IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 13, 2018, accepted October 31, 2018, date of publication November 20, 2018,
date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2880211

DeCNT: Deep Deformable CNN for Table Detection

SHOAIB AHMED SIDDIQUI[1,2], MUHAMMAD IMRAN MALIK[3], STEFAN AGNE[1], ANDREAS DENGEL[1,2], AND SHERAZ AHMED[1]

[1]German Research Center for Artificial Intelligence, 67663 Kaiserslautern, Germany
[2]Department of Computer Science, University of Kaiserslautern, 67663 Kaiserslautern, Germany
[3]School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Islamabad, Pakistan

Corresponding author: Shoaib Ahmed Siddiqui (shoaib_ahmed.siddiqui@dfki.de)

This work was supported in part by the BMBF Project DeFuseNN under Grant 01IW17002 and in part by the NVIDIA AI Lab Program.

ABSTRACT This paper presents a novel approach for the detection of tables present in documents, leveraging the potential of deep neural networks. Conventional approaches for table detection rely on heuristics that are error prone and specific to a dataset. In contrast, the presented approach harvests the potential of data to recognize tables of arbitrary layout. Most of the prior approaches for table detection are only applicable to PDFs, whereas, the presented approach directly works on images making it generally applicable to any format. The presented approach is based on a novel combination of deformable CNN with faster R-CNN/FPN. Conventional CNN has a fixed receptive field which is problematic for table detection since tables can be present at arbitrary scales along with arbitrary transformations (orientation). Deformable convolution conditions its receptive field on the input itself allowing it to mold its receptive field according to its input. This adaptation of the receptive field enables the network to cater for tables of arbitrary layout. We evaluated the proposed approach on two major publicly available table detection datasets: ICDAR-2013 and ICDAR-2017 POD. The presented approach was able to surpass the state-of-the-art performance on both ICDAR-2013 and ICDAR-2017 POD datasets with a F-measure of 0.994 and 0.968, respectively, indicating its effectiveness and superiority for the task of table detection.

INDEX TERMS Deep learning, representation learning, convolutional neural networks, object detection, deformable convolution, table detection, table spotting, faster R-CNN, FPN.

I. INTRODUCTION

Paper based documents are still the most prevalent form of documents. With the presence of these paper based documents, there is an increasing demand for accurate detection of tabular data embedded in those documents for automated extraction of relevant information. Table detection and recognition is of particular interest to the document analysis community [1] due to its importance in sectors like finance where large statement sheets are formulated containing hundreds of values whose precise extraction is of prime importance, or business documents where the administration has to deal with thousands of documents on daily basis, or even digitization of archives where large amount of data is embedded in tabular form.

Table detection is a step towards the solution to the complete table understanding and analysis problem which is of prime importance to the end-users. Despite of significant efforts, generic analysis of tables with arbitrary layouts is extremely difficult. The problem of table detection is challenging due to the high intra-class and low inter-class variance. High intra-class variance is a result of presence of tables with arbitrary layouts, where some of the tables contains the ruling lines, while others don't carry any such information. Low inter-class variance between tables, figures, graphics, code listings, structurally laid out text, or flow charts also poses a significant challenge for successful detection of tables and reduction of false positives [2]. These challenges makes it particularly difficult to devise heuristics for table detection which are reliable and robust to changes [1].

Different methods have been proposed in the past for detection and analysis of tables but most of these methods are only applicable to digital-born PDFs where they exploit additional meta-data which aids and simplifies the analysis [3], [4]. Furthermore, heuristics are devised on top of the extracted information whose applicability is limited to a particular dataset or relies on assumptions which are not generally true. There have been only a limited number of attempts for the development of methods leveraging raw images directly for

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

the problem of table detection and recognition which makes the problem significantly harder to tackle [5]–[7]. Despite of the recent advancements in the domain of computer vision and document analysis, and a range of different methods proposed for this purpose, perfect recognition of tables with arbitrary layouts is still not a reality.

This paper presents an end-to-end system for the detection of tables in document images. The proposed deep learning based method automatically discovers features which are useful for the detection of tables in a wide and diverse range of documents. This representation learning based approach eliminates the requirement of defining custom heuristics which are very limited in their applicability. These heuristics were a major drawback of prior methods proposed for this task. The presented system is applicable to all types of documents such as born-digital PDFs as well as scanned document images. We leverage deformable convolution instead of the conventional convolution operation due to its dynamic receptive field which enables the network to detect tables present at arbitrary scales along with arbitrary transformations. Due to the lack of availability of large amount of data to train a network from scratch, we utilized pretrained deep networks on ImageNet as the backbone leveraging the advantages of transfer learning.

In particular, following are the contributions of this publication:

- We introduced a novel table detection framework leveraging the potential of deformable convolutional neural networks. The proposed method was able to achieve state-of-the-art performance on two well-known publicly available table detection datasets.
- We propose a combination of different datasets for creating a significantly large table detection dataset.
- We demonstrated the effectiveness of this methodology by testing on ICDAR-13, UNLV and Mormont dataset without using even a single image from these datasets for training, highlighting the generalization capabilities of the system in the real-world.

This paper is structured as follows: Section II provides a brief history of the literature already present on the topic of table detection. Section III discusses the models and ideas used in our experiments. Section IV provides details regarding the publicly available datasets employed for the corresponding experimentation. Section V presents the achieved results along with a comprehensive analysis. Finally, concluding remarks along with the possible future work is presented in section VI.

II. RELATED WORK

There has been a wide range of literature available on the topic of table detection and understanding. *TINTIN* (Text INformation-based Text INquiry) [3] utilized custom heuristics to extract structural elements from text and filter out tablular regions. *T-Recs* [8] is also a rule based system which isolates cells within a tabular structure and clusters cells to form rows and columns. *PDF-TREX* [4] is an extension of the T-Recs system for detection and extraction of tables from PDF documents by leveraging heuristics for alignment and grouping of elements. This approach does not rely on visual or linguistic features. *TARTAR* (Transforming ARbitrary TAbles into fRames) [9] is able to transform arbitrary tables (HTML, PDF, EXCEL etc.) into logical structure. This method uses a hierarchy of token types to discover functional type of each cell, which are then used to determine the orientation of the table using similarity of cells and their geometric positions. These cells are then arranged into logical units forming different regions of the table. We refer readers to [1], [2], [10]–[12] for a more detailed summarization of these conventional approaches.

Cesarini *et al.* [13] (2002) made one of the first attempts to incorporate machine learning techniques for the table detection task. The proposed method, *Tabfinder*, first converts a document into a MXY tree representation and then searches for blocks surrounded by horizontal or vertical lines for hypothesizing the presence of a table. This hypothesis is verified afterwards by identification of perpendicular lines or white spaces within the hypothesized region. Since the method relies on presence of ruling lines, it is not applicable to tables without such information.

Silva *et al.* (2009) [14] also presented a data-driven approach based on Hidden Markov Models (HMMs) modeling the joint probability distribution over sequential observations of visual page elements and the hidden state of a line regarding whether it is associated with a table or not. Silva *et al.* [15] (2010) built on top of their earlier findings and emphasized the importance of probabilistic models and the combination of multiple approaches over brittle heuristics. Since this method also relies on ruling lines, it is also limited to cases where this information is present.

Kasar *et al.* [16] (2013) computed a set of hand-crafted features for training the SVM classifier. Despite of being based on generic features, the method's area of application is limited due to its reliance on visible ruling lines making it unadaptable for table layouts without any such information.

Fan and Kim (2015) [17] utilized an unsupervised learning of weak labels for each line in the document as well as the textual information extracted from a region for the detection of tables. They trained an ensemble of generative and discriminative models to detect the tables in documents. Since their method relies on hand-crafted features, this again limits its applicability.

Tran *et al.* [18] (2015) proposed a method based on regions of interest and the spatial arrangement of extracted text blocks. As opposed to most other conventional methods, their method worked directly on images. There is no information present regarding which parts of the ICDAR-2013 table dataset were utilized for the design of the algorithm, ruling out a direct comparison to our method. Follow up work from Tran et al. also suffers from the lack of information regarding the dataset split.

There has been some recent attempts for the application of deep learning techniques for the task of table detection.

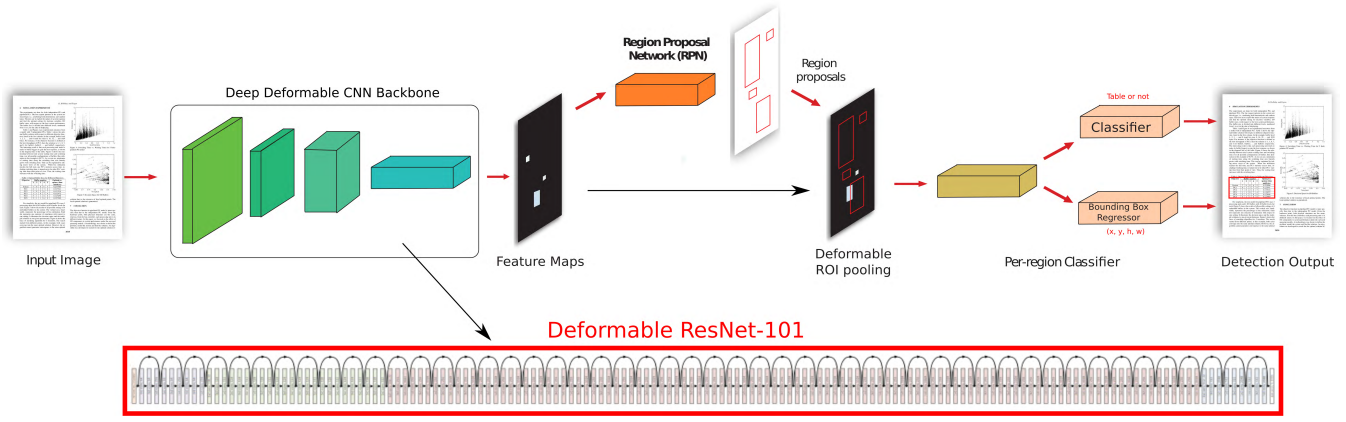


FIGURE 1. DeCNT - The proposed system pipeline.

The first technique was proposed by Hao *et al.* [19] (2016) but was limited to only PDF documents. The authors utilized self-defined heuristics, meta-information from PDF along with the features learned by the deep model to reach a final prediction.

An image based deep learning table detection method has been proposed by Schreiber *et al.* [5] (2017) where they utilized Faster R-CNN for detection of documents achieving state-of-the-art performance on ICDAR-2013. Their model was pretrained on the Pasval-VOC dataset. They utilized the pretrained ZFNet [20] and VGG-16 [21] while all our experiments are based on a much more powerful base model (ResNet-101). Their method also relies on raw images for the detection of tables.

Another deep learning method has been proposed by Gilani *et al.* [6] (2017) which is also based on Faster R-CNN for the detection of tables. Instead of feeding in the raw image pixel values, their system relies on distance transforms as tabular structures contains a well-defined spacing pattern. The authors claimed to utilize the whole ICDAR-13 dataset for evaluation but no information regarding the training dataset was present in the paper. This evaded the possibility for a direct comparison with their methodology. A very recent work powered by deep learning came to our attention towards the finalization of this project by Kavasidis *et al.* [7] (2018). The proposed system generates saliency maps using Fully-Convolutional Networks (FCN) [22] and uses Conditional-Random Fields on top of it to apply additional constraints towards the generated masks. C binary classifiers (where C was the number of classes) were trained to supplement the saliency network performance. They used dilated convolutions as proposed by Yu and Koltun [23] (2015) to increase the effective receptive field of the layers without any loss of resolution. Their method was trained on a large corpus of 50,442 randomly crawled images from the Internet, thus supporting our claim of increase in performance with the increase in the dataset size. The method was able to obtain state-of-the-art performance but required multiple training steps for the complete pipeline making the process overly complicated without any significant boost in the desired metrics.

III. DECNT: THE PROPOSED APPROACH

The proposed framework is comprised of a novel combination of deformable CNN with Faster R-CNN/FPN [24] as illustrated in Fig. 1. Convolutional neural networks are automatic feature extractors equipped with the ability to automatically discover features which are useful for the task at hand. This automatic extraction of features is based on a hierarchy of layers where initial layers extract primitive features like edges and gradients while layers on top of the hierarchy extracts very abstract features like complete objects or some prominent parts of it [25]. This traversal in hierarchy results in an increase in the effective receptive field of a particular neuron in the original input image. The conventional 2-D convolution operation can be represented mathematically as:

$$(F * I)(i,j) = \sum_{m=-K}^{K} \sum_{n=-K}^{K} F(m,n) \times I(i-m, j-n)$$
$$\forall i = 1, \ldots, H, \ \forall j = 1, \ldots, W \tag{1}$$

where $*$ denotes the convolution operation, F is the filter, I is the image, K is defined as $\lfloor FilterSize/2 \rfloor$, H is the image height, W is the image width, and i, j defines the location where the convolution operation is performed.

The effective receptive field of all neurons in a given convolutional layer is the same. This property is problematic for layers located on top of the hierarchy where different objects may appear at arbitrary scales along with arbitrary transformations. The presence of these transformations demands the ability to dynamically adapt the receptive field of a neuron based on its input. Therefore, we equip the Faster R-CNN/FPN model with a deformable CNN instead of the conventional CNN. Deformable convolutional layers were proposed by Dai *et al.* [26] (2017) where the neurons were not limited to a predefined receptive field. Each neuron can alter its receptive field according to its input via generating explicit

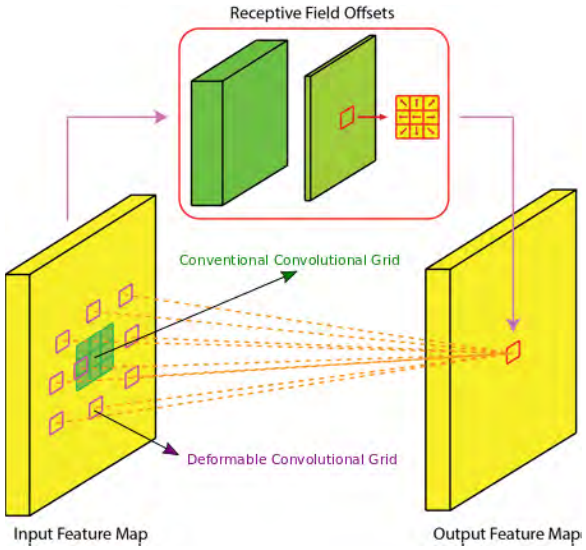


FIGURE 2. Deformable convolutional layer.

offsets which are themselves dependent on the preceding feature maps. This allows the convolutional layer filters to adapt to different scales and transformations by conditioning its receptive field on the input itself. This deformable convolutional layer is illustrated in Fig. 2 where a set of convolutional layers are added to generate filter offsets for each position in the image. As tables can be present at arbitrary scales along with arbitrary transformations (orientation etc.), deformable convolution operation is particularly useful for the task of table detection. The deformable 2-D convolution operation contains additional offsets which can be expressed mathematically as:

$$\begin{aligned}(F \circ I)(i,j) = &\sum_{m=-K}^{K}\sum_{n=-K}^{K} F(i,j) \\ &\times I(i-m+\delta^{vertical}_{i,j,m,n}, j-n+\delta^{horizontal}_{i,j,m,n}) \\ &\forall i = 1,\ldots,H,\ \forall j = 1,\ldots,W \end{aligned} \tag{2}$$

where we use $\circ$ to denote the deformable convolution operation such that all mutual parameters are same. $\delta^{vertical}_{i,j,m,n}$ defines the vertical offset while $\delta^{horizontal}_{i,j,m,n}$ defines the horizontal offset. These vertical and horizontal offsets are produced by another convolutional block consuming the feature maps produced by the preceding layer (Fig. 2). This conditioning of the offsets on the input features makes the offsets adaptable to local scale and transformations. The offsets can be fractional and are implemented via bilinear interpolation.

Furthermore, ROI-pooling is a core component for all region based detection methods [24], [27], [28]. The pooling layer converts feature maps of arbitrary size to a fixed volume to be fed to the final classification head which expects a fixed size input. However, we have a similar problem as the one in a conventional CNN i.e. fixed-receptive field. Deformable receptive field can also be introduced for the pooling layer in order to allow it to cater for the arbitrary scales and transformations.

Given a feature map F, the top-left corner of ROI (i_0, j_0), and a ROI of size $w \times h$, the ROI-pooling layer converts the ROI to a fixed size of $k \times k$. The ROI-pooling operation can be represented mathematically as:

$$\begin{aligned} &ROI - Pool(F,m,n) \\ &= \sum F(i_0+i, j_0+j)/n_{m,n}, \\ &\forall i = \{1,\ldots,h | \lfloor m\times(h/k)\rfloor \le i < \lceil (m+1)\times(h/k)\rceil\}, \\ &\forall j = \{1,\ldots,w | \lfloor m\times(w/k)\rfloor \le j < \lceil (m+1)\times(w/k)\rceil\} \end{aligned} \tag{3}$$

where $n_{m,n}$ is the number of pixels in the bin (m, n). If there are C input feature maps, the overall output from the layer will be $k \times k \times C$ which will be fed to the classification head.

Deformable ROI-pooling, just like the convolution counterpart, adds an offset to the ROI-pooling layer so that the layer can adapt its receptive field given the input. This can be written as:

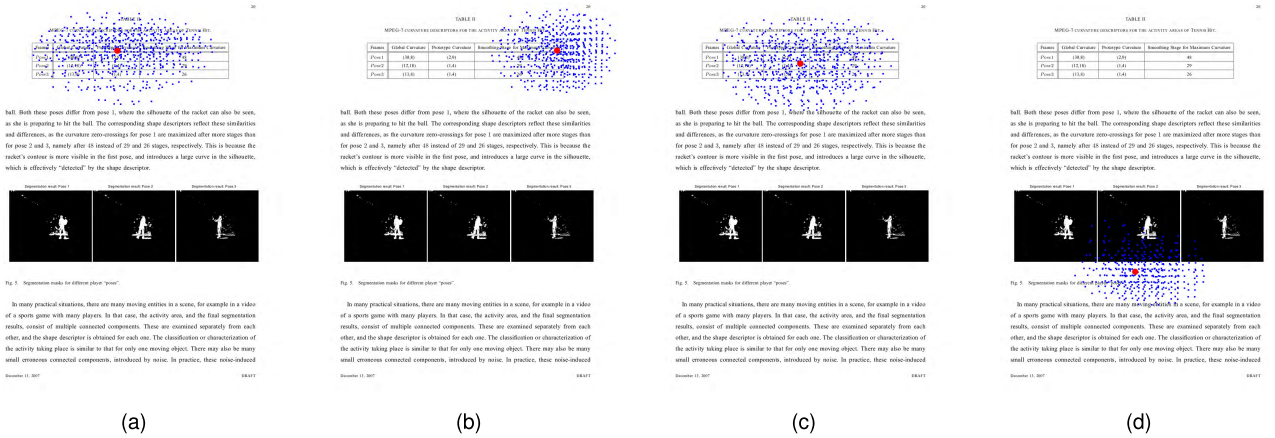
$$\begin{aligned} &DeformableROI - Pool(F,m,n) \\ &= \sum F(i_0+i+\delta^{vertical}_{m,n}, j_0+j+\delta^{horizontal}_{m,n})/n_{m,n}, \\ &\forall i = \{1,\ldots,h | \lfloor m\times(h/k)\rfloor \le i < \lceil (m+1)\times(h/k)\rceil\}, \\ &\forall j = \{1,\ldots,w | \lfloor m\times(w/k)\rfloor \le j < \lceil (m+1)\times(w/k)\rceil\} \end{aligned} \tag{4}$$

where all mutual parameters are the same. $\delta^{vertical}_{i,j}$ defines the vertical offset while $\delta^{horizontal}_{i,j}$ defines the horizontal offset. The offsets in this case can also be fractional and are again implemented via bilinear interpolation.

Since we generate explicit offsets in a deformable convolutional layer for translation of each neuron's receptive field, we visualized the receptive field of a particular deformable convolutional layer in Fig. 3. The red point indicates the filter center while the blue points are obtained after addition of the generated offset. The receptive field of a conventional convolution operation is uniformly distributed on a 2-D grid. On the other hand, in the deformable convolution case, it is evident from the figure that each neuron adapted its receptive field based on its input. The receptive field expanded to cover up the complete table when close to a tabular region (Fig. 3(a), Fig. 3(c)), but remained compact at other locations (Fig. 3(b), Fig. 3(d)).

A. DEFORMABLE ARCHITECTURES

We equipped two different object detection models with deformable convolution for experiments. The first model is a deformable Faster R-CNN which is comprised of a deformable base model along with replacement of conventional ROI-pooling layer with deformable ROI-pooling layer. We refer to this model in the paper as **Model A**. The second model is a deformable Feature Pyramid Network (FPN) which adapts the FPN framework [29]. In deformable FPN, we again use the deformable base model along with replacement of position-sensitive ROI-pooling layer with



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

FIGURE 3. Deformable convolution receptive field at different image locations.

deformable position sensitive ROI-pooling layer. We refer to this model in the paper as **Model B**.

In all our experiments, we used the base model of ResNet-101. Since deformable convolution is a memory intensive operation due to generation of explicit offsets for each location in the feature maps, we just replaced three higher level layers in the ResNet-101 model to convert it into its deformable counterpart (deformable receptive field is primarily helpful for the layers on top of the hierarchy). These layers are *res5a_branch2b*, *res5b_branch2b* and *res5c_branch2b*. For the FPN case, we additionally replaced the layers *res3b3_branch2b* and *res4b22_branch2b* with their deformable counterparts to aid the multi-scale feature extraction.

Since we had insufficient amount of data to train the model from scratch, we leveraged transfer learning to train our model. As we were using deformable ResNet-101, we initialized the offsets for the deformable convolutional layers to zero (zero offsets translate to fixed receptive field making it equivalent to the conventional convolution operation). As the network was fine-tuned on the new dataset, the offsets adapted in order to cope with the scale and transformations of the tabular structures.

It is important to note that the only significant change that we incorporate into the object detection models is the use of deformable base model (deformable ResNet-101) and the use of deformable ROI-pooling instead of the conventional ROI-pooling. This transforms the conventional object detectors to their deformable counterparts. In order to establish a comparison, a ResNet-101 model equipped with the conventional convolution operation was also trained for the task at hand. We refer to this non-deformable model as **Model C**.

B. HYPERPARAMETERS

For training the model A (deformable faster R-CNN), we used three different anchor ratios (0.5, 1 and 2) and five different anchor scales (2, 4, 8, 16 and 32). For training the model B (deformable FPN), we used the same anchor ratios (0.5, 1 and 2) but only one anchor scale (8) since FPN is additionally equipped with a top-down pathway for multi-scale detection. The model was optimized with an initial learning rate of 0.000125 ($\times$ NumGPUs in case of multi-GPU training) for the first 250 iterations. A learning rate of 0.00125 ($\times$ NumGPUs in case of multi-GPU training) was then used with learning rate decay steps at 4, 16, and 32 epochs. The model was optimized for 50 epochs. The maximum image size was limited to 1280 $\times$ 800. Images exceeding this size were resized keeping the aspect ratio intact.

TABLE 1. Datasets.

Dataset	Total Images	Used Images	Train Images	Test Images
ICDAR-13	238	238	-	238
ICDAR-17	2417	2417	1600	817
Mormot	2000	1967	-	-
UNLV*	2889	424	-	-

* Only 424 images contained tabular regions

IV. DATASETS

We used four famous publicly available table detection datasets for our experiments. The details for each dataset are mentioned below and summarized in Table 1.

A. ICDAR-13

ICDAR-2013 [30] is one of the most famous datasets for the task of table detection and structure recognition. The dataset is comprised of PDF files which we converted to images to be used within our system. This was required as our system is only applicable to images as opposed to most other methods which relies on meta-information available in the PDF documents. The dataset also contains structure information for table structure recognition task. The dataset contains 238 images in total. Since most of the prior work on this dataset uses a threshold of 0.5 for IoU to compute the F1-measure [5], we also evaluated our model based on this threshold.

B. ICDAR-17 POD

A recent dataset has been released for a competition (ICDAR-2017 POD)[1] focusing on the task of table, figure and mathematical equation detection from images. The dataset is comprised of 2417 images in total. The training set is comprised of 1600 images while the rest of the 817 images are used for testing. We only evaluated the system for the task of table detection which was the focus of our work. Since all of the submissions in the competition were evaluated for two different IoU thresholds of 0.6 and 0.8, we report our performance on both of these thresholds.

C. MORMOT

The largest publicly available dataset for table recognition is Mormot[2] published by the Institute of Computer Science and Technology (Peking University) and further described in [31]. The total number of images in the dataset is 2000. The ratio of positive to negative images is approximately 1:1 for both sets. There were many instances of incorrect ground-truth annotations in the dataset. Therefore, we used the cleaned out version of the dataset for our experiments by Schreiber *et al.* [5] (2017). The cleaned version of the dataset was comprised of 1967 images which we used in our experiments.

D. UNLV

UNLV dataset is comprised of a variety of documents which includes technical reports, business letters, newspapers and magazines etc. The dataset contains a total of 2889 scanned documents where only 424 documents contains a tabular region. We only used the images containing a tabular region in our experiments.

V. EVALUATION

The performance of deep learning models improves with the increase in the amount of data [33]. Since all of the evaluated methods were data-driven, there was a need for a sufficiently large dataset. In order to improve the generalization capabilities of the system, we combined different available datasets for training our models. To validate the generalization capabilities, we performed testing by following the leave-one-out scheme. We left out a complete specific dataset at the time of training and used that complete dataset for testing except for ICDAR-17 where we used its training set in all of our experiments.

Following the evaluation protocol defined for ICDAR-17 POD, we first compute the number of true positives, false positives and false negatives from the entire test set. These numbers are then in turn used to compute the precision, recall and F-Measure which are the most commonly reported metrics in the literature. This evaluation scheme is different from [5], [30] where they first compute the precision, recall and F-Measure for every document followed by the averaging over the entire test set. However, in our case, the numbers have negligible difference for the ICDAR-13 dataset, therefore, we stick to the ICDAR-17 evaluation scheme which is much more general.

Table 2 compares the performance of the proposed approach with the prior work on ICDAR-2017 POD and ICDAR-2013 datasets. For the sake of completion, we also report results on UNLV and Mormot but these datasets were not a focus of our work. It is important to mention that the systems relying on PDF documents are not directly comparable with our system, as they make use of meta-data included in the PDF files, while our approach relies on just raw images with no additional meta-data. This makes the problem much more challenging. For the sake of completion, we also report the results from these methods.

We performed 7 scale testing where we perform the detection at 7 different scales (3 smaller scales, the original scale as well as 3 larger scales). We kept all detections having at least three votes from the 7 different scales. This helped in mitigating simple false positives. The initial detections at every scale were thresholded at a confidence of 0.75 followed by the combination. The final detections obtained after voting were thresholded at a confidence of 0.95. The red bounding box highlights the detected tabular region while the green bounding box highlights the ground-truth in all figures.

A. ICDAR-13

The ICDAR-2013 dataset is comprised of 238 images containing 156 tables. We used all of the images in the dataset for testing without any of them being used in the training. The system was able to correctly identify all of the tabular regions present in the dataset except one resulting in 99.4% recall. Similarly, the system incorrectly labeled only one region as belonging to tables (false positive) resulting in a precision of 99.4%. Representative examples of correct and incorrect detections from ICDAR-13 dataset including true positives, false positives, and false negatives are presented in Fig. 4. With the achieved F-Measure of 99.4%, we were able to outperform the previous state-of-the-art method on ICDAR-2013 dataset comprehensively.

Schreiber *et al.* [5] used a Faster R-CNN based approach which is based on the conventional convolution operation. Since their backbone is based on ZFNet [20] and VGG-16 [21], their model is not directly comparable. Therefore, we added experimental results for model C which is equipped with conventional convolution operation with the same ResNet-101 backbone. The results clearly indicates that deformable convolution comprehensively outperforms its conventional counterpart.

B. ICDAR-17 POD

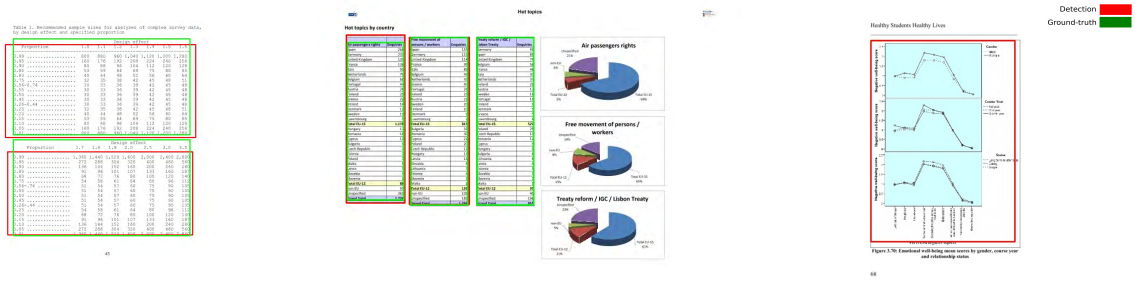
The ICDAR-2017 POD challenge was comprised of 817 images containing 317 tables. All the entries in the competition were evaluated on two different IoU thresholds of 0.6 and 0.8 for calculation of the relevant metrics.

[1] http://www.icst.pku.edu.cn/cpdp/ICDAR2017_PODCompetition/index.html

[2] http://www.icst.pku.edu.cn/cpdp/data/marmot_data.htm

TABLE 2. Table detection performance on different datasets.

Dataset	System Input	IoU Threshold	System	Recall	Precision	F1-measure
ICDAR-2017 (Test set) [817 Images]	**Images**	**0.6**	**Ours (Model A)**	**0.971**	**0.965**	**0.968**
			(Model B)	0.946	0.977	0.961
			(Model C)	0.940	0.903	0.921
			NLPR-PAL	0.953	0.968	0.960
			School of Software	0.940	0.934	0.937
			VisInt	0.918	0.924	0.921
			maitai-ee	0.890	0.842	0.865
			icstpku	0.773	0.857	0.813
			UITVN	0.940	0.670	0.782
			IU-vision	0.221	0.230	0.225
			HustVision	0.959	0.071	0.132
		0.8	**Ours** (Model A)	0.952	0.946	0.949
			(Model B)	**0.937**	**0.967**	**0.952**
			(Model C)	0.915	0.879	0.896
			NLPR-PAL	0.958	0.943	0.951
			VisInt	0.823	0.829	0.826
			School of Software	0.798	0.793	0.796
			maitai-ee	0.798	0.755	0.776
			icstpku	0.726	0.804	0.763
			UITVN	0.763	0.544	0.635
			IU-vision	0.114	0.118	0.116
			HustVision	0.836	0.062	0.115
ICDAR-2013 (Complete Set) [238 Images]	**Images**	**0.5**	**Ours** (Model A)	1.000	0.945	0.972
			(Model B)	**0.996**	**0.996**	**0.996**
			(Model C)	1.000	0.939	0.969
			Kavasidis et al. (2018) [7]	0.981	0.975	0.978
			Sebestian et al. (2017) [5]	0.962	0.974	0.968
			Tran et al. (2017) [18]	0.967	0.952	0.958
	PDF	0.5	Hao et al. [19]	0.922	0.972	0.946
			Silva [15]	0.9831	0.9292	0.9554
			Nurminen [30]	0.9077	0.9210	0.9143
			Yildiz [32]	0.8530	0.6399	0.7313
Mormot (Complete Set) [1967 Images]	**Images**	**0.5**	Ours (Model A)	0.946	0.849	0.895
UNLV (Complete Set) [424 Images]	**Images**	**0.5**	Ours (Model A)	0.749	0.786	0.767



(a) (b) (c)

FIGURE 4. DeCNT results on the ICDAR-13 Table competition dataset. (a) True Positive. (b) False Negative. (c) False Positive.

The deformable Faster R-CNN (model A) performed well on this task for an IoU threshold of 0.6 achieving 96.8% F-Measure with 97.1% recall and 96.5% precision. The deformable FPN (model B) achieved state-of-the-art results for an IoU threshold of 0.8 achieving F-Measure of 95.3% with 93.1% recall and 97.7% precision. Representative

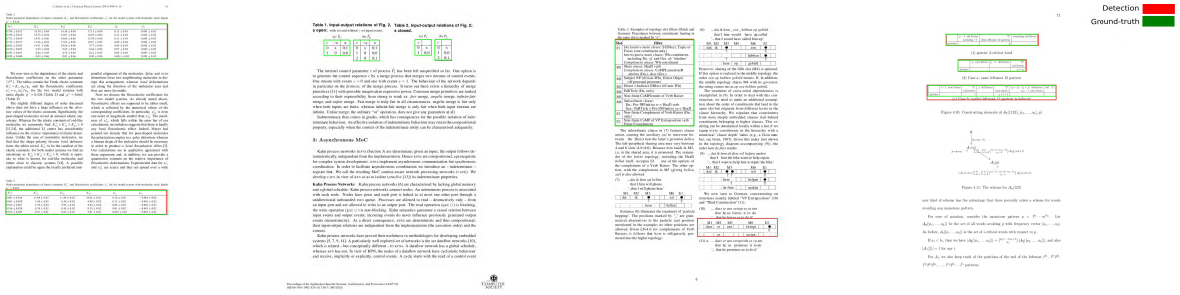


FIGURE 5. DeCNT results on the ICDAR-17 POD dataset. (a) True Positive. (b) False Negative. (c) False Positive. (d) Low IoU.

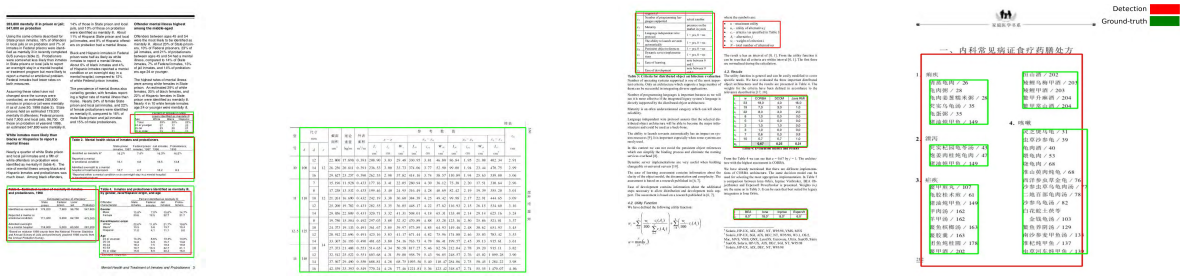


FIGURE 6. DeCNT results on the Mormot dataset. (a) True Positive. (b) False Negative. (c) False Positive. (d) Under-Segmented.

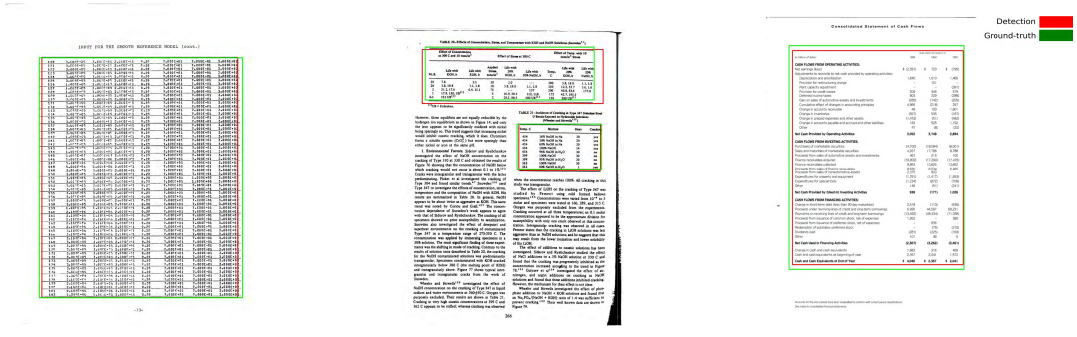


FIGURE 7. DeCNT correct detection results on the UNLV dataset.

examples of correct and incorrect detections from ICDAR-17 POD dataset including true positives, false positives, and false negatives are presented in Fig. 5. With the achieved results, we were able to outperform all other ICDAR-2017 POD challenge participants on the task of table detection for both IoU thresholds of 0.6 and 0.8.

Analysis of the incorrect results on ICDAR-2017 revealed that most of the mistakes were related to IoU. The reason being that different datasets combined had different annotations in terms of distances to the table border. To the extreme, there were some cases where empty cells within the table were not considered a part of the tabular region.

We again compared our results with the conventional convolutional counterpart. The difference between deformable and its conventional counterpart was also consistent in this case.

C. MORMOT

The Mormot dataset is comprised of 1967 images containing a total of 1348 tables. The deformable Faster R-CNN trained on the rest of the three datasets except Mormot was able to correctly detect 1275 table instances. The system also produced 226 false positives and 73 false negatives resulting in recall of 94.6% and precision of 84.9%. This led to the final F-Measure of 89.5%. Representative examples of correct and incorrect detections from Mormot dataset including true positives, false positives, and false negatives are presented in Fig. 6.

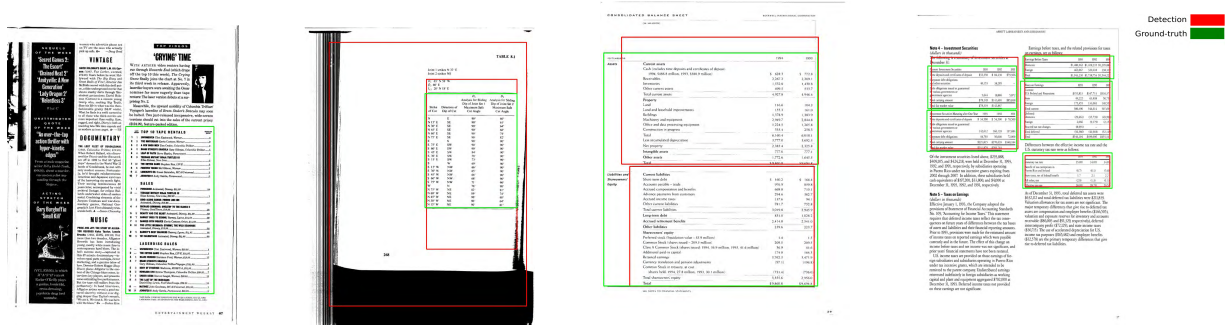


FIGURE 8. DeCNT incorrect detection results on the UNLV dataset. (a) False Negative. (b) False Positive. (c) Over-Segmented. (d) Under-Segmented.

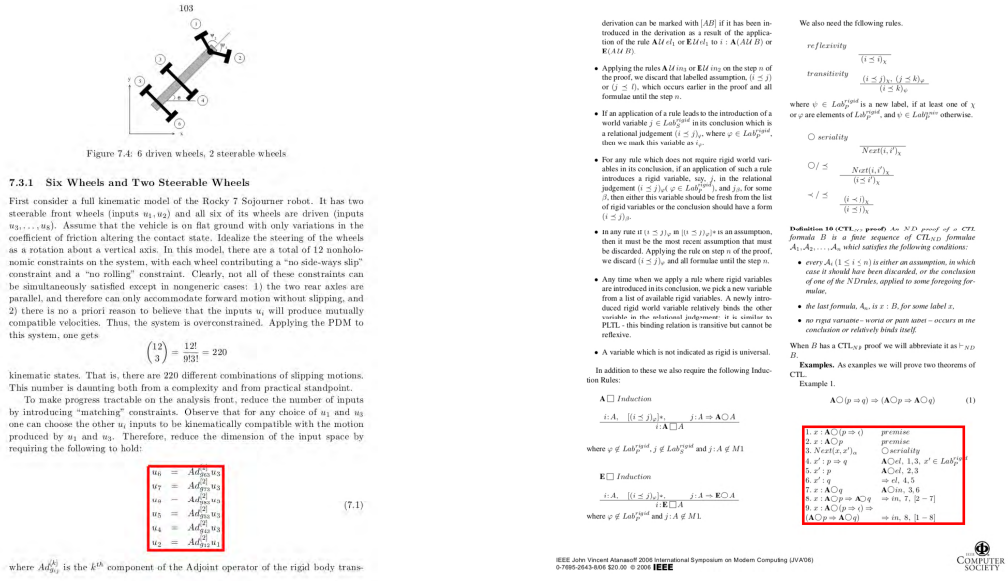


FIGURE 9. Confusion between tabular structure and structurally laid out equations.

D. UNLV

UNLV dataset in the same way is comprised of 424 images containing a total of 558 tables. The deformable Faster R-CNN trained in the same leave-one-out scheme was able to correctly detect 418 table instances. The system also produced 114 false positives and 140 false negatives resulting in recall of 74.9%, precision of 78.6%, and final F-Measure of 76.7%. Examples of correctly classified tabular regions for UNLV are presented in Fig. 7 while the incorrectly classified tabular regions are visualized in Fig. 8.

Manual inspection of the failure cases on all datasets revealed ambiguity between tables and figures in some cases due to high overlap in terms of layout. There were even confusions in cases where the equations were laid out in a sequential order. Two sample cases of such ambiguity are visualized in Fig. 9. In order to mitigate these issues, we also utilized the annotations provided in the ICDAR-2017 POD dataset for formula and figures to train the network to disambiguate between these distinct entities. Therefore, all the results that we report uses these formula and figure annotations.

With the obtained results, it is evident that the system was able to detect tables with arbitrary layouts through the proposed data-driven methodology. The system was evaluated in a leave-one-out scheme to test the situations which are encountered in the real-world. The model generalized well to a wide range of different documents types indicating its generality.

VI. CONCLUSION

This paper presented a novel end-to-end table detection method based on region-based deformable convolutional neural networks. It is evident from the extensive evaluation of the proposed methodology that the deep architectures developed for the task of object detection in natural scenes supplemented with the deformable property can outperform their non-deformable counterparts comprehensively.

Our extensive evaluation on the publicly available datasets of ICDAR-2013, ICDAR-2017 POD, UNLV and Mormont shows superior or comparable performance to existing methods, even though in some cases, the different methods are not directly comparable due to extensive use of PDF metadata, or different use of evaluation metrics.

A promising direction for future work could be to evaluate the performance of the deformable convolutional neural networks for table structure recognition which is much more challenging due to high intra-class variability and poor data availability. The structure recognition task is essential in terms of its applicability in business and finance due to presence of a large amount of tabular data in documents. Due to lack of available datasets in that direction, a very significant contribution could be made by introducing new datasets for the structure recognition task. The deformable network could also be supplemented with CoordConv [34] introduced very recently to improve network's localization capabilities by explicitly adding the location coordinates.

REFERENCES

[1] B. Coüasnon and A. Lemaitre, "Recognition of tables and forms," in *Handbook of Document Image Processing and Recognition*, D. Doermann and K. Tombre, Eds. London, U.K.: Springer, 2014, pp. 647–677.
[2] D. W. Embley, M. Hurst, D. Lopresti, and G. Nagy, "Table-processing paradigms: a research survey," *Int. J. Document Anal. Recognit.*, vol. 8, nos. 2–3, pp. 66–86, 2006.
[3] P. Pyreddy and W. B. Croft, "TINTIN: A system for retrieval in text tables," in *Proc. 2nd ACM Int. Conf. Digit. Libraries*, Philadelphia, PA, USA, Jul. 1997, pp. 193–200, doi: 10.1145/263690.263816.
[4] E. Oro and M. Ruffolo, "PDF-TREX: An approach for recognizing and extracting tables from PDF documents," in *Proc. 10th Int. Conf. Document Anal. Recognit. (ICDAR)*, Barcelona, Spain, Jul. 2009, pp. 906–910, doi: 10.1109/ICDAR.2009.12.
[5] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed, "DeepDeSRT: Deep learning for detection and structure recognition of tables in document images," *Proc. ICDAR*, Nov. 2017, pp. 1162–1167.
[6] A. Gilani, S. R. Qasim, I. Malik, and F. Shafait, "Table detection using deep learning," in *Proc. ICDAR*, Nov. 2017, pp. 771–776.
[7] I. Kavasidis *et al.* (Apr. 2018). "A saliency-based convolutional neural network for table and chart detection in digitized documents." [Online]. Available: https://arxiv.org/abs/1804.06236
[8] T. Kieninger and A. Dengel, "Applying the T-Recs table recognition system to the business letter domain," in *Proc. 6th Int. Conf. Document Anal. Recognit. (ICDAR)*, Seattle, WA, USA, Sep. 2001, pp. 518–522, doi: 10.1109/ICDAR.2001.953843.
[9] A. Pivk, P. Cimiano, Y. Sure, M. Gams, V. Rajkovič, and R. Studer, "Transforming arbitrary tables into logical form with TARTAR," *Data Knowl. Eng.*, vol. 60, no. 3, pp. 567–595, 2007, doi: 10.1016/j.datak.2006.04.002.
[10] R. Zanibbi, D. Blostein, and R. Cordy, "A survey of table recognition: Models, observations, transformations, and inferences," *Int. J. Document Anal. Recognit.*, vol. 7, no. 1, pp. 1–16, 2004.
[11] A. C. e Silva, A. M. Jorge, and L. Torgo, "Design of an end-to-end method to extract information from tables," *Int. J. Document Anal. Recognit.*, vol. 8, nos. 2–3, pp. 144–171, 2006.
[12] S. Khusro, A. Latif, and I. Ullah, "On methods and tools of table detection, extraction and annotation in PDF documents," *J. Inf. Sci.*, vol. 41, no. 1, pp. 41–57, 2015.
[13] F. Cesarini, S. Marinai, L. Sarti, and G. Soda, "Trainable table location in document images," in *Proc. 16th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2002, pp. 236–240.
[14] A. C. e Silva, "Learning rich hidden Markov models in document analysis: Table location," in *Proc. 10th Int. Conf. Document Anal. Recognit. (ICDAR)*, 2009, pp. 843–847.
[15] A. C. e Silva, "Parts that add up to a whole: A framework for the analysis of tables," Ph.D. dissertation, School Inform., Univ. Edinburgh, Scotland, U.K., 2010.
[16] T. Kasar, P. Barlas, S. Adam, C. Chatelain, and T. Paquet, "Learning to detect tables in scanned document images using line information," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 1185–1189.
[17] M. Fan and D. S. Kim. (2015). "Detecting table region in PDF documents using distant supervision." [Online]. Available: https://arxiv.org/abs/1506.08891
[18] D. N. Tran, T. A. Tran, A.-R. Oh, S.-H. Kim, and I.-S. Na, "Table detection from document image using vertical arrangement of text blocks," *Int. J. Contents*, vol. 11, no. 4, pp. 77–85, 2015.
[19] L. Hao, L. Gao, X. Yi, and Z. Tang, "A table detection method for PDF documents based on convolutional neural networks," in *Proc. 12th IAPR Workshop Document Anal. Syst. (DAS)*, Apr. 2016, pp. 287–292.
[20] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision—ECCV* (Lecture Notes in Computer Science), D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 818–833.
[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–14.
[22] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.
[23] F. Yu and V. Koltun. (Nov. 2015). "Multi-scale context aggregation by dilated convolutions." [Online]. Available: https://arxiv.org/abs/1511.07122
[24] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Piscataway, NJ, USA, Dec. 2015, pp. 1440–1448.
[25] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," in *Proc. Int. Conf. Mach. Learn. Deep Learn. Workshop (ICML)*, 2015, pp. 1–12.
[26] J. Dai *et al.*, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 764–773.
[27] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2016, pp. 379–387.
[28] S. Ren, K. He, R. B. Girshick, and J. Sun. (2015). "Faster R-CNN: Towards real-time object detection with region proposal networks." [Online]. Available: https://arxiv.org/abs/1506.01497
[29] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. (2016). "Feature pyramid networks for object detection." [Online]. Available: https://arxiv.org/abs/1612.03144
[30] M. C. Göbel, T. Hassan, E. Oro, and G. Orsi, "ICDAR 2013 table competition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 1449–1453.
[31] J. Fang, X. Tao, Z. Tang, R. Qiu, and Y. Liu, "Dataset, ground-truth and performance metrics for table detection evaluation," in *Proc. 10th IAPR Int. Workshop Document Anal. Syst. (DAS)*, M. Blumstein, U. Pal, and S. Uchida, Eds., Mar. 2012, pp. 445–449.
[32] B. Yildiz, K. Kaiser, and S. Miksch, "pdf2table: A method to extract table information from PDF files," in *Proc. 2nd Indian Int. Conf. Artif. Intell. (IICAI)*, B. Prasad, Ed., 2005, pp. 1773–1785.
[33] C. Sun, A. Shrivastava, S. Singh, and A. Gupta. (2017). "Revisiting unreasonable effectiveness of data in deep learning era." [Online]. Available: https://arxiv.org/abs/1707.02968
[34] R. Liu *et al.* (Jul. 2018). "An intriguing failing of convolutional neural networks and the coordconv solution." [Online]. Available: https://arxiv.org/abs/1807.03247

SHOAIB AHMED SIDDIQUI received the bachelor's degree in computer science from the National University of Sciences and Technology, Pakistan, in 2016, and the M.S. degree from the University of Kaiserslautern. He is currently pursuing the Ph.D. degree with the German Research Center for Artificial Intelligence (DFKI GmbH) and the University of Kaiserslautern, under supervision of Dr. Prof. H. C. A. Dengel. His areas of interests include the interpretability and robustness of deep learning models (including adversarial examples and defenses), document understanding, time-series analysis, and extreme classification. He is also a Reviewer for the *ICES Journal of Marine Science* and IEEE Access.

MUHAMMAD IMRAN MALIK received the bachelor's and the master's degrees in computer science from the University of Kaiserslautern, Germany, and the Ph.D. degree with the German Research Center for Artificial Intelligence, Germany, under the supervision of Dr. H. C. A. Dengel and Dr. H. M. Liwicki. In bachelor's thesis, he worked in the domains of real time object detection and image enhancement. In master's thesis, he primarily worked for the development of various systems for signature identification and verification. His Ph.D. topic was automated forensic handwriting analysis on which he focused on both the perspectives of forensic handwriting examiners and pattern recognition researchers. He is currently an Assistant Professor with the National University of Sciences and Technology, Pakistan. He has more than 35 publications on the said and related topics including two journal papers.

STEFAN AGNE received the Diploma degree in computer science and the Ph.D. degree under the supervision of Dr. A. Dengel from the University of Kaiserslautern, Germany, in 1995, and the Ph.D. degree from the University of Bern, in 2008, under the supervision of Dr. H. Bunke. Since 1992, he has been working in document analysis and understanding with the German Research Center for Artificial Intelligence (DFKI GmbH), Germany. He is currently leading the topic area pattern recognition with the Smart Data and Knowledge Services Department, German Research Center for Artificial Intelligence, Kaiserslautern.

ANDREAS DENGEL received the Diploma degree in computer science from the University of Kaiserslautern and the Ph.D. from the University of Stuttgart. In 1993, he became a Professor with the Computer Science Department, University of Kaiserslautern, where he currently holds the Chair of knowledge-based systems. Since 2009, he has been a Professor (Kyakuin) with the Department of Computer Science and Information Systems, Osaka Prefecture University. He was with IBM, Siemens, and Xerox PARC. He is currently the Scientific Director of the German Research Center for Artificial Intelligence (DFKI GmbH), Kaiserslautern. He has authored more than 300 peer-reviewed scientific publications and supervised more than 170 Ph.D. and master's theses. His main scientific emphases are in the areas of pattern recognition, document understanding, information retrieval, multimedia mining, semantic technologies, and social media. He is a member of several international advisory boards, chaired major international conferences, and founded several successful start-up companies. Moreover, he is the co-editor of international computer science journals and has written or edited 12 books. He is an IAPR Fellow and received prominent international awards.

SHERAZ AHMED received the master's degree in computer science from the University of Kaiserslautern, Germany, and the Ph.D. degree with the German Research Center for Artificial Intelligence (DFKI GmbH), Kaiserslautern, Germany, under the supervision of Dr. H. C. A. Dengel and Dr. H. M. Liwicki. His Ph.D. topic was generic methods for information segmentation in document images. Over the last few years, he had primarily worked for the development of various systems for information segmentation in document images. From 2012 to 2013, he visited Osaka Prefecture University, Osaka, Japan, as a Research Fellow, supported by the Japanese Society for the Promotion of Science. In 2014, he visited the University of Western Australia, Perth, Australia, as a Research Fellow, supported by DAAD, Germany, and Group of Eight, Australia. He is currently a Senior Researcher with the German Research Center for Artificial Intelligence, where he is also leading the areas of time series and document analysis. His research interests include document understanding, generic segmentation framework for documents, gesture recognition, pattern recognition, data mining, anomaly detection, and natural language processing. He has more than 30 publications on his research-related topics, including three journal papers and two book chapters. He is a frequent Reviewer of various journals, including *Patter Recognition Letters*, *Neural Computing and Applications*, and IJDAR. He is also a frequent Reviewer of conferences, including ICDAR, ICFHR, and DAS.