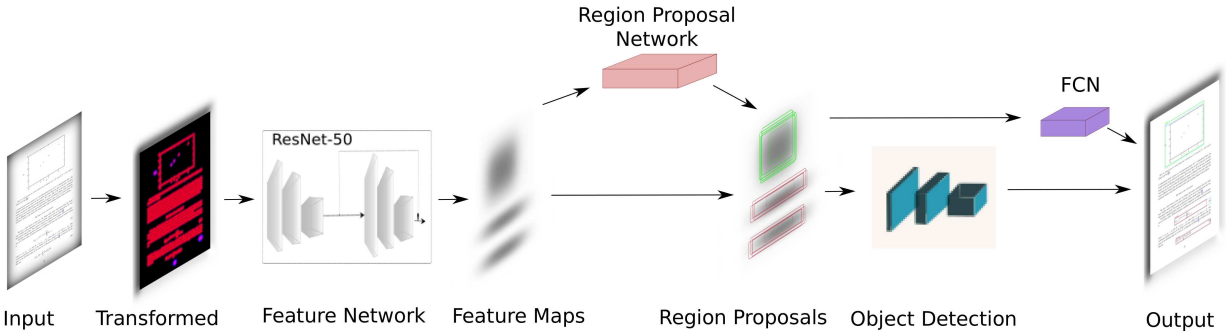


Fig. 3: FFD pipeline with all its components.

models are trained for 100 epochs with a learning rate of 0.001 with the learning rate scheduling. A non-maximum suppression (NMS) threshold of 0.3 in combination with class score is used on region proposals for bounding box regression. The confidence threshold to retain the prediction is set to 0.6. All models were trained on a single 1080Ti GPU.

C. Evaluation Protocol

We follow the standard evaluation protocol defined for page object detection in the ICDAR-2017 POD competition. For object detection results are computed by area under Intersection-over-Union (IoU) metric. So, results presented in this work are computed on IoU thresholds of 0.6, & 0.8. First, we compute true positives (TPs), false positives (FPs) and false negatives (FNs), which were then used to compute metrics of precision, recall, f1-score, average precision and mean average precision (mAP).

V. Results and Discussion

We evaluate the presented approach using faster-RCNN and mask RCNN on *FFD* (our collected dataset) and the publicly available ICDAR-2017 POD dataset for page object detection. We report results on the standard IoU threshold of 0.6 and 0.8 defined for the ICDAR-2017 POD competition. Best results are achieved by mask RCNN for figure detection on the IoU threshold of 0.6, F1-score of 0.906 with a precision of 0.908 and recall of 0.905.

Faster-RCNN detected figures on the IoU threshold of 0.6 with the precision and recall of 0.89 and 0.899, which translated into F1-score of 0.894, while evaluating on the *FFD* dataset. On the IoU threshold of 0.6 formulas are detected with the precision of 0.916, recall of 0.89 and f1-score of 0.903. When the IoU threshold is set to 0.8, numbers for faster-RCNN on formulas detection dropped down to the F1-score of 0.591 with a precision and recall of 0.596 and 0.577 respectively. Similarly, results for figure detection also drop to 0.77, 0.781, and 0.776 in terms of precision, recall, and F1-score, respectively. Average precision for formula detection is 0.875 and 0.851 for figure detection.

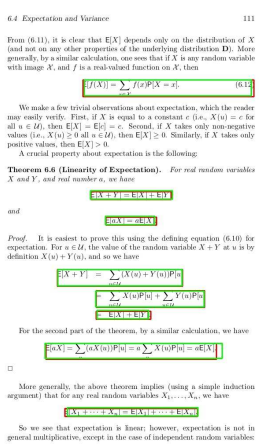
Mask RCNN produced better results in comparison to faster-RCNN for both figure and formula detection, as shown in Table II. Figures are detected with a precision of 0.908, recall is translated into numbers as 0.905 and F1-score measures to 0.906 on the IoU threshold of 0.6. Formulas are detected with a precision of 0.898, recall and F1-score are 0.913 and 0.905, respectively. Precision for figure detection on IoU 0.8 is calculated as 0.809 with a recall of 0.814 and F1-score of 0.811. Numbers for formula detection realized to 0.711, 0.723, and 0.717 as precision, recall, and F1-score, respectively. Average precision for figure detection comes as 0.894 and that for formulas is 0.892.

On the ICDAR-2017 POD dataset, FFD performed equally well as the results show in Table II. On the IoU threshold of 0.6, both faster-RCNN and mask RCNN were competitive in terms of performance. When IoU threshold is increased to 0.8, a significant drop in numbers for formulas detection using faster-RCNN is observed in comparison to mask RCCN. On the IoU threshold of 0.6, mask RCNN recognized figures with a precision of 0.894 against the recall of 0.918 and F1-score is computed as 0.905. Formulas are detected with a precision, recall, and F1-score of 0.894, 0.921, and 0.907, respectively. On the IoU threshold of 0.8 figures and formulas are detected with the f1-score of 0.816 and 0.811, respectively.

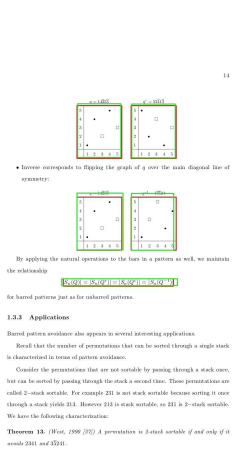
Results produced by faster-RCNN and mask RCNN establish the connotation of object detection deep models for document images. Results also institute the potential of the presented method for figure and formula detection in document images, as shown in Table II. FFD performs equally well on both the ICDAR-2017 POD dataset along with the *FFD* dataset. As results depict, our method also demonstrates its convergence strength, as it achieves competitive results on the *FFD* dataset in comparison to the ICDAR-2017 POD (about three times larger) dataset, keeping in mind deep neural networks are data-driven methods.

Results produced by FFD using both faster-RCNN and mask RCNN are very encouraging because of their ability to characterize formulas and figures from other page objects, as shown in Figure 5 and 6. Most of the time, figures were correctly identified, as shown in Figure 5b and 6b. On a few occasions, figures were confused with tables and algorithms (code snippets), as visualized in Figure 6c. A few times, figure detection also faces the problem of under-segmentation (partial detection) or over-segmentation, as shown in Figure 5c. For formula detection, numbers might not translate to the actual potential of FFD. Visual results shown in Figure 5 and 6, show formula regions are detected correctly most of the time, but due to over-segmentation or under-segmentation and higher IoU threshold, they don't decipher to correct prediction, as shown in Figure 5d & 6d. For simplicity's sake, we avoided any post-processing on the results. However, we do expect that simple post-processing on detected regions will result in a significant boost in performance.

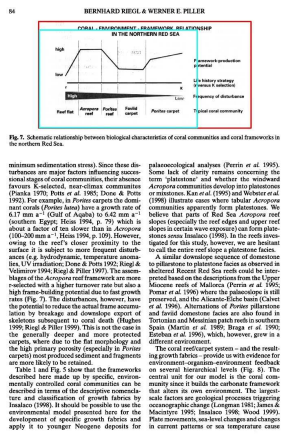
Existing state-of-the-art systems based on either conventional computer vision techniques or machine learning/deep learning methods are implemented based on pre-processing and/or post-processing modules. Performance of these pre-processing and/or post-processing modules completely depends on cherry-picked heuristics, making their systems tailor-made for only on a given dataset. The use of heavy heuristics limits the generality and scalability of existing state-of-the-art systems waiving out their applications for real-world use. On the Contrary, FFD has no bells and whistles associated with it and results shown in Table II establish its performance across different datasets. The main advantage FFD owes to existing state-of-the-art systems is its ability to be an end-to-end system, which makes it generic and scalable. However, results displayed by FFD in this work are marginally lower than the existing state-of-the-art method, which can be compensated against much lower computational costs required by FFD being an end-to-end system. Moreover, FFD can be adopted



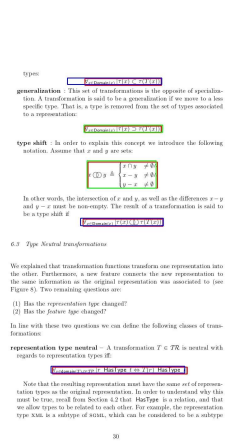
(a) True positives of formulas



(b) True Positives of figures



(c) FPs & FNs for figures



(d) FPs & FNs for formulas

Fig. 4: Analysis of results generated by FFD detector using mask RCNN on ICDAR-2017 POD dataset at $IoU = 0.8$, red colour annotates ground truth and false negatives, green colour highlights TPs, cyan annotates FPs for figures and blue colour represents FPs for formulas.

TABLE II: Comparison of FFD with existing state-of-the-art methods

Method	Class	$IoU = 0.6$				$IoU = 0.8$			
		Precison	Recall	F1-score	AP	Precision	Recall	F1-score	AP
NLPR-PAL [26]	Formula	0.901	0.929	0.915	0.839	0.888	0.916	0.902	0.816
ICDAR-2017 POD	Figure	0.920	0.933	0.927	0.849	0.892	0.904	0.898	0.805
Li et al. [10]	Formula	0.930	0.953	0.942	0.878	0.921	0.944	0.932	0.863
ICDAR-2017 POD	Figure	0.948	0.940	0.944	0.896	0.921	0.913	0.917	0.850
Faster-RCNN	Formula	0.894	0.889	0.897	0.873	0.760	0.570	0.650	0.671
ICDAR-2017 POD	Figure	0.894	0.900	0.897	0.862	0.811	0.801	0.806	0.787
Mask RCNN	Formula	0.894	0.921	0.907	0.897	0.788	0.835	0.811	0.776
ICDAR-2017 POD	Figure	0.894	0.918	0.905	0.886	0.805	0.828	0.816	0.794
Faster-RCNN	Formula	0.916	0.89	0.903	0.875	0.596	0.577	0.591	0.448
FFD dataset	Figure	0.890	0.899	0.894	0.851	0.770	0.781	0.776	0.750
Mask RCNN	Formula	0.898	0.913	0.905	0.892	0.711	0.723	0.717	0.621
FFD dataset	Figure	0.908	0.905	0.906	0.894	0.809	0.814	0.811	0.791

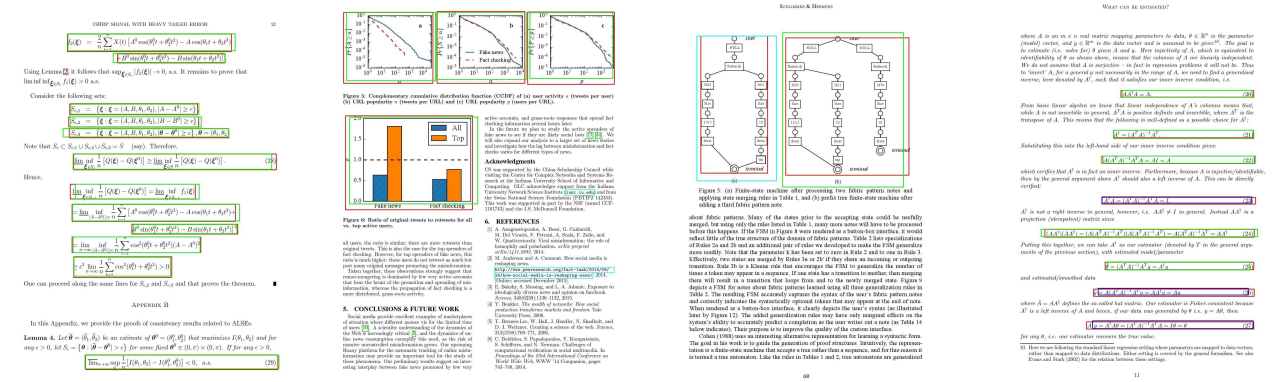
to any real-world scenario for figure and formulas detection with minimal effort.

We also present a new open-access dataset for figure and formulas detection in document images. Why a new dataset? Since the ICDAR-2107 POD dataset is the only publicly available dataset for page object detection to the best of author's knowledge. There are labeling problems in the said dataset, which include missing labels, wrong labels and most of all non-uniform labeling conventions. By non-uniform labeling, we refer to annotations where some annotations are inconsistent with the rest of the dataset, e.g., at one instance, a block of formulas was labeled as a single entity, while similar blocks on other pages were labeled correctly as distinct entities. The same is the case with figure annotation; at one instance, figures were annotated considering outer boundary, whereas on other pages outer boundary was totally neglected. These problems affect the performance of systems. Therefore, a new and clean dataset based on uniform labeling conventions is needed. Our proposed dataset addresses all problems found in the existing ICDAR-2017 POD dataset, since it has been manually annotated by a single human, which significantly enhances the uniformity in terms of labeling conventions. Hence the presentation of a new dataset will help in achieving generalized and scalable systems capable of performing equally well in different scenarios for the same problem. Moreover, we didn't annotate tables as existing open-source and commercially available document image processing tools i.e., Tesseract, Abbyy, etc., have already embedded table detection in document images.

VI. Conclusion and Future Work

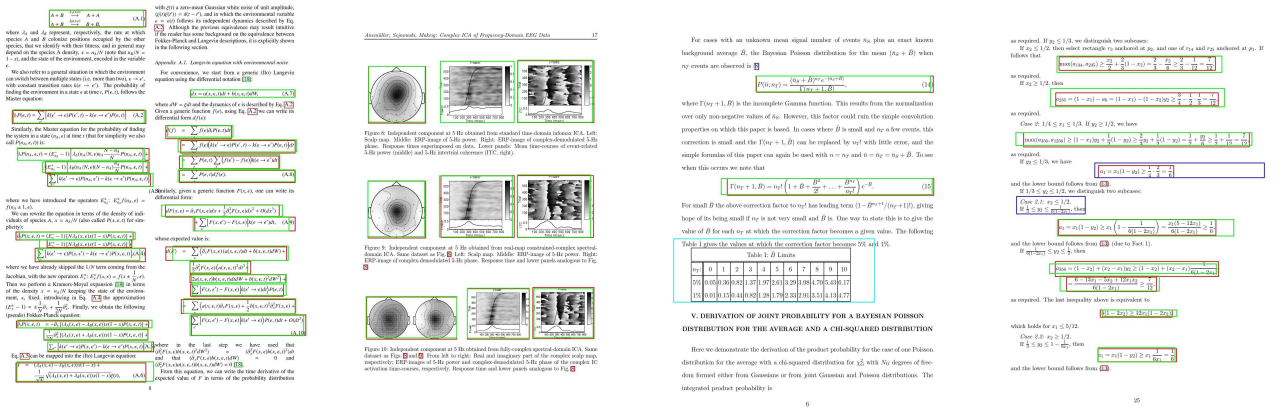
We propose FFD, an RCNN based approach for figure and formulas detection in document images. FFD uses a novel combination of existing computer vision techniques to aid deep learning classifiers. Our presented approach, FFD, is an end-to-end system without any bells and whistles associated with it, which existing state-of-the-art systems lack. Moreover, FFD is a generic method to detect figures and formulas from document images because of its equitable performance on the publicly available ICDAR-2017 POD and the newly proposed *FFD* dataset. We also propose a new publicly available dataset for figure and formulas detection to further push the boundaries of research in this direction.

One of the most trivial future directions is to try existing methods on the newly proposed dataset in order to establish baseline results. Another direction is to extend the dataset itself by adding annotations for other page objects e.g. tables, textual regions, section headings, etc. Moreover, the potential of deformable neural networks can also be explored for page object detection. We also plan to extend the



(a) True positives of formulas (b) True Positives of figures (c) False positive for figures (d) FPs & FNs for formulas

Fig. 5: Analysis of results generated by FFD detector using Faster-RCNN on FFD dataset, red colour annotates ground truth and false negatives, green colour highlights TPs, cyan annotates FPs for figures and blue colour represents FPs for formulas.



(a) True positives of formulas (b) True Positives of figures (c) False positive for figures (d) FPs & FNs for formulas

Fig. 6: Analysis of results generated by FFD detector using mask RCNN on FFD dataset, red colour annotates ground truth and false negatives, green colour highlights TPs, cyan annotates FPs for figures and blue colour represents FPs for formulas.

presented work to establish its utility in multiple scenarios for figure and formulas detection.

VII. ACKNOWLEDGMENT

Authors would like to thank Shoaib Ahmed Siddiqui for support. This work is partially funded by Higher Education Commission (HEC), Pakistan.

REFERENCES

[1] C. Shih and R. Kasturi. Extraction of graphic primitives from images of paper based line drawings. *Machine Vision and Applications*, 2(2):103–113, Mar 1989.

[2] R. M. Haralick and I. T. Phillips. Recursive x-y cut using bounding boxes of connected components. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 2, pages 952–955 vol.2, Aug 1995.

[3] T. Kieninger and A. Dengel. The T-Recs Table Recognition and Analysis System. In Seong-Whan Lee and Yasuaki Nakano, editors, *Document Analysis Systems*, Lecture Notes in Computer Science, pages 255–270. Springer, Berlin, 1999.

[4] J. Younas, M. Z. Afzal, M. I. Malik, F. Shafait, P. Lukowicz, and S. Ahmed. D-star: A generic method for stamp segmentation from document images. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 248–253, Nov 2017.

[5] Yuting Q., Yanwei F., Yanwen G., Zhi-Hua Z., and Leonid S. Learning to generate posters of scientific papers, 2016.

[6] R. Spicer, Y. Lin, A. Kelliher, and H. Sundaram. Nextslide-please: Authoring and delivering agile multimedia presentations. *TOMCCAP*, 8:53:1–53:20, 2012.

[7] A. K. Das, S. P. Chowdhury, S. Mandal, and B. Chanda. Automated segmentation of math-zones from document images. In *2013 12th International Conference on Document Analysis and Recognition*, volume 3, page 755, Los Alamitos, CA, USA, aug 2003. IEEE Computer Society.

[8] 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 *ICDAR*, pages 1162–1167. IEEE, 2017.

[9] 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), volume 01, pages 230–235, Nov 2017.

[10] X. Li, F. Yin, and C. Liu. Page object detection from pdf document images by deep structured prediction and supervised clustering. *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3627–3632, 2018.

[11] A. Gilani, S. Qasim, I. Malik, and F. Shafait. Table detection using deep learning. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pages 771–776, Los Alamitos, CA, USA, nov 2017. IEEE Computer Society.

[12] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015.

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, Oct 2017.

[14] L. Gao, X. Yi, Y. Liao, Z. Jiang, Z. Yan, and Z. Tang. A deep learning-based formula detection method for PDF documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 553–558, Nov 2017.

[15] R. Zanibbi and D. Blostein. Recognition and retrieval of mathematical expressions. *IJDAR*, 15(4):331–357, 2012.

[16] I. Kavasidis, S. Palazzo, C. Spampinato, C. Pino, D. Giordano, D. Giuffrida, and P. Messina. A saliency-based convolutional neural network for table and chart detection in digitized documents. 04 2018.

[17] S. Siddiqui, M. Malik, S. Agne, A. Dengel, and S. Ahmed. Decnt: Deep deformable CNN for table detection. *IEEE Access*, 6:74151–74161, 2018.

[18] K. Iwatsuki, T. Sagara, T. Hara, and A. Aizawa. Detecting in-line mathematical expressions in scientific documents. In *Proceedings of the 2017 ACM Symposium on Document Engineering*, DocEng '17, pages 141–144, New York, NY, USA, 2017. ACM.

[19] J. Cronje. Figure detection and part label extraction from patent drawing images. In *23rd Annual Symposium of the Pattern Recognition Association of South Africa (PRASA)*, 11 2012.

[20] G. Kamola, M. Spytkowski, M. Paradowski, and U. Markowska-Kaczmar. Image-based logical document structure recognition. *Pattern Analysis and Applications*, 18(3):651–665, Aug 2015.

[21] N. Vo, K. Nguyen, T. Nguyen, and K Nguyen. Ensemble of deep object detectors for page object detection. In *Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication*, IMCOM '18, pages 11:1–11:6, New York, NY, USA, 2018. ACM.

[22] S. Bukhari, M. Al Azawi, F. Shafait, and T. Breuel. Document image segmentation using discriminative learning over connected components. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, DAS '10, pages 183–190, New York, NY, USA, 2010. ACM.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014.

[25] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. https://github.com/facebookresearch/detectron, 2018.

[26] 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)*, volume 01, pages 1417–1422, Nov 2017.