Article

CasTabDetectoRS: Cascade Network for Table Detection in Document Images with Recursive Feature Pyramid and Switchable Atrous Convolution

Khurram Azeem Hashmi [1,2,3,*], Alain Pagani[3], Marcus Liwicki[4], Didier Stricker [1,3] and Muhammad Zeshan Afzal [1,2,3*]

[1] Department of Computer Science, Technical University of Kaiserslautern, 67663 Kaiserslautern, Germany; khurram_azeem.hashmi@dfki.de (K.A.H.); muhammad_zeshan.afzal@dfki.de (M.Z.A.); alain.pagani@dfki.de (A.P.); didier.stricker@dfki.de (D.S.);
[2] Mindgarage, Technical University of Kaiserslautern, 67663 Kaiserslautern, Germany
[3] German Research Institute for Artificial Intelligence (DFKI), 67663 Kaiserslautern, Germany,
[4] Department of Computer Science, Luleå University of Technology, 971 87 Luleå, Sweden; marcus.liwicki@ltu.se (M.L.);
* Correspondence: khurram_azeem.hashmi@dfki.de

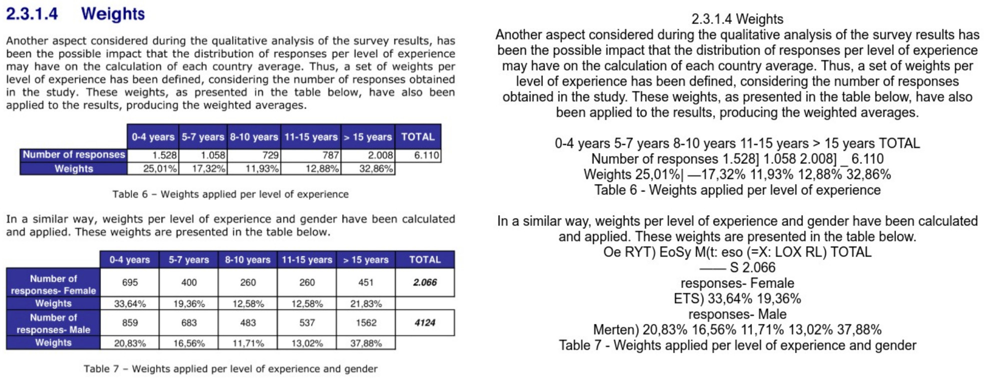
Abstract: Table detection is a preliminary step in extracting reliable information from tables in scanned document images. We present CasTabDetectoRS, a novel end-to-end trainable table detection framework that operates on Cascade Mask R-CNN, including Recursive Feature Pyramid network and Switchable Atrous Convolution in the existing backbone architecture. By utilizing a comparatively lightweight backbone of ResNet-50, this paper demonstrates that superior results are attainable without relying on pre and post-processing methods, heavier backbone networks (ResNet-101, ResNeXt-152), and memory-intensive deformable convolutions. We evaluate the proposed approach on five different publicly available table detection datasets. Our CasTabDetectoRS outperforms the previous state-of-the-art results on four datasets (ICDAR-19, TableBank, UNLV, and Marmot) and accomplishes comparable results on ICDAR-17 POD. Upon comparing with previous state-of-the-art results, we obtain a significant relative error reduction of 56.36%, 20%, 4.5%, and 3.5% on the datasets of ICDAR-19, TableBank, UNLV, and Marmot, respectively. Furthermore, this paper sets a new benchmark by performing exhaustive cross-datasets evaluations to exhibit the generalization capabilities of the proposed method.

Keywords: table detection; table recognition; cascade Mask R-CNN; atrous convolution; recursive feature pyramid networks; document image analysis; deep neural networks; computer vision, object detection.

1. Introduction

The process of digitizing documents has received significant attention in various domains such as industrial, academic, and commercial sectors. The digitization of documents facilitates the process of extracting information without manual intervention. Apart from the text, documents contain graphical page objects such as tables, figures, and formulas [1,2]. Albeit modern Optical Character Recognition (OCR) systems [3–5] can extract the information from scanned documents, they fail to interpret information from graphical page objects [6–9]. Figure 1 exhibits the problem of extracting tabular information from a document by applying open-source Tesseract OCR [10]. It is evident that even the state-of-the-art OCR system fail to parse information from tables in document images. Therefore, for complete table analysis, it is essential to develop accurate table detection systems for document images.

The problem of accurate table detection in document images is still an open problem in the research community [8,11–14]. The high amount of intra-class variance (arbitrary layouts of tables, varying presence of ruling lines) and low amount of inter-class variance (figures, charts, and algorithms equipped with horizontal and vertical lines look alike tables) makes the task of classifying and localizing tables in document images even more challenging. Owing to these involved intricacies in table detection, custom heuristics based methods lack in producing robust solutions [15,16].



2.3.1.4 Weights

Another aspect considered during the qualitative analysis of the survey results, has been the possible impact that the distribution of responses per level of experience may have on the calculation of each country average. Thus, a set of weights per level of experience has been defined, considering the number of responses obtained in the study. These weights, as presented in the table below, have also been applied to the results, producing the weighted averages.

	0-4 years	5-7 years	8-10 years	11-15 years	> 15 years	TOTAL
Number of responses	1.528	1.058	729	787	2.008	6.110
Weights	25,01%	17,32%	11,93%	12,88%	32,86%	

Table 6 – Weights applied per level of experience

In a similar way, weights per level of experience and gender have been calculated and applied. These weights are presented in the table below.

	0-4 years	5-7 years	8-10 years	11-15 years	> 15 years	TOTAL
Number of responses- Female	695	400	260	260	451	***2.066***
Weights	33,64%	19,36%	12,58%	12,58%	21,83%	
Number of responses- Male	859	683	483	537	1562	***4124***
Weights	20,83%	16,56%	11,71%	13,02%	37,88%	

Table 7 – Weights applied per level of experience and gender

```
2.3.1.4 Weights
Another aspect considered during the qualitative analysis of the survey results has
been the possible impact that the distribution of responses per level of experience
may have on the calculation of each country average. Thus, a set of weights per
level of experience has been defined, considering the number of responses
obtained in the study. These weights, as presented in the table below, have also
been applied to the results, producing the weighted averages.

0-4 years 5-7 years 8-10 years 11-15 years > 15 years TOTAL
Number of responses 1.528] 1.058 2.008] _ 6.110
Weights 25,01%| —17,32% 11,93% 12,88% 32,86%
Table 6 - Weights applied per level of experience

In a similar way, weights per level of experience and gender have been calculated
and applied. These weights are presented in the table below.
Oe RYT) EoSy M(t: eso (=X: LOX RL) TOTAL
—— S 2.066
responses- Female
ETS) 33,64% 19,36%
responses- Male
Merten) 20,83% 16,56% 11,71% 13,02% 37,88%
Table 7 - Weights applied per level of experience and gender
```

Input Document Image

Extracted information from OCR

Figure 1. Illustrating the need of applying table detection before extracting information in document images. We apply open source Tesseract-OCR [10] on a document image containing two tables. Besides the textual content, the OCR system fails miserably in interpreting information from tables.

Prior works have tackled the involved challenges of table detection through leveraging meta-data or utilizing morphological information from tables. However, these methods are vulnerable in case of scanned document images [17,18]. Later, the utilization of deep learning-based approaches to attempt the task of table detection in document images have shown a remarkable improvement in the past few years [8]. Intuitively, the task of table detection has been formulated as an object detection problem [7,19–21], in which, a table can be a targeted object present in a document image instead of a natural scene image. Consequently, the rapid progress in object detection algorithms have led to the extraordinary improvement in state-of-the-art table detection systems [11–13,20]. However, the prior approaches struggle in predicting precise localization of tabular boundaries in distinctive datasets. Moreover, they either rely on external pre/post-processing methods to further refine their predictions [11,13] or incorporate memory intensive deformable convolutions [12,20]. Furthermore, prior state-of-the-art methods relied on heavy and high resolution backbones such as ResNext-101 [22] and HRNet [23] which require expensive process of training.

To tackle the above mentioned issues present in existing approaches, we present CasTabDetectoRS, an end-to-end trainable novel object detection pipeline by incorporating the idea of Recursive Feature Pyramids (RFP) and Switchable Atrous Convolutions (SAC) [24] into Cascade Mask R-CNN [25] for detection of tables in document images. Furthermore, this paper empirically establishes that generic and robust table detection systems can be built without depending on pre/post-processing methods and heavy backbone networks.

To summarize, the main contribution of this work are explained below:

- We present CasTabDetectoRS, a novel deep learning-based table detection approach that operates on Cascade Mask R-CNN equipped with recursive feature pyramid and switchable atrous convolution.
- We experimentally deny the dependency of custom heuristics or heavier backbone networks to achieve superior results on table detection in scanned document images.

- We accomplish state-of-the-art results on four publicly available table detection datasets of ICDAR-19, TableBank, Marmot, and UNLV (See Table 2, 3, 4, and 5).
- We demonstrate the generalization capabilities of the proposed CasTabDetectoRS by performing the exhaustive cross-datasets evaluation.

The remaining paper is structured as follows. Section 2 categorizes the prior literature into rule-based, learning-based, and object detection-based methods. Section 3 describes the proposed table detection pipeline by addressing all the essential modules such as RFP (Section 3.1), SAC (Section 3.2), and Cascade Mask R-CNN (Section 3.3). Section 4 presents the comprehensive overview of employed datasets, experimental details, evaluation criteria, along with quantitative and qualitative analysis that follows with a comparison with previous state-of-the-art results and cross datasets evaluation. Section 5 concludes the paper and outlines possible future directions.

2. Related work

The problem of table detection in documents has been investigated over the past few decades [16,26]. Earlier, researchers employed rule-based systems to solve table detection [16,26–29]. Afterwards, researchers exploited statistical learning mainly machine learning-based approaches which are eventually replaced with deep learning-based methods [7,8,11,12,19,20,30–34].

2.1. Rule-Based Methods

To the best of our knowledge, Itonori et al. [26] addressed the problem of table detection in document images by employing a rule-based method. The proposed approach leveraged the arrangements of text-blocks and position of ruling lines to detect tables in documents. Chandran and Kasturi [27] proposed another method that operates on ruling lines to resolve table detection. Similarly, Pyreddy and Croft [35] published a heuristics-based table detection method that first identify structural elements from a document and then filters the table.

Researchers have defined tabular layouts and grammars to detect tables in documents [29,36]. The correlation of white spaces and vertical connected component analysis is employed to predict tables [37]. Another method that transforms tables present in HTML documents into a logical structure is proposed by Pivk et al.[36]. Shigarov et al. [18] capitalized the meta-data from PDF files and treated each word as a block of text. The proposed method restructured the tabular boundaries by leveraging bounding boxes of each word.

We direct our readers to [15,16,38–40] for the thorough understanding of these rule-based methods. Although the prior rule-based systems detect tables in document having limited patterns, they rely on manual intervention to look for optimal rules. Furthermore, they are vulnerable in producing generic solutions.

2.2. Learning-Based Methods

Similar to the field of computer vision, the domain of table analysis have experienced a notable progress after incorporating learning-based methods. Initially, researchers investigate machine learning-based methods to resolve table detection in document images. Unsupervised learning was implemented by Kieninger and Dengel [41] to improve table detection in documents. Later, Cesarini et al. [42] employed supervised learning-based system to find tables in documents. Their system reforms document into MXY tree representation. later, the method predicts the tables by searching for blocks that are surrounded with ruling lines. Kasar et al. [43] proposed a blend of SVM classifier and custom heuristics [43] to resolve table detection in documents. Researchers have also explored the capabilities of Hidden Markov Models (HMMs) to localize tabular areas in documents [44,45]. Even though machine learning-based approaches have alleviated the research for table detection in documents, they require external meta-data to execute reliable predictions. Moreover, they fail to obtain generic solutions on document images.

Analogous to the field of computer vision, the power of deep learning has made a remarkable impact in the field of table analysis in document images [2,8]. To the best of our knowledge, Hao et al. [46] introduced the idea of implementing Convolutional Neural Network (CNN) to identify spatial features from document images. The authors merged these features with the extracted meta-data to predict tables in PDF documents.

Although researchers have employed Fully Convolutional Network (FCN) [47,48] and Graph Neural Network (GNN) [34,49] to perform table detection in document images, object detection-based approaches [7,8,11,12,19,20,30–34] have delivered state-of-the-art results.

2.3. *Table Detection as an Object Detection Problem*

There has been a direct relationship with the progress of object detection networks in computer vision and table detection in document images [8]. Gilani et al. [19] formulated the problem of table detection as an object detection problem by applying Faster R-CNN [50] to detect tables in document images. The presented work employed distance transform methods to modify pixels in raw document images fed to the Faster R-CNN.

Later, another Schreiber et al. [7] presented another method that exploits Faster R-CNN [50] equipped with pre-trained base networks (ZFNet [51] and VGG-16 [52]) to detect tables in document images. Furthermore, Siddiqui et al. [20] published another Faster R-CNN-based method equipped with deformable convolutions [53] to address table detection having arbitrary layouts. Moreover, in [33], the authors employed Faster R-CNN with a coroner locating an approach to improve the predicted tabular boundaries in document images.

Saha et al. [54] empirically established that Mask R-CNN [55] produces better results as compared to Faster R-CNN [50] in detecting tables, figures, and formulas. Zhong et al. [56] presented a similar conclusion by applying Mask R-CNN to localize tables. Moreover YOLO [57], SSD [58], and RetinaNet [59] have been employed to exhibit the benefits of closed domain fine-tuning on table detection in document images.

Recently, researchers have incorporated novel object detection algorithms like Cascade Mask R-CNN [25] and Hybrid Task Cascade (HTC) [60] to alleviate the performance of table detection systems in document images [11–14]. Although these prior methods have progressed state-of-the-art results, there is significant room for improvement in localizing accurate tabular boundaries in scanned document images.

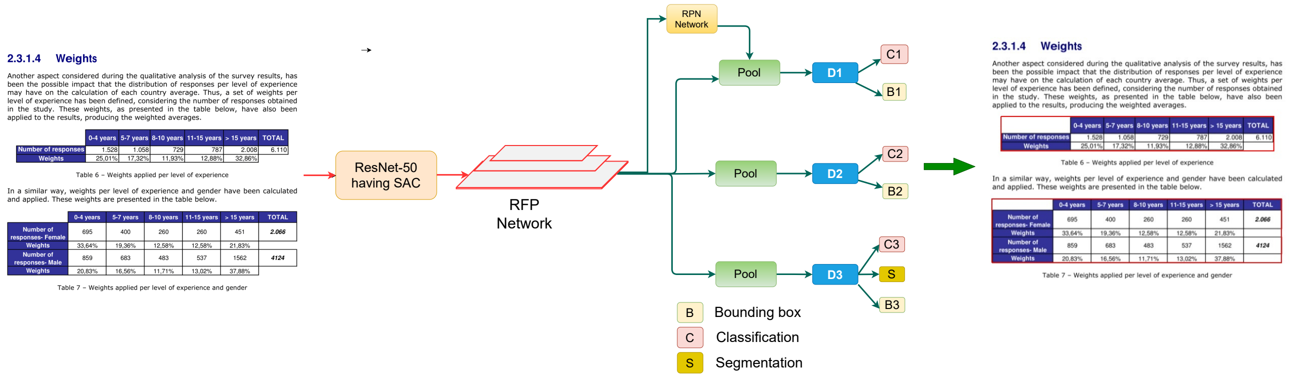
3. Method

The presented approach incorporates RFP and SAC into a Cascade Mask R-CNN to attempt table detection in scanned document images as exhibited in Figure 2. Section 3.1 discusses the RFP module, whereas Section 3.2 talks about SAC module. Section 3.3 describes the employed Cascade Mask R-CNN along with complete description of the proposed pipeline.

3.1. *Recursive Feature Pyramids*

Instead of the traditional Feature Pyramid Networks (FPN) [61] in our table detection framework, we incorporate Recursive Feature Pyramids (RFP) [24] to improve the processing of feature maps. To understand the conventional FPN, let N_j denote the j-th stage of a bottom-up backbone network and F_j represent the j-th top-down FPN function. The backbone network N having FPN produces a set of feature maps, where total feature maps are equal to the number of stages. For instance, a backbone network with three stages is demonstrated in Figure 3. Therefore, with a number of stages $S = 3$, the output feature f_j is given by:

$$f_j = F_j(f_{j+1}, i_j), \quad i_j = N_j(i_{j-1}) \tag{1}$$



2.3.1.4 Weights

Another aspect considered during the qualitative analysis of the survey results, has been the possible impact that the distribution of responses per level of experience may have on the calculation of each country average. Thus, a set of weights per level of experience has been defined, considering the number of responses obtained in the study. These weights, as presented in the table below, have also been applied to the results, producing the weighted averages.

	0-4 years	5-7 years	8-10 years	11-15 years	> 15 years	TOTAL
Number of responses	1.528	1.058	729	787	2.008	6.110
Weights	25,01%	17,32%	11,93%	12,88%	32,86%	

Table 6 – Weights applied per level of experience

In a similar way, weights per level of experience and gender have been calculated and applied. These weights are presented in the table below.

	0-4 years	5-7 years	8-10 years	11-15 years	> 15 years	TOTAL
Number of responses- Female	695	400	260	260	451	***2.066***
Weights	33,64%	19,36%	12,58%	12,58%	21,83%	
Number of responses- Male	859	683	483	537	1562	***4124***
Weights	20,83%	16,56%	11,71%	13,02%	37,88%	

Table 7 – Weights applied per level of experience and gender

Figure 2. Presented table detection framework consisting of Cascade Mask R-CNN, incorporating RFP and SAC in backbone network (ResNet-50). The modules RFP and SAC are illustrated in Figure 3 and 4, respectively.

where j iterates over 1, ..., S, i_0 represents the input image and f_{S+1} is set to 0. However, in the case of RFP, feedback connections are added to the conventional FPN as illustrated in Figure 3 with solid black arrows. If we include feature transformations T_j befoe joining the feedback connections from FPN to the bottom-up backbone, then the output feature f_j of RFP is explained in [24] as:

$$f_j = F_j(f_{j+1}, i_j), \;\; i_j = N_j(i_{j-1}, T_j(f_j)) \tag{2}$$

where j enumerates over S, the transformation of FPN to RFP makes it a recursive function. If we unfold the RFP to a sequence of T, mathematically, it is given by:

$$f_j^t = F_j^t(f_{j+1}^t, i_j^t), \;\; i_j^t = N_j^t(i_{j-1}^t, T_j^t(f_j^t)) \tag{3}$$

where t enumerates over U and U is the number of unfolded steps. The superscript t represents the function and the features at unfolded step t. We empirically set U = 2 in our experiments. For a comprehensive explanation of the RFP module, please refer to [24].

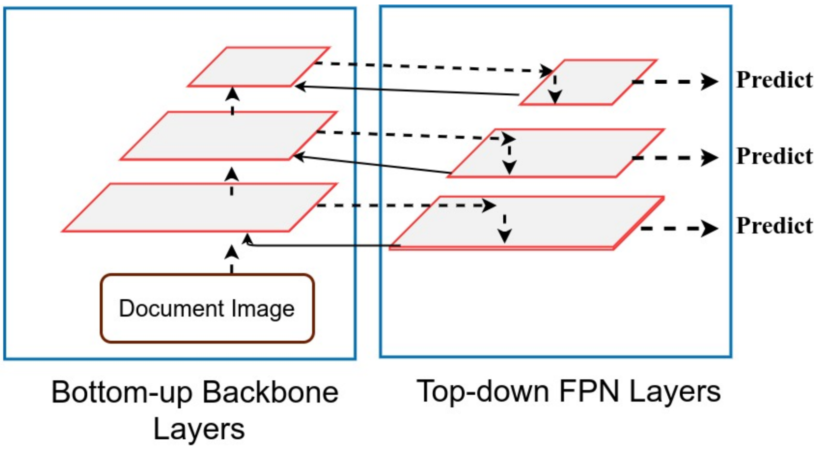


Figure 3. Illustrating design of Recursive Feature Pyramid module. The Recursive Feature Pyramid includes feedback connections that are highlighted with solid lines. The top-down FPN layers send the feedback to the bottom-up backbone layers by inspecting the image twice.

3.2. Switchable Atrous Convolution

We replace the conventional convolutions present in backbone network ResNet [62] and FPN with SAC. The artous convolution also referred to as dilated convolution [63] enables to increase size of effective receptive field by introducing an atrous rate. For an atrous rate of l in atrous convolution, it adds $l - 1$ zeros between the values of consecutive filter. Due to which the kernal with a size of $k \times k$ filter, enlarges to a size of $k + (k-1)(l-1)$ without causing any change in the number of network parameters. Figure 4 depicts an example of a 3 × 3 artous convolution with the artous rate of 1 (displayed in red), whereas an artous rate of 2 is demonstrated in green color.

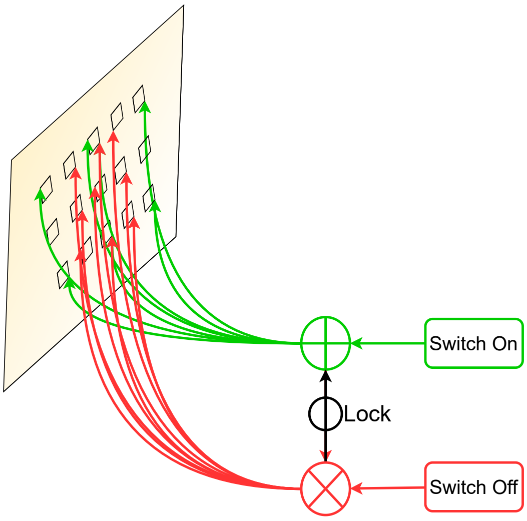


Figure 4. Illustrating Switchable Atrous Convolution. The red symbol $\otimes$ depicts artous convolutions with an artous rate set to 1, whereas the green symbol $\oplus$ denotes an artous rate of 2 in a 3 × 3 convolutional layer.

To transform a convolutional layer to SAC, we employ the basic artous convolutional operation Con that takes input i, weights w, and an artous rate l and outputs y. Mathematically, it is given by:

$$y = Con(i, w, l) \tag{4}$$

In case of SAC explained in [24], the above convolutional layer converts into:

$$Con(i, w, 1) \xrightarrow{SAC} S(i) \,.\, Con(i, w, l) + (1 - S(i)) \,.\, Con(i, w + \Delta w, l) \tag{5}$$

where $S(.)$ defines the switch function which is implemented is a combination of an average pooling and convolution layer with kernel of 5×5 and 1×1, respectively. The symbol Δw is trainable weight and l is a hyper-parameter. Owing to switch function, our backbone network adapts to arbitrary scales of tabular images, defying the need for deformable convolutions [53]. We empirically set the artous rate, l to 3 in our experiments. Moreover, we implement the idea of locking mechanism [24] by setting the weights to $w + \Delta w$ in order to exploit the backbone network pre-train on MS-COCO dataset [64]. Initially, $\Delta w = 0$ and w is set according to the pre-trained weights. We refer readers to [24] for the detailed explanation on SAC.

3.3. Cascade Mask R-CNN

To investigate the effectiveness of Recursive Feature Pyramid (RFP) and Switchable Atrous Convolution (SAC) modules on the task of table detection in scanned document images, we fuse these components into a cascade Mask R-CNN. The cascade Mask R-CNN is a direct combination of Mask R-CNN [55] and a recently proposed Cascade R-CNN [25]. The architecture of our utilized cascade Mask R-CNN closely follows the cascaded architecture introduced in [25] along with the addition of segmentation branch at the final network head [55]. The proposed CasTabDetectoRS consists of three detectors operating on rising IoU thresholds of 0.5, 0.6, and 0.7, respectively. The Region of Interest (ROI) pooling takes learned proposals from the Region proposal Network (RPN) and propagates the extracted ROI features to a series of network heads. The first network head receives the ROI features and performs classification and regression. The output of the first detector is treated as an input for the subsequent detector. Therefore, the predictions from the deeper network are refined and less prone to produce false positives. Furthermore, each regressor is enhanced with the localization distribution estimated by the previous regressor instead of the actual initial distribution. This enables the network head operating on a higher IoU threshold to predict optimally localized bounding boxes. In the final stage of cascaded networks, along with regression and classification, the network performs segmentation to advance the final predictions further.

As illustrated in Figure 2, the proposed CasTabDetectoRS employs ResNet-50 [62] as a backbone network. The lightweight ResNet-50 backbone equipped with SAC, generates feature maps from the input scanned document image. The extracted feature maps are passed to the RFP that optimally transforms the features by leveraging feedback connections. Subsequently, these optimized features are passed to the RPN and ROI Pooling to make the final tabular predictions.

4. Experimental Results

4.1. Datasets

4.1.1. ICDAR-17 POD

The competition about detecting graphical Page Object Detection (POD) [1] is organized at ICDAR in 2017, which yielded the ICDAR-2017 POD dataset. The dataset contains bounding box information for tables, formulas, and figures. From 2417 images present in the dataset, 1600 images are used to fine-tune our network, and 817 images are utilized as a test set. Since the previous methods [12,20,30] have reported results on varying IoU thresholds, we present our results with an IoU threshold value ranging from $0.5 - 0.9$ to draw a direct comparison with prior methods.

4.1.2. ICDAR-19

Another competition for Table Detection and Recognition (cTDaR) [65] is organized at ICDAR in 2019. For the task of table detection (TRACK A), two new datasets (historical and modern) are introduced in the competition. The historical dataset comprises handwritten accounting ledgers, train timetables, whereas the modern dataset consists of scientific papers, forms, and commercial documents. In order to have a direct comparison against prior state-of-the-art [11], we report results on the modern datasets with an IoU threshold ranging from 0.5-0.9.

4.1.3. TableBank

Currently, TableBank [66] is one of the enormous datasets publicly available for the task of table detection in document images. The dataset comprises 417K annotated document images that are obtained by crawling documents from the arXiv database. It is important to highlight that we take 1500 images from the splits of Word and Latex and 3000 samples from Word+Latex split. This enables our results to have a straightforward comparison with earlier state-of-the-art results [11].

4.1.4. UNLV

UNLV [67] dataset comprises scanned document images collected from commercial documents, research papers, and magazines. The dataset has around 10K images. However, only 427 images contain tables. Since prior state-of-the-art methods [20] have only used tabular images, we follow the identical split for direct comparison.

4.1.5. Marmot

Earlier, Marmot [68] was one of the most widely exploited datasets in the table community. This dataset is published by the Institute of Computer Science and Technology (PekingUniversity) by collecting samples from Chinese and English conference papers. The dataset consists of 2K images with an almost 1:1 ratio between positive to negative

doi:10.20944/preprints202109.0059.v1

samples. For direct comparison with previous work [20], we used the cleaned version of the dataset by [7] and did not incorporate any sample of the dataset in the training set.

4.2. Implementation Details

We implement CasTabDetectoRS in Pytorch by leveraging the MMdetection framework [69]. Our table detection method operates on ResNet-50 backbone network [62] pre-trained on ImageNet [70]. Furthermore, we transform all the 3 × 3 conventional convolutions present in the bottom-up backbone network to SAC. We closely follow the experimental configurations of Cascade Mask R-CNN [25] in order to execute the training process. All input documents images are resized with a maximum size of 1200 × 800 by preserving the actual aspect ratio. We train all the models for straight 14 epochs by initially setting the learning rate of 0.0025 with a learning rate decay of 0.1 after six epochs and ten epochs. We set the IoU threshold values to [0.5, 0.6, 0.7] for the respective three stages of R-CNN, respectively. We use a single anchor scale of 8, whereas the anchor ratios are set to [0.5, 1.0, 2.0]. We train all the models with a batch size of 1. We train all the models on NVIDIA GeForce RTX 1080 Ti GPU with 12 GB memory.

4.3. Evaluation Protocol

Analogous to the prior table detection method on scanned document images [7, 8,11,12,19,20,30–33], we assess the performance of our CasTabDetectoRS on precision, recall, and F1-score. We have reported the IoU threshold values along with the achieved results for direct comparison with the existing approaches.

4.3.1. Precision

The precision [71] computes the ratio of true positive samples over the total predicted samples. Mathematically, it is calculated as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{6}$$

4.3.2. Recall

The recall [71] is defined as the ratio of true positives over all all correct samples from the ground truth. It is calculated as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{7}$$

4.3.3. F1-Score

The f1-score [71] is defined as the harmonic mean of precision and recall. Mathematically, it is given by:

$$\text{F1-Score} = \frac{2\times\ \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8}$$

4.3.4. Intersection Over Union

Intersection over Union (IoU) [72] computes the intersecting region between the predicted and the ground truth region. The formula for the calculation of IoU is:

$$\text{IoU(A,B)} = \frac{\text{Area of Overlap region}}{\text{Area of Union region}} = \frac{|A \cap B|}{|A \cup B|} \tag{9}$$

4.4. Result and Discussion

To evaluate the performance of the proposed CasTabDetectoRS, we report the results on five different publicly available table detection datasets. This section presents

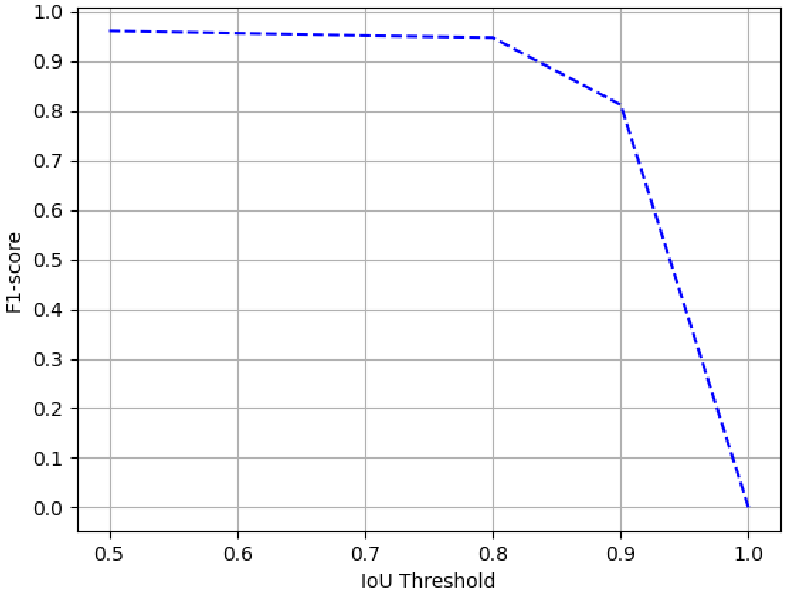


Figure 5. Performance evaluation of our CasTabDetectoRS in terms of f1-score over the varying IoU thresholds ranging from 0.5 to 1.0 on the ICDAR-2017-POD table detection dataset.

a comprehensive quantitative and qualitative analysis of our presented approach on all the datasets.

4.4.1. ICDAR-17 POD

The ICDAR-17 POD challenge dataset consists of 817 images with 317 tables in the test set. For direct comparison with previous entries in the competition [1] and previous state-of-the-art results, we report the results on the IoU threshold value of 0.6 and 0.8. Table 1 summarizes the results achieved by our model. On an IoU threshold value of 0.6, our CasTabDetectoRS achieves a precision of 0.941, recall of 0.972, and f1-score of 0.956. On increasing the IoU threshold from 0.6 to 0.8, the performance of our network only indicates a slight drop with a precision of 0.962, recall of 0.932, and f1-score of 0.947. Furthermore, Figure 5 illustrates the effect of various IoU thresholds on our table detection system. The qualitative performance of our proposed method on the ICDAR-17 POD dataset is highlighted in Figure 6. Analysis of incorrect results discloses that the network fails to localize precise tabular areas or produce false positives.

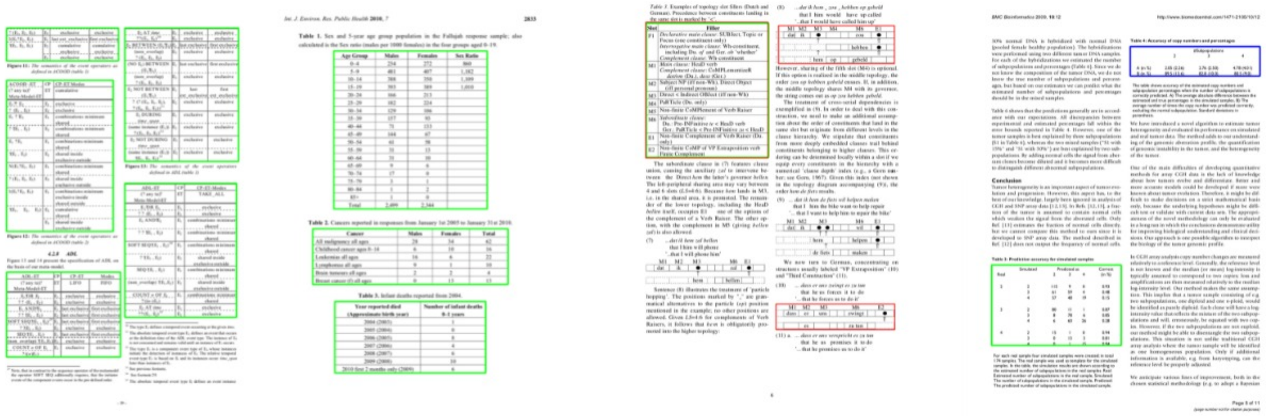
Comparison with State-of-the-art Approaches

By looking at Table 1, it is evident that our network achieves comparable results with the existing state-of-the-art approaches on the ICDAR-17 POD dataset. It is important to emphasize that methods introduced in [20] and [1] either rely on the heavy backbone with memory-intensive deformable convolutions [53] or are dependent on multiple pre and post-processing methods to achieve the results. On the contrary, our CasTabDetectoRS operates on a lighter weight ResNet-50 backbone with switchable atrous convolutions. Furthermore, it is vital to mention that the system [54] that produced state-of-the-art results on this dataset learns to classify tables, figures, and equations. However, the proposed system only trains on the limited tabular information and has no idea about other similar graphical page objects like figures and equations. Therefore, having low inter-class variance between the different graphical page objects and tables in this dataset, our network tends to produce more false positives and fail to surpass state-of-the-art results.

doi:10.20944/preprints202109.0059.v1

Table 1. Performance comparison between the proposed CasTabDetectoRS and previous state-of-the-art results on table detection dataset of ICDAR-17 POD.

Method	IoU = 0.6			IoU = 0.8		
	Recall	Precision	F1-score	Recall	Precision	F1-score
DeCNT [20]	0.971	0.965	0.968	0.952	0.946	0.949
NLPR-PAL [1]	0.953	0.968	0.960	**0.958**	0.943	0.951
VisInt [1]	0.918	0.924	0.921	0.823	0.829	0.826
GOD [54]	-	-	**0.989**	-	-	**0.971**
CDeC-Net [12]	0.931	**0.977**	0.954	0.924	**0.970**	0.947
HybridTabNet [14]	**0.997**	0.882	0.936	**0.994**	0.879	0.933
CasTabDetectoRS (Ours)	0.941	0.972	0.956	0.932	0.962	0.947



(a) True Positives

(b) True Positive and False Positives

(c) True Positive and a False Negative

Figure 6. CasTabDetectoRS results on the ICDAR-2017 POD table detection dataset. Green represents true positive, red denotes false positive, and blue colour highlights false negative. In this figure, part (a) represents a couple of samples containing true positives. Part (b) highlights true positive and false positives, and part (c) depicts a true positive and a false negative.

Table 2. Performance comparison between the proposed CasTabDetectoRS and previous state-of-the-art results on the dataset of ICDAR 19 Track A (Modern).

Method	IoU = 0.8			IoU = 0.9		
	Recall	Precision	F1-Score	Recall	Precision	F1-Score
TableRadar [65]	0.940	0.950	0.945	0.890	0.900	0.895
NLPR-PAL [65]	0.930	0.930	0.930	0.860	0.860	0.860
Lenovo Ocean [65]	0.860	0.880	0.870	0.810	0.820	0.815
CascadeTabNet [11]	-	-	0.925	-	-	0.901
CDeC-Net [12]	0.934	0.953	0.944	0.904	0.922	0.913
HybridTabNet [14]	0.933	0.920	0.928	0.905	0.895	0.902
CasTabDetectoRS (Ours)	**0.988**	**0.964**	**0.976**	**0.951**	**0.928**	**0.939**

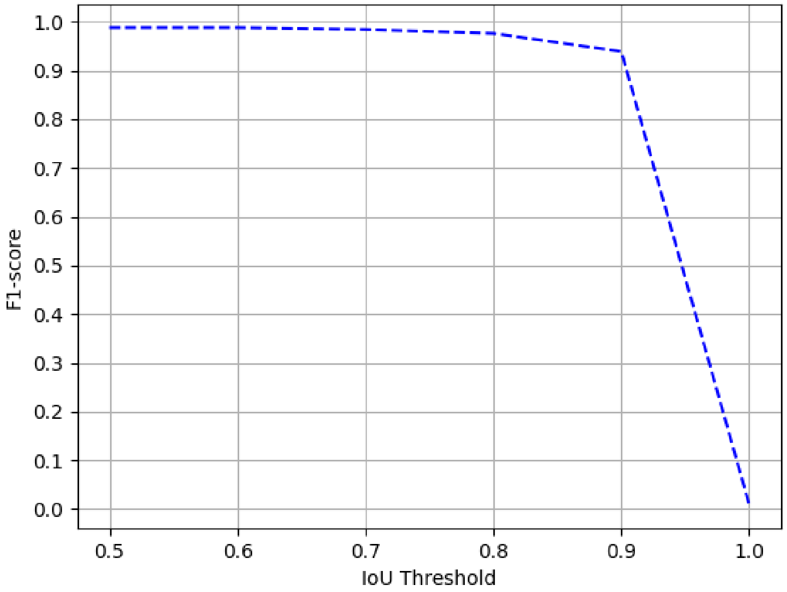


Figure 7. Performance evaluation of our CasTabDetectoRS in terms of f1-score over the varying IoU thresholds ranging from 0.5 to 1.0 on the ICDAR-2019 Track A (Modern) dataset.

4.4.2. ICDAR-19

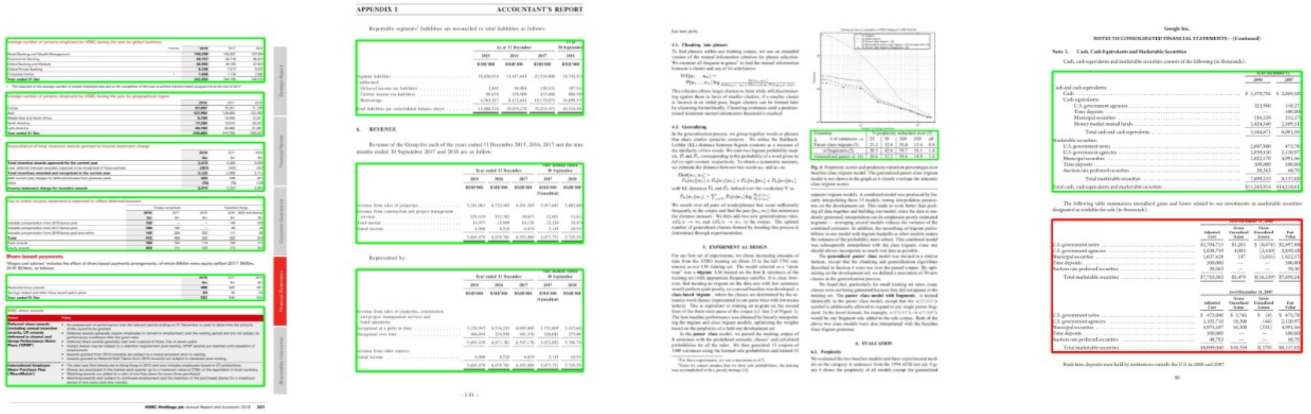
In this paper, the ICDAR-19 represents the Modern Track A part of the table detection dataset introduced in the table detection competition at ICDAR 2019 [65]. In order to draw strict comparisons with participants of the competition and existing state-of-the-art results, we evaluate the performance of our proposed method on the higher IoU threshold of 0.8 and 0.9. Table 2 presents the quantitative analysis of our proposed method, whereas the performance in terms of f1-score of our table detection method on various IoU thresholds are illustrated in Figure 7. The qualitative analysis is demonstrated in Figure 8. After analyzing false positives yielded by our network, we realize that the ground truth of the ICDAR-19 dataset has unlabelled tables present in the modern document images. One instance of such a scenario is exhibited in Figure 8(b).

Comparison with State-of-the-art Approaches

Along with presenting our achieved results on the ICDAR-19 dataset, Table 2 compares the performance of our CasTabDetectoRS with the prior state-of-the-art approaches. It is evident that our introduced cascade network equipped with RFP and SAC surpassed the previous state-of-the-art results with a significant margin. We accomplish a precision of 0.964, recall of 0.988, and an f1-score of 0.976 on an IoU threshold of 0.8. Upon increasing the IoU threshold to 0.9, the proposed table detection method achieves a precision of 0.928, recall of 0.951, and f1-score of 0.939. The higher difference between the f1-score of our method and the previously achieved f1-score clearly exhibits the superiority of our CasTabDetectoRS.

4.4.3. TableBank

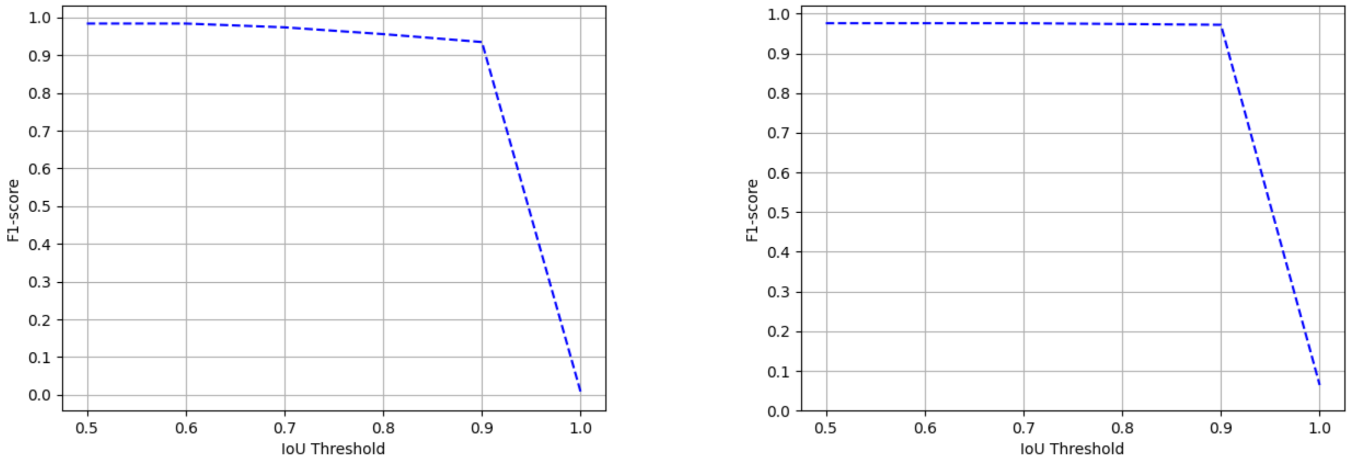
We evaluate the performance of the proposed method on all the three splits of TableBank dataset [66]. To establish a straightforward comparison with the recently achieved state-of-the-art results [11] on TableBank, we report the results on the IoU threshold of 0.5. Furthermore, owing to the superior predictions of our proposed method, we present results on a higher IoU threshold of 0.9. Table 3 summarizes the performance of our CasTabDetectoRS on the splits of TableBank-Latex, TableBank-Word, and TableBank-



(a) True Positive Samples

(b) True Positive and a False Positive

Figure 8. CasTabDetectoRS results on the table detection dataset of ICDAR-2019 Track A (Modern). Green represents true positive, whereas red denotes false positive. In this figure, part (a) highlights a couple of samples containing true positives, whereas part (b) represents a true positive and a false positive.



(**a**) TableBank-LaTeX.

(**b**) TableBank-Word.

Figure 9. Performance evaluation of our CasTabDetectoRS in terms of f1-score over the varying IoU thresholds ranging from 0.5 to 1.0 on TableBank-LaTeX and TableBank-Word datasets.

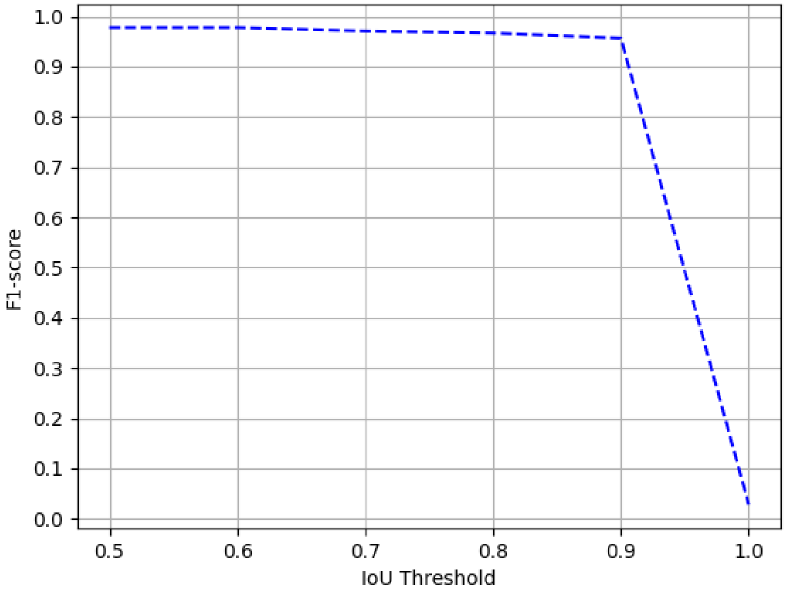


Figure 10. Performance evaluation of our CasTabDetectoRS in terms of f1-score over the varying IoU thresholds ranging from 0.5 to 1.0 on the TableBank-Both dataset.

Both. Along with the quantitative results, we demonstrate the performance of the proposed system in terms of f1-score by increasing the IoU thresholds from 0.5 to 1.0. Figure 9 depicts the drop in performance on the split of TableBank-Latex and TableBank-Word, whereas, Figure 10 explains the f1-score on the split of TableBank-Both. Figure 11 depicts a couple of true positives and one instance each of false positive and a false negative.

Comparison with State-of-the-art Approaches

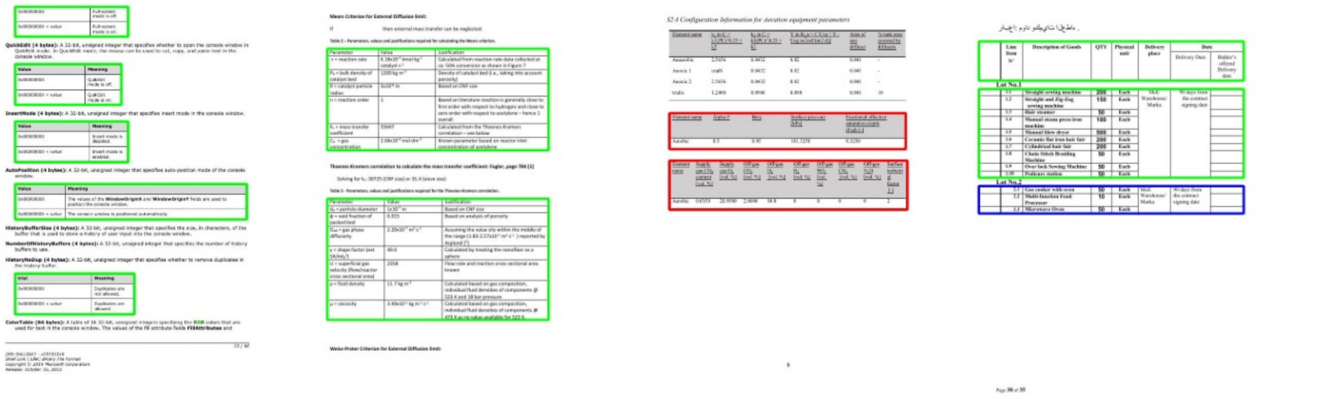
Table 3 provides the comparison between existing state-of-the-art table detection methods and our proposed approach. It is clear that our proposed CasTabDetectoRS has surpassed the previous baseline and state-of-the-art methods on all the three splits of the TableBank dataset. On the dataset split of TableBank-Latex, we achieve an f1-score of 0.984 and 0.935 with an IoU threshold of 0.5 and 0.9, respectively. Similarly, we accomplish f1-scores of 0.976 and 0.972 on the IoU threshold of 0.5 and 0.9, respectively, on the TableBank-Word dataset. Moreover, we attain f1-scores of 0.978 and 0.957 on IoU of 0.5 and 0.9, respectively on TableBank-(Word+LaTex) dataset.

4.4.4. Marmot

The Marmot dataset consists of 1967 document images comprising 1348 tables. Since prior state-of-the-art approaches [12,20] have employed the model trained on the ICDAR-17 dataset to evaluate the performance on the Marmot dataset, we have identically reported the results to have a direct comparison. Table 4 presents the quantitative analysis of our proposed method, whereas Figure 12 illustrates the effect of our CasTabDetectoRS on increasing the IoU threshold from 0.5 to 1.0. Figure 13 portrays the qualitative assessment of our table detection system on the Marmot dataset by illustrating samples of true positives, false positives, and a false negative.

Table 3. Performance comparison between the proposed CasTabDetectoRS and previous state-of-the-art results on various splits of TableBank dataset. The double horizontal lines divide the different splits.

Method	Dataset	IoU = 0.5			IoU = 0.9		
		Recall	Precision	F1-score	Recall	Precision	F1-score
CascadeTabNet [11]	TableBank-LaTeX	0.972	0.959	0.966	-	-	-
Li et al. [66]	TableBank-LaTeX	0.962	0.872	0.915	-	-	-
HybridTabNet [14]	TableBank-LaTeX	-	-	0.980	-	-	0.934
CasTabDetectoRS (Ours)	TableBank-LaTeX	**0.984**	**0.983**	**0.984**	**0.935**	**0.935**	**0.935**
CascadeTabNet [11]	TableBank-Word	0.955	0.943	0.949	-	-	-
Li et al. [66]	TableBank-Word	0.803	0.965	0.877	-	-	-
HybridTabNet [14]	TableBank-Word	-	-	0.970	-	-	0.962
CasTabDetectoRS (Ours)	TableBank-Word	**0.985**	**0.967**	**0.976**	**0.981**	**0.963**	**0.972**
CascadeTabNet [11]	TableBank-Both	0.957	0.944	0.943	-	-	-
Li et al. [66]	TableBank-Both	0.904	0.959	0.931	-	-	-
HybridTabNet [14]	TableBank-Both	-	-	0.975	-	-	0.949
CasTabDetectoRS (Ours)	TableBank-Both	**0.982**	**0.974**	**0.978**	**0.961**	**0.953**	**0.957**



(a) True Positives (b) False Positives (c) True Positives and a False Negative

Figure 11. CasTabDetectoRS results on the TableBank dataset. Green represents true positive, red denotes false positive, and blue colour highlights false negative. In this figure, part (a) represents a couple of samples containing true positives. Part (b) illustrates false positives, and part (c) depicts true positives and false negatives.

Table 4. Performance comparison between the proposed CasTabDetectoRS and previous state-of-the-art results on the Marmot dataset.

Method	IoU = 0.5			IoU = 0.9		
	Recall	Precision	F1-score	Recall	Precision	F1-score
DeCNT [20]	0.946	0.849	0.895	-	-	-
CDeC-Net [12]	0.930	0.975	0.952	0.765	0.774	0.769
HybridTabNet [14]	0.961	0.951	0.956	0.903	0.900	0.901
CasTabDetectoRS (Ours)	**0.965**	**0.952**	**0.958**	**0.901**	**0.906**	**0.904**

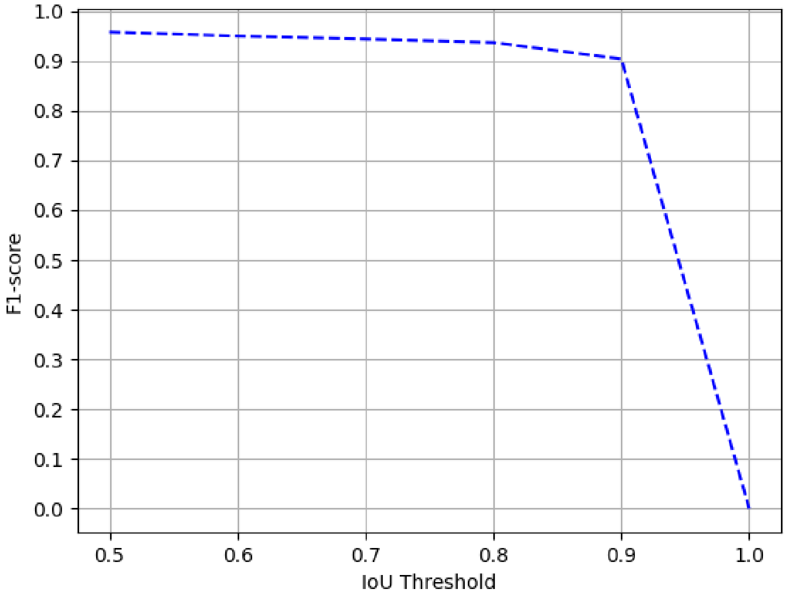


Figure 12. Performance evaluation of our CasTabDetectoRS in terms of f1-score over the varying IoU thresholds ranging from 0.5 to 1.0 on the Marmot dataset.

Comparison with State-of-the-art Approaches

Table 4 summarizes the performance comparison between the previous state-of-the-art results and the results achieved by our CasTabDetectoRS Marmot dataset. Our proposed method outperforms the previous results with an f1-score of 0.958 and 0904 on the IoU threshold values of 0.5 and 0.9, respectively.

4.4.5. UNLV

The UNLV dataset comprises 424 document images containing a total of 558 tables. We evaluate the performance of our presented method on the UNLV dataset to exhibit the completeness of our approach. Similarly, for direct comparison with prior works [12,19] on this dataset, we present our results on the IoU threshold of 0.5 and 0.6 as summarized in Table 5. Moreover, Figure 14 explains the deterioration in performance of the system on increasing the IoU threshold from 0.5 to 1.0. For the qualitative analysis on the UNLV dataset, examples of true positives, false positives and a false negative are illustrated in Figure 15.

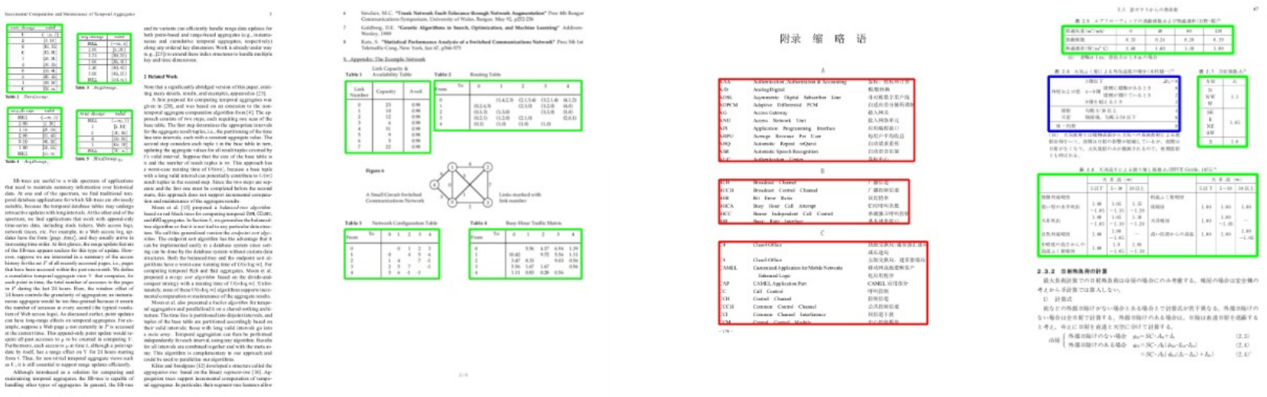
Comparison with State-of-the-art Approaches

The performance comparison between the proposed method and previous attempts on the UNLV dataset is summarized in Table 5. With the obtained results, it is apparent that our proposed system has outsmarted earlier methods with f1-scores of 0.946 and 0.933 on the IoU threshold values of 0.5 and 0.6, respectively.

4.4.6. Cross-Datasets Evaluation

Currently, the deep learning-based table detection methods are preferred over rule-based methods due to their better generalization capabilities over distinctive datasets. To investigate how well our proposed CasTabDetectoRS generalize over different datasets, we perform cross-dataset evaluation by incorporating four state-of-the-art table detection models inferred over five different datasets. We summarize all the results in Table 6.

With the table detection model trained on the TableBank-Latex dataset, apart from ICDAR-19, we achieve impressive results on ICDAR-17, TableBank-Word, Marmot, and UNLV with an average f1-score of 0.865. After manual inspection, we observe that the



(a) True Positives (b) False Positives (c) True Positives and a False Negative

Figure 13. CasTabDetectoRS results on the Marmot dataset. Green represents true positive, red denotes false positive, and blue colour highlights false negative. In this figure, part (a) exhibits a couple of samples containing true positives. Part (b) illustrates false positives, and part (c) depicts true positives and false negatives.

Table 5. Performance comparison between the proposed CasTabDetectoRS and previous state-of-the-art results on the UNLV dataset.

Method	IoU = 0.5			IoU = 0.6		
	Recall	**Precision**	**F1-score**	**Recall**	**Precision**	**F1-score**
Gilani et al. [19]	0.907	0.823	0.863	-	-	-
CDeC-Net [12]	0.906	0.914	0.910	0.805	0.961	0.883
HybridTabNet [14]	0.926	0.962	0.944	**0.914**	0.949	0.932
CasTabDetectoRS (Ours)	**0.928**	**0.964**	**0.946**	**0.914**	**0.952**	**0.933**

system produces several false positives due to the varying nature of document images in ICDAR-19 and TableBank-LaTeX. The table detection model trained on the ICDAR-17 dataset yields the average f1-score of 0.812 owing to the poor results achieved on the ICDAR-19 and UNLV datasets. The network trained on the ICDAR-19 dataset becomes the most generalized model accomplishing the average f1-score of 0.924. Although the size of the UNLV dataset is small (424 document images), the model trained on this dataset generates second-best results with an average f1-score of 0.897.

Manual investigation of cross-datasets evaluation yields the misinterpretation of other graphical page objects [2] with tables. However, with the obtained results, it is evident that our proposed CasTabDetectoRS produces state-of-the-art results on a specific dataset and generalizes well over the other datasets. Such types of well-generalized table detection systems for scanned document images are required in several domains. [8].

5. Conclusion and Future Work

This paper presents CasTabDetectoRS, the novel table detection framework for scanned document images, which comprises Cascade Mask R-CNN with a Recursive Feature Pyramid (RFP) network with Switchable Atrous Convolutions (SAC). The proposed CasTabDetectoRS accomplishes state-of-the-art performances on the four different table detection datasets (ICDAR-19 [65], TableBank [66], UNLV [67], and Marmot [68]) while achieving comparable results on the ICDAR-17-POD [1] dataset.

Table 6. Examining the generalization capabilities of the proposed CasTabDetectoRS through cross datasets evaluation.

Training Dataset	Testing Dataset	Recall	Precision	F1-Score	Average F1-Score
TableBank-Latex	ICDAR-19	0.605	0.778	0.680	0.865
	ICDAR-17	0.866	0.958	0.910	
	TableBank-Word	0.967	0.947	0.957	
	Marmot	0.893	0.963	0.927	
	UNLV	0.918	0.856	0.885	
ICDAR-17	ICDAR-19	0.649	0.778	0.686	0.812
	TableBank-Word	0.983	0.943	0.963	
	Marmot	0.965	0.952	0.958	
	UNLV	0.607	0.685	0.644	
ICDAR-19	ICDAR-17	0.894	0.917	0.906	0.924
	TableBank-Word	0.981	0.921	0.950	
	Marmot	0.925	0.956	0.940	
	UNLV	0.898	0.876	0.887	
UNLV	ICDAR-17	0.867	0.879	0.881	0.897
	TableBank-Word	0.903	0.941	0.922	
	Marmot	0.874	0.945	0.908	
	ICDAR-19	0.839	0.918	0.877	

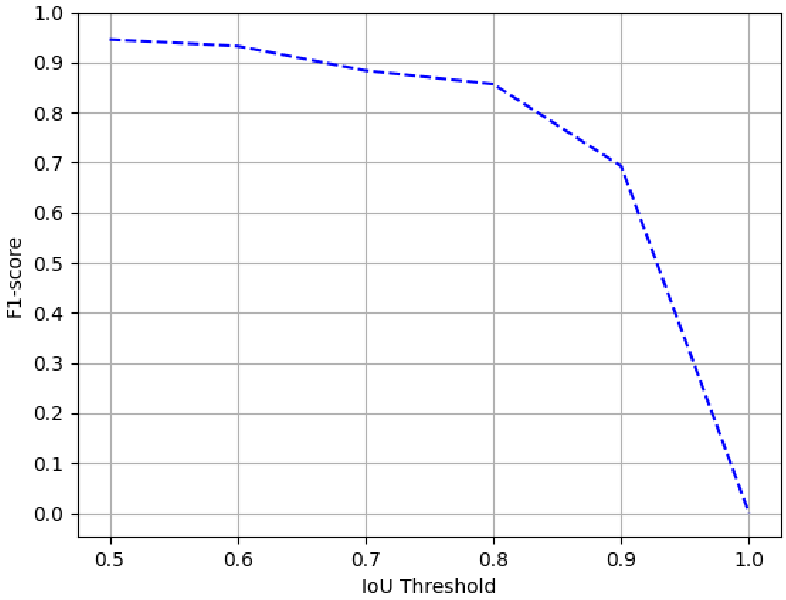
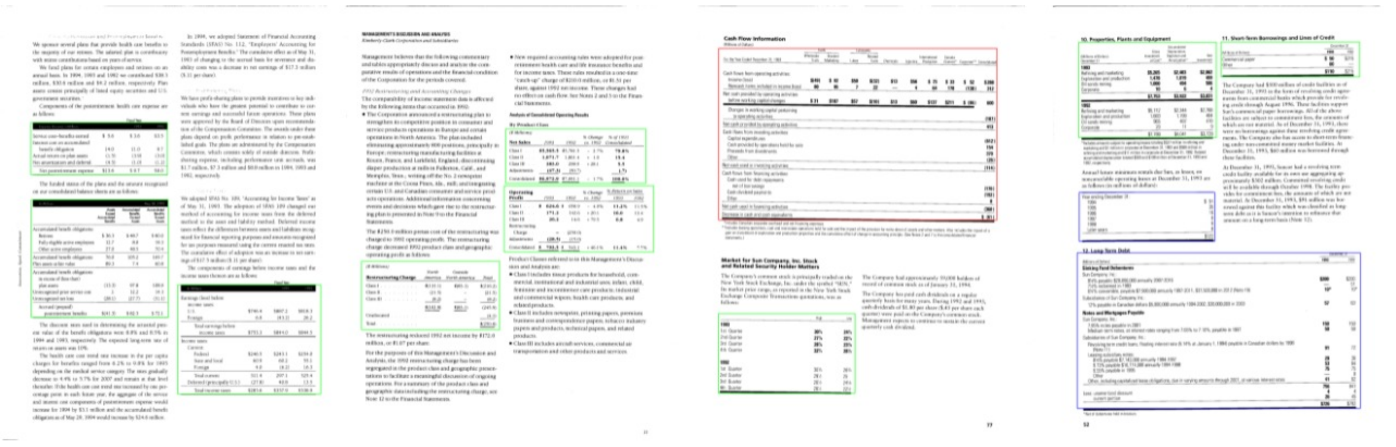


Figure 14. Performance evaluation of our CasTabDetectoRS in terms of f1-score over the varying IoU thresholds ranging from 0.5 to 1.0 on the UNLV dataset.

Upon direct comparison against previous state-of-the-art results on ICDAR-19 Track A (Modern) dataset, we reduce the relative error by 56.36% and 29.89% in terms



(a) True Positives (b) True Positive and a False Positive (c) True Positives and a False Negative

Figure 15. CasTabDetectoRS results on the UNLV dataset. Green represents true positive, red denotes false positive, and blue colour highlights false negative. In this figure, part (a) highlights a couple of samples containing true positives. Part (b) represents a true positive and a false positive, whereas part (c) depicts true positives and false negatives.

of achieved f1-score on IoU thresholds of 0.8 and 0.9, respectively. On the dataset of TableBank-Latex and TableBank-Word, we decrease the relative error by 20% on each dataset splits. On TableBank-Both, we reduce the relative error by 12%. Similarly, on the Marmot dataset [68], we observe a 4.55% reduction, whereas the system achieves a relative error reduction of 3.5% on the UNLV dataset [67]. Furthermore, this paper empirically establishes that instead of incorporating heavy backbone networks [11, 12] and memory exhaustive deformable convolutions [20], state-of-the-art results are achievable by employing a relatively lightweight backbone network (ResNet-50) with SAC. Moreover, this paper demonstrates the generalization capabilities of the proposed CasTabDetectoRS through extensive cross-datasets evaluations.

In the future work, we plan to extend the proposed framework by tackling the even more challenging task of table structure recognition in scanned document images. We expect that our cross-datasets evaluation sets a benchmark that will be followed in future examinations of table detection methods. Furthermore, the backbone network and the region proposal network of the proposed pipeline can be enhanced by exploiting the attention mechanism [73,74].

Author Contributions: writing—original draft preparation, K.A.H.; writing—review and editing, K.A.H., M.Z.A.; supervision and project administration, M.L., A.P., D.S. All authors have read and agreed to the submitted version of the manuscript.

Funding: The work leading to this publication has been partially funded by the European project INFINITY under Grant Agreement ID 883293.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gao, L.; Yi, X.; Jiang, Z.; Hao, L.; Tang, Z. ICDAR2017 competition on page object detection. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR), 2017, Vol. 1, pp. 1417–1422.
2. Bhatt, J.; Hashmi, K.A.; Afzal, M.Z.; Stricker, D. A Survey of Graphical Page Object Detection with Deep Neural Networks. *Applied Sciences* **2021**, *11*, 5344.

3. Zhao, Z.; Jiang, M.; Guo, S.; Wang, Z.; Chao, F.; Tan, K.C. Improving deep learning based optical character recognition via neural architecture search. IEEE Congr. Evol. Computation (CEC), 2020, pp. 1–7.
4. Hashmi, K.A.; Ponnappa, R.B.; Bukhari, S.S.; Jenckel, M.; Dengel, A. Feedback Learning: Automating the Process of Correcting and Completing the Extracted Information. Int. Conf. Document Anal. Recognit. Workshops (ICDARW), 2019, Vol. 5, pp. 116–121.
5. van Strien, D.; Beelen, K.; Ardanuy, M.C.; Hosseini, K.; McGillivray, B.; Colavizza, G. Assessing the Impact of OCR Quality on Downstream NLP Tasks. ICAART (1), 2020, pp. 484–496.
6. Kieninger, T.G. Table structure recognition based on robust block segmentation. Document Recognit. V, 1998, Vol. 3305, pp. 22–32.
7. Schreiber, S.; Agne, S.; Wolf, I.; Dengel, A.; Ahmed, S. Deepdesrt: Deep learning for detection and structure recognition of tables in document images. 14th IAPR Int. Conf. document Anal. Recognit. (ICDAR), 2017, Vol. 1, pp. 1162–1167.
8. Hashmi, K.A.; Liwicki, M.; Stricker, D.; Afzal, M.A.; Afzal, M.A.; Afzal, M.Z. Current Status and Performance Analysis of Table Recognition in Document Images with Deep Neural Networks. *IEEE Access* **2021**.
9. Hashmi, K.A.; Pagani, A.; Liwicki, M.; Stricker, D.; Afzal, M.Z. Cascade Network with Deformable Composite Backbone for Formula Detection in Scanned Document Images. *Applied Sciences* **2021**, *11*, 7610.
10. Smith, R. An overview of the Tesseract OCR engine. Ninth international conference on document analysis and recognition (ICDAR 2007). IEEE, 2007, Vol. 2, pp. 629–633.
11. Prasad, D.; Gadpal, A.; Kapadni, K.; Visave, M.; Sultanpure, K. CascadeTabNet: An approach for end to end table detection and structure recognition from image-based documents. Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit. Workshops, 2020, pp. 572–573.
12. Agarwal, M.; Mondal, A.; Jawahar, C. CDeC-Net: Composite Deformable Cascade Network for Table Detection in Document Images. *arXiv:2008.10831* **2020**.
13. Zheng, X.; Burdick, D.; Popa, L.; Zhong, X.; Wang, N.X.R. Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context. Proc. IEEE/CVF Winter Conf. Appl. Comput. Vision, 2021, pp. 697–706.
14. Afzal, M.Z.; Hashmi, K.; Liwicki, M.; Stricker, D.; Nazir, D.; Pagani, A. HybridTabNet: Towards Better Table Detection in Scanned Document Images. *Preprints:2021080360* **2021**.
15. Coüasnon, B.; Lemaitre, A. Handbook of Document Image Processing and Recognition, chapter Recognition of Tables and Forms. *D. Doermann and K. Tombre, Eds. London, U.K.: Springer* **2014**, pp. 647–677.
16. Zanibbi, R.; Blostein, D.; Cordy, J.R. A survey of table recognition. *Document Anal. Recognit.* **2004**, *7*, 1–16.
17. Kieninger, T.; Dengel, A. Applying the T-RECS table recognition system to the business letter domain. Proc. 6th Int. Conf. Document Anal. Recognit., 2001, pp. 518–522.
18. Shigarov, A.; Mikhailov, A.; Altaev, A. Configurable table structure recognition in untagged PDF documents. Proc. 2016 ACM Symp. Document Eng., 2016, pp. 119–122.
19. Gilani, A.; Qasim, S.R.; Malik, I.; Shafait, F. Table detection using deep learning. 14th IAPR Int. Conf. document Anal. Recognit. (ICDAR), 2017, Vol. 1, pp. 771–776.
20. Siddiqui, S.A.; Malik, M.I.; Agne, S.; Dengel, A.; Ahmed, S. Decnt: Deep deformable cnn for table detection. *IEEE Access* **2018**, *6*, 74151–74161.
21. Hashmi, K.A.; Stricker, D.; Liwicki, M.; Afzal, M.N.; Afzal, M.Z. Guided Table Structure Recognition through Anchor Optimization. *arXiv:2104.10538* **2021**.
22. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated Residual Transformations for Deep Neural Networks. *arXiv preprint arXiv:1611.05431* **2016**.
23. Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; others. Deep high-resolution representation learning for visual recognition. *IEEE Trans. pattern Anal. Mach. Intell.* **2020**.
24. Qiao, S.; Chen, L.C.; Yuille, A. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv preprint arXiv:2006.02334* **2020**.
25. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. Proc. IEEE Conf. Comput. vision pattern Recognit., 2018, pp. 6154–6162.
26. Itonori, K. Table structure recognition based on textblock arrangement and ruled line position. Proc. 2nd Int. Conf. Document Anal. Recognit. (ICDAR'93), 1993, pp. 765–768.
27. Chandran, S.; Kasturi, R. Structural recognition of tabulated data. Proc. 2nd Int. Conf. Document Anal. Recognit. (ICDAR'93), 1993, pp. 516–519.
28. Hirayama, Y. A method for table structure analysis using DP matching. Proc. 3rd Int. Conf. Document Anal. Recognit., 1995, Vol. 2, pp. 583–586.
29. Green, E.; Krishnamoorthy, M. Recognition of tables using table grammars. Proc. 4th Annu. Symp. Document Anal. Inf. Retrieval, 1995, pp. 261–278.
30. Huang, Y.; Yan, Q.; Li, Y.; Chen, Y.; Wang, X.; Gao, L.; Tang, Z. A YOLO-based table detection method. Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 813–818.
31. Casado-García, Á.; Domínguez, C.; Heras, J.; Mata, E.; Pascual, V. The benefits of close-domain fine-tuning for table detection in document images. Int. Workshop Document Anal. Sys. Springer, Cham, 2020, pp. 199–215.

32. Arif, S.; Shafait, F. Table detection in document images using foreground and background features. Digit. Image Computing : Techn. Appl. (DICTA), 2018, pp. 1–8.
33. Sun, N.; Zhu, Y.; Hu, X. Faster R-CNN based table detection combining corner locating. Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 1314–1319.
34. Qasim, S.R.; Mahmood, H.; Shafait, F. Rethinking table recognition using graph neural networks. Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 142–147.
35. Pyreddy, P.; Croft, W.B. Tintin: A system for retrieval in text tables. Proc. 2nd ACM Int. Conf. Digit. libraries, 1997, pp. 193–200.
36. Pivk, A.; Cimiano, P.; Sure, Y.; Gams, M.; Rajkovič, V.; Studer, R. Transforming arbitrary tables into logical form with TARTAR. *Data & Knowl. Eng.* **2007**, *60*, 567–595.
37. Hu, J.; Kashi, R.S.; Lopresti, D.P.; Wilfong, G. Medium-independent table detection. Document Recognit. Retrieval VII. International Society for Optics and Photonics, 1999, Vol. 3967, pp. 291–302.
38. e Silva, A.C.; Jorge, A.M.; Torgo, L. Design of an end-to-end method to extract information from tables. *Int. J. Document Anal. Recognit. (IJDAR)* **2006**, *8*, 144–171.
39. Khusro, S.; Latif, A.; Ullah, I. On methods and tools of table detection, extraction and annotation in PDF documents. *J. Inf. Sci.* **2015**, *41*, 41–57.
40. Embley, D.W.; Hurst, M.; Lopresti, D.; Nagy, G. Table-processing paradigms: a research survey. *Int. J. Document Anal. Recognit. (IJDAR)* **2006**, *8*, 66–86.
41. Kieninger, T.; Dengel, A. The t-recs table recognition and analysis system. Int. Workshop Document Anal. Sys. Springer, 1998, pp. 255–270.
42. Cesarini, F.; Marinai, S.; Sarti, L.; Soda, G. Trainable table location in document images. Object Recognit. supported user interaction service robots, 2002, Vol. 3, pp. 236–240.
43. Kasar, T.; Barlas, P.; Adam, S.; Chatelain, C.; Paquet, T. Learning to detect tables in scanned document images using line information. 12th Int. Conf. Document Anal. Recognit., 2013, pp. 1185–1189.
44. e Silva, A.C. Learning rich hidden markov models in document analysis: Table location. 10th Int. Conf. Document Anal. Recognit., 2009, pp. 843–847.
45. Silva, A. Parts that add up to a whole: a framework for the analysis of tables. *Edinburgh Univ. UK* **2010**.
46. Hao, L.; Gao, L.; Yi, X.; Tang, Z. A table detection method for pdf documents based on convolutional neural networks. 12th IAPR Workshop Document Anal. Sys. (DAS), 2016, pp. 287–292.
47. Kavasidis, I.; Palazzo, S.; Spampinato, C.; Pino, C.; Giordano, D.; Giuffrida, D.; Messina, P. A saliency-based convolutional neural network for table and chart detection in digitized documents. *arXiv:1804.06236* **2018**.
48. Paliwal, S.S.; Vishwanath, D.; Rahul, R.; Sharma, M.; Vig, L. Tablenet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images. Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 128–133.
49. Holeček, M.; Hoskovec, A.; Baudiš, P.; Klinger, P. Table understanding in structured documents. Int. Conf. Document Anal. Recognit. Workshops (ICDARW), 2019, Vol. 5, pp. 158–164.
50. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv:1506.01497* **2015**.
51. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. Eur. Conf. Comput. vision. Springer, Cham, 2014, pp. 818–833.
52. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556* **2014**.
53. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. Proc. IEEE Int. Conf. Comput. vision, 2017, pp. 764–773.
54. Saha, R.; Mondal, A.; Jawahar, C. Graphical object detection in document images. Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 51–58.
55. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. Proc. IEEE Int. Conf. Comput. vision, 2017, pp. 2961–2969.
56. Zhong, X.; ShafieiBavani, E.; Yepes, A.J. Image-based table recognition: data, model, and evaluation. *arXiv:1911.10683* **2019**.
57. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv:1804.02767* **2018**.
58. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. Eur. Conf. Comput. vision. Springer, Cham, 2016, pp. 21–37.
59. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. Proc. IEEE Int. Conf. Comput. vision, 2017, pp. 2980–2988.
60. Chen, K.; Pang, J.; Wang, J.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Shi, J.; Ouyang, W.; others. Hybrid task cascade for instance segmentation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4974–4983.
61. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.
62. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
63. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence* **2017**, *40*, 834–848.

doi:10.20944/preprints202109.0059.v1

64. Lin, T.Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L.; Dollar, P. Microsoft COCO: common objects in context (2014). *arXiv preprint arXiv:1405.0312* **2019**.
65. Gao, L.; Huang, Y.; Déjean, H.; Meunier, J.L.; Yan, Q.; Fang, Y.; Kleber, F.; Lang, E. ICDAR 2019 competition on table detection and recognition (cTDaR). Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 1510–1515.
66. Li, M.; Cui, L.; Huang, S.; Wei, F.; Zhou, M.; Li, Z. Tablebank: Table benchmark for image-based table detection and recognition. Proc. The 12th Lang. Resour. Eval. Conf., 2020, pp. 1918–1925.
67. Shahab, A.; Shafait, F.; Kieninger, T.; Dengel, A. An open approach towards the benchmarking of table structure recognition systems. Proc. 9th IAPR Int. Workshop Document Anal. Sys., 2010, pp. 113–120.
68. Fang, J.; Tao, X.; Tang, Z.; Qiu, R.; Liu, Y. Dataset, ground-truth and performance metrics for table detection evaluation. 10th IAPR Int. Workshop Document Anal. Sys., 2012, pp. 445–449.
69. Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; Zhang, Z.; Cheng, D.; Zhu, C.; Cheng, T.; Zhao, Q.; Li, B.; Lu, X.; Zhu, R.; Wu, Y.; Dai, J.; Wang, J.; Shi, J.; Ouyang, W.; Loy, C.C.; Lin, D. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv preprint arXiv:1906.07155* **2019**.
70. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Advances neural Inf. Process. Sys.* **2012**, *25*, 1097–1105.
71. Powers, D.M. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv:2010.16061* **2020**.
72. Blaschko, M.B.; Lampert, C.H. Learning to localize objects with structured output regression. Eur. Conf. Comput. vision. Springer, Cham, 2008, pp. 2–15.
73. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159* **2020**.
74. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030* **2021**.