A GAN-based Feature Generator for Table Detection

Yibo Li*, Qinqin Yan†, Yilun Huang*, Liangcai Gao*,
Zhi Tang*
*Institute of Computer Science & Technology, Peking University, Beijing, China
†State Key Laboratory of Digital Publishing Technology, Founder Group Co., LTD, Beijing, China

***Abstract*—Table detection is of great significance for the documents analysis and recognition. Although many methods have been proposed and great progress have been made, it is still a great challenge to recognize the less-ruled tables due to the lack of table line features. In this paper, we propose a novel network to generate the layout features for table text to improve the performance of less-ruled table recognition. This feature generator model is similar to the Generative Adversarial Networks (GAN). We force the feature generator model to extract similar features for both ruling tables and less-ruled tables. It can be added into some common object detection and semantic segmentation models such as Mask R-CNN, U-Net. Extensive experiments are conducted on the dataset of ICDAR2017 Page Object Detection Competition dataset and a closed dataset full of the less-ruled tables and non-ruled tables. The primary experimental results show that the proposed GAN-based feature generator is very helpful for less-ruled table detection.**

***Index Terms*—Table detection; Feature generator; Semantic segmentation; Object detection; Generative adversarial network**

I. Introduction

As an important element in documents, tables can express more information in fewer words. Recognizing table in digital documents is as old as the analysis of structured documents itself. Although a large number of methods have been already proposed to solve it, this task still proves to be difficult.

In the beginning, researchers tended to use layout analysis methods to work out table detection problems. They analyzed the position of texts, lines, and whitespaces and some heuristic methods were proposed. In these methods, many rules were defined to detect tables in documents. Later, machine learning methods became popular. Researchers extracted representative features including the position of texts, ruling lines, even the textual information such as the position of word “table” to train classifiers to divide text segmentation into table or non-table.

In recent years, with the development of deep learning, a large number of useful networks were proposed. Two fields of them, semantic segmentation and object detection, are similar to the table detection problems. Some networks, such as U-net [1], Faster R-CNN [2], Deeplab [3], Mask R-CNN [4], can be used to solve table detection problems after some adjustments. Therefore, there are lots of deep learning-based methods being proposed. The semantic segmentation methods try to classify each pixel in page images belonging to a table or not, and use some rule-based strategies to get the accurate location of the table. Object detection methods try to detect tables in documents with accurate coordinates as results. Both methods present good performance on table detection.

However, due to the diversity and complexity of tables, both methods make mistakes when detecting some tables. After analyzing the mistakes that both methods have made, we find that the ruling line feature plays an important role on table detecting. When detecting less-ruled tables or non-ruled tables, these models prone to error. And when there are some ruling lines in a figure, these models are easy to mistake a figure for a table, which means that layout information is much more vital.

In this paper, we propose a novel feature generator, which is based on Generative Adversarial Networks (GAN) [5]. We force the feature generator model to extract similar features for both ruling tables and less-ruled tables. We erase the ruling lines of tables in images and substitude these images for the noise in GAN. The generator tries to generate the layout feature from the images and the discriminator tries to discriminate whether the feature is generated from the real images or not. By training this network, the generator will get a similar feature map from the real images and the images without ruling lines of tables. We add this feature generator into U-net [1] and Mask R-CNN [4], which are commonly used in semantic segmentation and object detection. We conduct experiments on the dataset of ICDAR2017 Page Object Detection Competition [6] and a closed dataset full of documents which have less-ruled tables. The feature generator shows significant improvement on both models. And the whole model present excellent performance on datasets.

The rest of the paper is organized as follows. In Section II, we summarize the previous work on table detection and table recognition. In Section III, we introduce the details of the feature generator and how to add this model into U-net and Mask R-CNN. In Section IV, we introduce the dataset we use and present the experimental results of both networks. Section V presents the conclusion of this paper.

II. Related Works

A large number of methods on table detection and recognition have been proposed. In the earlier years, researchers tried to solve these problems by rule-based methods.

A pioneering method is proposed by Kieninger et al. [7]. They develop table spotting and structure exaction system called T-Recs. The system regards word bounding boxes as

2379-2140/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDAR.2019.00127

inputs. These word boxes are clustered with a bottom-up approach into regions by building a segmentation graph. The document regions are then regarded as candidate table regions if they satisfy several certain criterions. Pdf2table system, proposed by Yildiz et al. [8], is the first relative research carried out on PDF documents. The table regions are spotted by detecting and merging multi-line with more than one text segments. However, their methods cannot handle documents having multi-column layouts well.

Fang et al. [9] propose an effective table detection method via visual separators and geometric content layout information, targeting at PDF documents. Firstly, they make the page layout analysis inspired by whitespace analysis algorithm proposed by Breuel [10] to determine page columns. Then they make use of graphics ruling lines parsed from PDF documents, some of which cannot be visible by human's eyes in real documents actually. It improves the detecting performance, especially for tables without ruling lines. Tupaj et al. [11] apply OCR technique to table detection method. The system searches for the lines that might be the table caption based on keywords, such as ”table”. The lines will be regarded as table caption, and the lines around them were then analyzed whether they belong to a table structure. However, a large number of tables in the documents don't have a table name. This approach will miss these tables when working.

At the same time, researchers also try some traditional machine learning approaches. Tabfinder system, proposed by Cesarini et al. [12], is one of the first techniques that applies machine learning method to table detection task. In this system, the documents are described by means of a hierarchical representation that is based on the MXY tree. And a table block can be hypothesized by searching parallel lines in the MXY tree and be verified by locating perpendicular lines or white spaces in the region between the parallel lines. Silva et al. [13] propose a technique for table detection task using Hidden Markov Models (HMMs). The system uses pdftotext Linux utility to extract text from PDF and computes feature vectors based on spaces between texts. Kesar et al. [14] present a method to locate tables by a set of features coming from the horizontal and vertical lines. These features will be passed to Support Vector Machine (SVM) to detect tables.

Then, with the development of deep learning on semantic segmentation and object detection, some deep-learning approaches were proposed. Semantic segmentation methods consider this problem as a pixel-wise classification task. He et al. [15] propose a multi-scale, multi-task fully convolutional neural network (FCN) for this task and use conditional random field (CRF) to make the network's output more accurate. Kavasidis et al. [16] proposed a saliency-based convolutional neural network for table and chart detection. In order to capture global information from the documents, they applied dilated convolutions to their network. The object detection methods can output accurate ordinates directly. Schreiber et al. [17] and Gilani et al. [18] both choose Faster R-CNN [2] as the basic framework, which still yields state-of-the-art performance in many works. In the lastest study on table detection, Siddiqui et al. [19] proposed DeCNT combining deformable CNN and Faster R-CNN/FPN. This model can adapt its receptive field according to its input. Li et al. [20] used rule-based method to get candidate regions, then classified them into tables, formulas and other labels by CRF and CNN model. Both methods achieved excellent performance on ICDAR2017 POD Competition dataset [6].

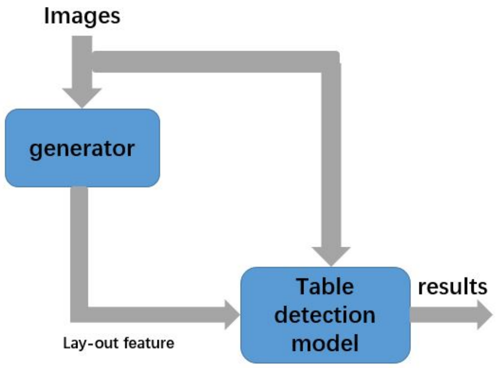


Fig. 1: Detail of the whole model

III. Methods

The whole model can be regarded as the combination of a feature generator and a general table detection network, which is shown in Figure 1. The inputs are documents images and it outputs the accurate coordinates of detected tables.

A. Feature Generative Adversarial Network

Just like GAN [5], the proposed network are composed of a feature generator and a discriminator. The goal of the generator is generating the layout feature from the real images and fake images. The goal of the discriminator is judging whether the features come from the real images or not. The based feature generative adversarial model is from DC-GAN [21].

The generator takes real images and fake images whose ruling lines of tables have been erased as input and outputs the layout features. The input's size is 512x512x3 and the output's size is 16x16x256. We use the network structure of VGG. It stacks 5 units, in which there are three 3x3 convolutions with a rectified linear unit (ReLU) as its activation, followed by a batch normalization and a 2x2 max-pooling operator with stride 2. After 5 units, we use a 3x3 convolution with Tanh activation to get the final feature map. There are 16 convolution layers, 5 max-pooling layers in this network and there are about 4 million parameters in total. The detail is shown in Figure 2

The discriminator takes the features which is produced by the generator as input and outputs a probability to judge whether the feature is from the real images. The detail of the discriminator network is shown in Figure 3. This network contains 3 units, in which there are three 3x3 convolutions with ReLU activation, followed by a batch normalization and

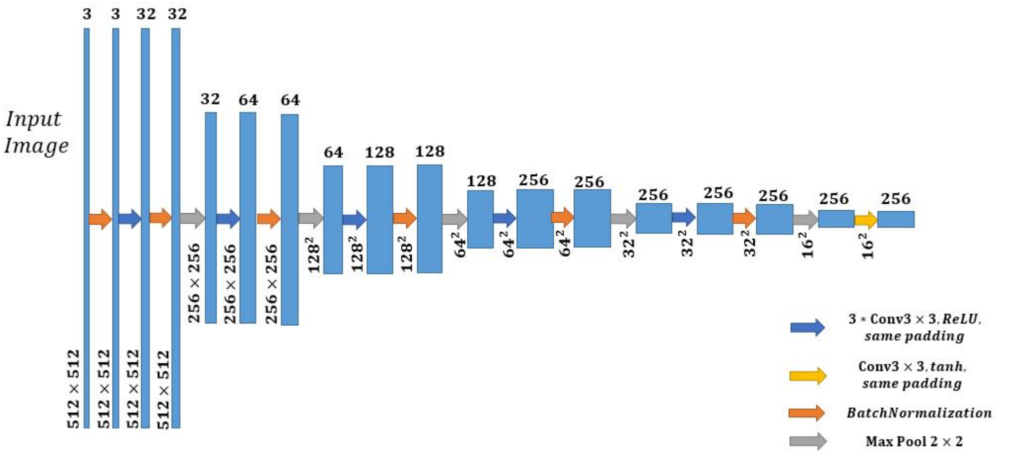


Fig. 2: Generator architecture. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. The arrows denote the different operations.

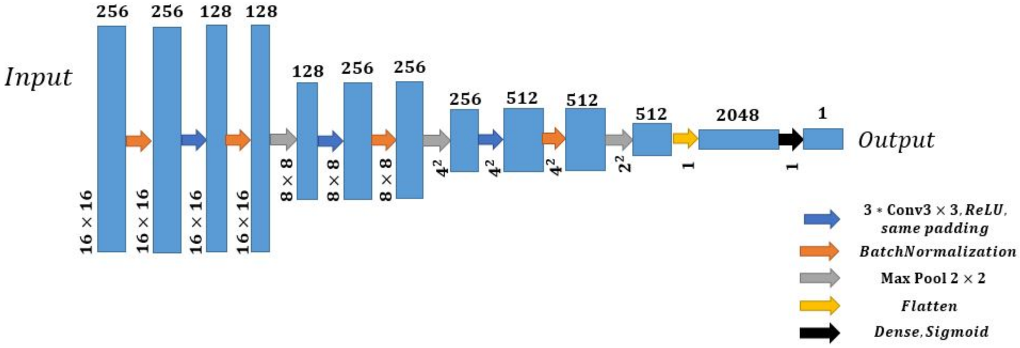


Fig. 3: Discriminator architecture. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. The arrows denote the different operations.

a 2x2 max pooling operator with stride 2. After 3 units, we use a flatten layer and a dense layer to get the final results. There are 9 convolution layers, 3 max-pooling layers, a flatten layer and a dense layer, about 8 million parameters in total. We can see the detail in Figure 3

When we begin to train the feature generative adversarial network. We can consider that there are two networks in this method. One is the discriminator model and another is a model merging the generator and discriminator which can be called "FGAN". Both models use binary cross-entropy as loss function. The detail of training process can be seen at Algorithm 1. In the discriminator model, the goal is to distinguish whether the feature is true or not, so the label of fake images is "false". In the "FGAN" model, the goal of the generator is generating the feature from fake images and making the discriminator consider them as "true" images. So the part of the discriminator is untrainable and the label of fake images is "true".

Algorithm 1 How to train feature generator

```
Input: Images from documents
1: Erase the ruling lines of table to get fake images
2: Get the feature map from images by using generator
3: Train the discriminator by using real feature maps and
   fake feature maps
4: Train the FGAN model by using fake and real images.
5: Goto step 2
```

B. Fature Generator+U-Net

The U-net network is representative in semantic segmentation field. It comes from the "fully convolution network" [22]. The main idea in "FCN" is to supplement a usual contracting network by successive layers, where pooling operators are replaced by upsampling operators. Hence, these layers increase the resolution of the output. In order to localize, high-resolution features from the contracting path are combined with the upsampled output. A successive convolution layer

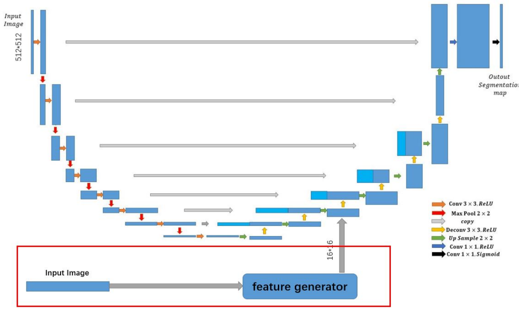


Fig. 4: U-Net architecture with feature generator. The generator in the red rectangle and the rest is U-net model.

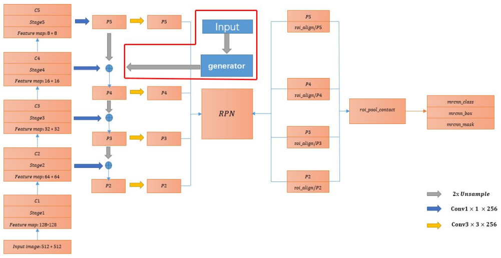


Fig. 5: Mask R-CNN architecture with feature generator. The generator in the red rectangle and the rest is MaskR-CNN model.

can then learn to assemble a more precise output based on this information. It is useful for table detection.

We add the feature generator into the U-net network [1]. The layout features will be added at the upsampling layers. Meanwhile, the generator is untrainable. We can consider it as a function to produce feature from the images. The detail of this network can be seen in Figure 4.

However, In the U-net network, a class label is supposed to be assigned to each pixel. What we want is the accurate coordinates of the table. Hence, we proposed an algorithm to get the coordinates. Each pixel will be classified to table or non-table, and we choose a table-pixel and diffuse until the pixel is non-table. We will ignore the predicted table if its size is smaller than 10x10 pixels.

C. Feature Generator+Mask R-CNN

Mask R-CNN, as a representative network in the object detection field, extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. Mask R-CNN is easy to train and adds only a small overhead to Faster R-CNN, running at 5 fps. It is easy to generalize to other tasks. So it is applied to solve table detection problem here.

Similar to feature+U-net network, we add the feature to the upsampling layer and the feature generator network is untrainable when we train the Mask R-CNN model. Figure 5 shows the detail.

D. Data Preprocesing

In section III-A, we proposed the Feature Generative Adversarial Network. The first step of training process is removing the ruling lines of tables. Hence, we need to solve the problem of erasing the ruling lines. Firstly, we resize every image to the same size of 512x512. And we convert them into gray images. Then for each bounding box of ground truth for images, we compute the average pixel value of each pixel line in the horizontal direction. If it is smaller than a fixed value, we will set each pixel of this line to white. We also repeat this process in the vertical direction.

IV. Experiments

A. Dataset

We train our model on the training set of ICDAR 2017 Page Object Detection (POD) Competition [6]. In this competition, competitors need to detect three kinds of structure from the documents: formulas, tables, and figures. We only choose the images which have tables to conduct experiments. Hence, after screening, there are 549 images with 699 tables in the training set and 243 images with 317 tables in the test set. To test our model's performance on less-ruled tables, we collected and labeled 543 images, which have less-ruled tables. Although the size of training set is not large, it shows good variety in different kinds of tables. Just like the common documents, there are ruled table, less-ruled table and non-ruled table and there are one or more tables in one image.

B. Performance Measurement

We use the classic Precision, Recall and F1 metric with two Intersection Over Union (IOU) under the thresholds of 0.6 and 0.8, which is one of the standard metrics used in ICDAR 2017 Competition. IOU is computed as:

$$IoU_i = \frac{P_i \cap G_i}{P_i \cup G_i} \tag{1}$$

where S_i denotes the region i predicted by our model and Y_i denotes the corresponding region of ground truth. If the IoU value of region i is larger than threshold, such as 0.6 or 0.8, this region would be considered as true positive (TP). Otherwise, this region is a false positive (FP). Those table regions that are not detected by our model are false negative (FN). Hence, precision, recall and F1 are computed as follows:

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \tag{4}$$

C. Experiment Results

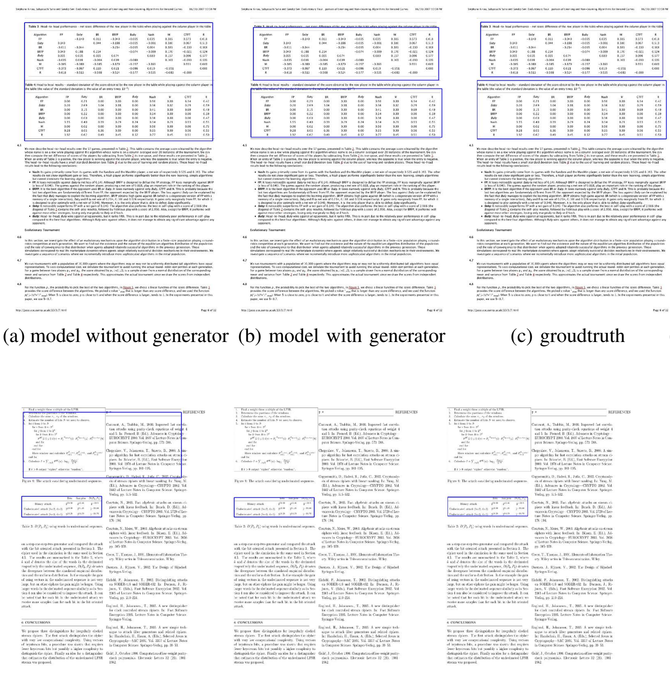
To verify the effectiveness of the feature generative adversarial network on semantic segment and object detection, we totally conduct 4 experiments, U-net, Mask R-CNN, feature+U-net and feature+Mask R-CNN on two datasets. We train the models by using the ICDAR 2017 training set. We

TABLE I: Experiment results on ICDAR 2017 POD Competition test dataset. 'F' means the feature generator.

Model	IoU = 0.6			IoU = 0.8		
	Precision	Recall	F1-measure	Precision	Recall	F1-measure
U-NET	0.880	0.897	0.888	0.762	0.793	0.777
F+U-NET	**0.891**	**0.915**	**0.903**	**0.785**	**0.802**	**0.793**
MASK R-CNN	0.936	0.925	0.930	0.893	**0.908**	0.900
F+MASK R-CNN	**0.944**	**0.944**	**0.944**	**0.903**	0.903	**0.903**

TABLE II: Experiments results on less-ruled table dataset. 'F' means feature generator.

Model	IoU = 0.6			IoU = 0.8		
	Precision	Recall	F1-measure	Precision	Recall	F1-measure
U-NET	0.922	0.958	0.940	0.796	0.828	0.812
F+U-NET	**0.926**	**0.959**	**0.942**	**0.841**	**0.871**	**0.857**
MASK R-CNN	0.958	**0.952**	0.955	0.934	**0.924**	0.928
F+MASK R-CNN	**0.967**	0.951	**0.959**	**0.945**	0.918	**0.931**



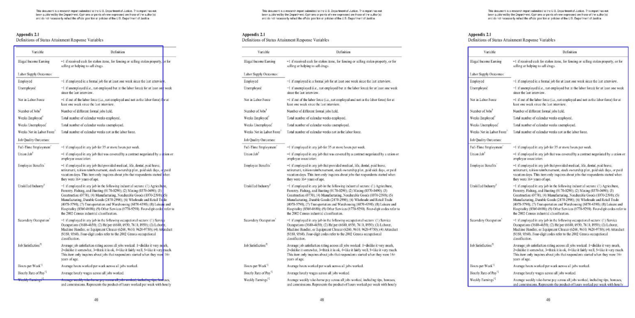
(a) model without generator (b) model with generator (c) groudtruth

(d) model without generator (e) model with generator (f) groudtruth

Fig. 6: Two samples of the results.

set the IOU threshold to 0.6 and 0.8. And test the model on the ICDAR 2017 test set. The results are shown in Table I. In order to prove that our models can make a better performance on the less-ruled tables, we test the models on closed dataset consisted of 543 images full of less-ruled tables, which are trained on the training set of ICDAR2017 POD Competition. The results are showed in Table II.

In general, our method improves the results of table detection. After adding the feature generator, the performance of U-Net and Mask R-CNN improves, especially the U-net model when IOU is 0.8. However, our best network Feature+MASK R-CNN is not the best when comparing with the latest results on ICDAR2017 POD Competition dataset [6]. For example, the approaches of Li et al. [20] and Siddiqui et al. [19] can achieve the F1-score of 0.968 when IOU is 0.6. But the F1-score of Feature+Mask RCNN is only 0.944. It's not hard to explain: with the feature generator added, the tables which has a similar layout feature to the text paragraphs may be missed, and the text paragraphs which are similar to tables will be mistaken for tables. A typical sample that a table is missed is shown in Figure 7. Although our approach is not the best on ICDAR2017 POD Competition, it achieves a state-of-the-art performance on closed less-ruled table dataset.



(a) model without generator (b) model with generator (c) groudtruth

Fig. 7: A typical sample of the results.

D. Error Analysis

The feature generator can enhance the layout information and help to detect less-ruled tables. Two samples of the results are shown in Figure 6. The six images from left to right are results of the original model, results of the model with feature generator, and the groundtruth. After observing the first line, we can find that the original model mistakes two tables as one table. Meanwhile, the model with the feature generator predicts the right result. As we can see, the layout of table headings and the layout of the table contents are different. The original model might not consider the difference so it makes a mistake. The model with the feature generator gets the layout information and makes the right judgment. On the second line, we can see that the original model predicts an algorithm with ruling lines and a part of texts which have a header line as a table. The layout of this area is obviously different from the table. Hence, the model with the feature generator doesn't make this mistake.

Although the feature generator can enhance the layout features and avoid mistakes, it still makes some problems in some situations. A typical sample can be seen in Figure 7. In this example, most cells of the table contain a large number of texts and the texts' lengths are different. The layout information is more similar to paragraphs than table. Therefore, the model with the feature generator cannot detect this table successfully.

V. Conclusion

In this paper, we propose a GAN-based feature generator for deep-learning methods of Table detection. The goal of this generator is to generate the layout features from the images and improve the performance of deep-learning methods when detecting less-ruled tables. We introduce how to build and train this generator and apply it to two common object detection and semantic segmentation networks. We test the original and new models on the ICDAR2017 POD dataset and a closed less-ruled table dataset. The result shows that our generator can improve performance and can enhance layout information indeed.

Acknowledgement

This woek is supported by the projects of National Nature Science Foundation of China (No. 61573028 and 61472014), which is also a research achievement of Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratoory of Intelligent Press Media Technology).

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox "U-Net: Convolutional Networks for BiomedicalImage Segmentation" *arXiv preprint arXiv:1505.04597v1*, 2015.

[2] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster RCNN: Towards Real-Time Object Detection with Region Proposal Networks," *Microsoft Research, Tech. Rep.,* 2015. [Online]. Available: https://arxiv.org/abs/1506.01497

[3] Liang-ChiehChen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and AlanL.Yuille "Semantic image segmentation with deep convolutional nets and fully connected CRFs" *International Conference on Learning Representations (ICLR),* 2015.

[4] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick "Mask R-CNN" *arXiv preprint arXiv:1703.06870v3*, 2018.

[5] Ian J.Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair†, Aaron Courville, and Yoshua Bengio "Generative Adversarial Nets" *arXiv preprint arXiv:1406.2661*, 2014.

[6] L. Gao, X. Yi, Z. Jiang, L. Hao, and Z. Tang, "Icdar2017 competition on page object detection," *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on,* vol. 1. IEEE, 2017, pp. 1417–1422.

[7] T. Kieninger, and A. Dengel,"Applying The T-Recs Table Recognition System To The Business Letter Domain" *Proc. of International Conference on Document Analysis and Recognition (ICDAR'01)*, 2001, pp. 0518.

[8] B. Yildiz, K. Kaiser, and S. Miksch, "pdf2table: A Method to Extract Table Information from PDF Files", *Proc. of Indian International Conference on Artificial Intelligence (IICAI'05)*, 2005, pp. 1773-1785.

[9] J. Fang, L. Gao, K. Bai, R. Qiu, X. Tao, and Z. Tang, "A Table Detection Method for Multipage Pdf Documents via Visual Seperators and Tabular Structures", *International Conference on Document Analysis and Recognition,* 2011, pp.779-783

[10] T.M.Breuel, "Two Geometric Algorithms for Layout Analysis", *The Workshop on Document Analysis Systems,* 2002, pp. 188–199.

[11] S. Tupaj, Z. Shi, C. H. Chang, and H. Alam, "Extracting tabular information from text files," *EECS Department, Tufts University, Medford, USA,* 1996.

[12] F. Cesarini, S. Marinai, L. Sarti, and G. Soda, "Trainable Table Location in Document Images," *16th International Conference on Pattern Recognition, ICPR 2002. IEEE Computer Society,* 2002, pp. 236–240.

[13] A. C. e Silva, "Learning rich Hidden Markov Models in document analysis: Table location," *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on. IEEE,* 2009, pp. 843–847.

[14] T. Kasar, P. Barlas, S. Adam, C. Chatelain, and T. Paquet, "Learning to Detect Tables in Scanned Document Images Using Line Information." *IEEE Computer Society,* 2013, pp. 1185–1189.

[15] D. He, S. Cohen, B. Price, D. Kifer, and C. L. Giles, "Multiscale multi-task fcn for semantic page segmentation and table detection," *Iapr International Conference on Document Analysis and Recognition,* 2018, pp. 254–261.

[16] I. Kavasidis, S. Palazzo, C. Spampinato, C. Pino, D. Giordano, D. Giuffrida, and P. Messina, "A saliency-based convolutional neural network for table and chart detection in digitized documents," *arXiv preprint arXiv:1804.06236,* 2018.

[17] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed, "Deepdesrt: Deep learning for detection and structure recognition of tables in document images," *Iapr International Conference on Document Analysis and Recognition,* 2017, pp. 1162–1167.

[18] A. Gilani, S. R. Qasim, I. Malik, and F. Shafait, "Table detection using deep learning," *Iapr International Conference on Document Analysis and Recognition,* 2017, pp. 771–776.

[19] S. A. Siddiqui, M. I. Malik, S. Agne, A. Dengel, and S. Ahmed, "Decnt:Deep deformable cnn for table detection," *IEEE Access,* vol. 6, pp. 74 151–74 161, 2018.

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

[21] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks" *International Conference on Learning Representations (ICLR),* 2016.

[22] Long, Jonathan and Shelhamer, Evan and Darrell, Trevor, "Fully convolutional networks for semantic segmentation" *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431-3440