IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Digital Object Identifier 10.1109/ACCESS.2024.0429000

Computer Vision-Based Framework for Data Extraction from Heterogeneous Financial Tables: A Comprehensive Approach to Unlocking Financial Insights

IFTAKHAR ALI KHANDOKAR[1] (PhD Candidate) , DR. PRIYA DESHPANDE[2] (Assistant Professor)
[1]Department of Electrical and Computer Engineering (EECE) , Marquette University, Milwaukee, Wisconsin, United States (e-mail: iftakhar.khandokar@marquette.edu)

Corresponding author: Iftakhar Ali Khandokar (e-mail: iftakhar.khandokar@marquette.edu).

ABSTRACT Information extraction from financial document images is crucial in computer vision and NLP, as financial data often exists in image or PDF format, enabling organizations to analyze and make informed business decisions using OCR advancements. The table contents of financial document images are one of the prominent structures to confine important portions of data of the document and many Deep learning-based methods have been proposed to detect Table regions inside document images. The shortcomings of the current approach are that it is bounded within the detection of the table region and struggles in cases such as handling different layouts and preserving the relation among the different attributes of the table. Therefore, in this work, we proposed an end-to-end architecture to extract information from Financial table images while preserving the column row structures of the attributes within the table. We divided the task into four modules and generated synthesized data with different augmentation techniques to overcome data scarcity challenges and boost the performance of the pipeline modules. In terms of information extraction, the proposed method acquired 85% accuracy in the target invoice dataset.

INDEX TERMS Computer vision, Deep Learning, Information Extraction, Transformer Model.

I. INTRODUCTION

EXTRACTING information from document images has been a popular topic of research in Computer vision also in the Natural Language Processing domain. Applications [7] such as OCR have led to the progress of resounding systems like automatically understanding and parsing the contents of documents, and enabled operations such as searching and extracting information from documents which was previously available in image or pdf format. Extraction of document data has been very insightful to organizations whether to analyze business data to underline any fault in the current strategy or to implement or agree to any decision that can impact the advancement of the ongoing business. Although many applications have been implemented throughout the year to extract information from financial documents there is a void when it comes to the quality of information that has been extracted from the tabular structures of the document image. Due to the variety of layouts or the horizontal and vertical lines of the tabular figure, the OCR output produces garbage character output also the output of the OCR doesn't follow the pattern or distribution of data which was inclined in the original table image. To fill the stated gap in the information extraction process we proposed an end-to-end information extraction pipeline solely for tabular structures of the document image. The shortcomings that we observed in some of the state-of-the-art works are that the current approaches struggle with dealing with the different layouts of the table images and the extracted information is structured as there is no intelligence applied in forming the extracted information, [19]. So in this work, we attempted to address the current challenge of introducing an automatic information extraction process while maintaining the structure of the original image with multi-layout support. Below we have listed the contributions of this work:

1) We have tackled the prevalent issues and limitations of cutting-edge table image information extraction techniques by implementing a comprehensive pipeline for end-to-end extraction of information from table images in a structured format.
2) The work demonstrates how to leverage Transfer

Learning to overcome the Data scarcity problem and how it can contribute to performance precision even through cross-domain Data space.

3) When it comes to augmentation techniques in document image data samples the prospects are very low, in this work we have proposed some synthesized data generation techniques that impacted the performance accuracy of the Information Extraction model.
4) The proposed end-to-end structured information extraction technique from tabular images overcomes the fallacies in state-of-the-art approaches when it comes to dealing with different layout table images which impact the accurateness of the extracted data.

The article is organized in the following manner we started with examining some of the existing work in the tabular image data space highlighted in Section (II).In section (III), we discussed the working mechanism that the following work proposes. Moving on to section (IV), is the performance evaluation of the proposed methodology on existing public and target datasets, Finally in section (V) we concluded with the summary of the work addressed in the paper along with future ventures that can contribute to other domains.

II. LITERATURE REVIEW

In this section, we would like to discuss some of state of the art approaches in the domain of table image information extraction. In our study, we have found different approaches in various domains to accomplish related information extraction tasks. At first, the review focuses on table region detection tasks of different domains like in (II-A), segmentation and threshold-based approaches in the table detection domain are highlighted following that section (II-B), and (II-C), consist of the machine learning and deep learning based approaches respectively. The fourth section (II-D), shows some of the profound works in terms of text recognition from text images which is the third important module of our proposed methodology. In the final section (II-E), of this chapter, we have discussed some of the current information extraction tasks in the document image data space.

A. CROSS-AXIS SEGMENTATION TECHNIQUES FOR TABLE DETECTION

In [15], the author proposed a table detection approach over scientific and handwritten documents using horizontal and vertical line detection techniques and using the intersection points of the detected line the authors extracted the target table region, they evaluated the proposed technique with intersection pixels ration of the target table object and detected object region. In [12], the authors followed a visual separators and content layout analyzer-based technique where they attempted to generalize the area of different components in documents like paragraphs and tables with a global threshold. The authors of [20] proposed and hierarchical clustering approach to determine the region of the tabular object and evaluated this approach with an acyclic attribute graph or table DAG (directed acyclic graph) to calculate the distance to the ground truth from the prediction. In [24] the authors experimented on the MAURDOR dataset to detect table regions using row line separators and their properties which means leveraging the run-length approach to detect horizontal and vertical lines which could be the desired area for the table to exist. The authors of [27] used ground truths of table structures of scanned documents to implement a neighborhood grouping mechanism to divide the components inside table structures into columns and rows, they evaluated the approach by comparing the neighborhood groups with the ground truths to determine the optimal cluster threshold.

B. MACHINE LEARNING-DRIVEN STRATEGIES FOR TABLE DETECTION

In [56] the author followed a machine-learning-based approach for table detection, for the data they used HTML web tables and manually labeled the data, after that they applied SVM (support vector machines) and decision tree algorithm to train the table detection model, and they evaluated their approach with F-measure and acquired 95.89% performance accuracy. The authors of [48] transformed the table information extraction into the NLP domain as they applied OCR to extract text from the table image and then trained a Machine Learning model to extract contextual features from the extracted text and thus used the embedding to group the table texts acquiring 97% accuracy on the subset of the UNLV dataset. In [34], they implemented low computation cost-based text line detection techniques using the distance of pixels across the horizontal axis of the document image and for detecting the table region they utilized the distance pattern that follows within the table structure text which is they are more apart from each other than the other blocks of the document image. The article [26] is also a clustering-based approach that works in the top-to-bottom approach by clustering words that are near to each other and analyzing the formation of each cluster where the table region is determined.

C. STATE-OF-THE-ART DEEP LEARNING APPROACHES TO TABLE DETECTION

The authors of [45] emphasized that for table recognition tasks neural network-based models are a more optimal choice than machine learning or segmentation-based approaches and to justify that claim they compared the performance of their work with other contemporary techniques, [60].In [41] the authors trained a deep learning model to detect table boundaries from mobile scanned images using the FCN (fully convolutional network) deep model architecture, their model was trained with a VGG-16 architecture [16], and for evaluating the proposed approach they used the ICDAR-19 and Marmot Dataset with 0.9098 F1-Score. In [47] the authors dealt with multi-layout table images and they trained an RNN-CBPTT model to detect table regions in document images. The authors in [55], worked with web tables and they focused on learning the labels of table components and HTML documents which is used as an example in the learning phase so

Ref	Dataset	Domain	Accessibility	Model Architecture	Pre-Post Processing	Evaluation Metrics
Gatos et al. [15]	Scientific journals	Segmentation	×	Contour Detection	Hough Transform	IoU
Yalin et al. [56]	Web Document	Machine Learning	×	SVM, Decision Tree	×	F-measure
Shubham et al. [41]	Scanned Document	Deep Learning	×	FCN, VGG-16	×	F-measure
Martha et al. [42]	PDF Document	Machine Learning	×	KNN	Erosion	F-measure
Minghao et al. [29]	Table Bank	Deep Learning	✓	R-CNN	×	F-measure
Prasad et al. [44]	ICDAR 13	Deep Learning	✓	R-CNN	Smudge	IoU
Zhong et al. [63]	Financial Document	Deep Learning	×	Transformer Model	×	Tree Edit Distance
Qasim et al. [45]	UNLV	Natural Language Processing	✓	CNN	GreyScaling	String Matching
Fang et al. [12]	e-book Dataset	Segmentation	×	Layout// Segmenting	×	Precision Recall
Rashid et al. [48]	UNLV	Machine Learning	✓	Logistic// Regression	×	Precision

TABLE 1. Table Region Detection Approaches

that the model can comprehend new samples with the learned string distance threshold similar to this work the authors in [59] implemented a heuristic-based approach to detect table region which is also based on finding the optimal threshold. Another approach in [43] was proposed to introduce a conditional random field-based table component detection by comparing them to the Hidden Markov Model which is like determining the table region by the distance of clustered black pixels within the table block. In the table (2) we have listed and compared all the different techniques, [50] and domains that have been adopted in accomplishing the Table detection task in the Domain column these are the detailed versions of the keywords used { ("Segmentation": "SG"), ("Machine Learning": "ML"), ("Deep Learning": "DL"), ("Natural Language Processing": "NLP") }. The other column values are self-explanatory where different evaluation metrics along with different model architectures and types of dataset used are listed from all of the contemporary works.

D. TRANSFORMATIVE TEXT RECOGNITION APPROACHES

Text recognition has been one of the prominent research areas both in the domain of computer vision and NLP, [31] and several approaches of different aspects have been proposed to accomplish the text detection task, [54]. The introduction of OCR has also paved the way for multiple applications in [57] the authors implemented a document searching mechanism using OCR where the search key can be any text within the document and the OCR will help retrieve the text from the document which will be stored in a database later on the user can search any document by typing any text related to the document and the based on the matching to the OCR output it will lead the user to the pointed document from which the text generated from. The authors of [1] worked with Telugu text where the author used graph neural network to segment the word and character blocks within the text images and transformed the problem into a multi-class classification problem, [9] where the class is the individual characters or words present in the corpus. In [11] the authors went one step further where they implemented an OCR system to detect Chinese text and make the model lightweight by conducting ablation on the model making the model faster while preserving the accuracy of the model. The task of optical character recognition is not limited to digital documents or machine-generated text, the task has also been addressed in handwritten documents data space like in [8] the author implemented CNN-based architecture to recognize handwritten English text and evaluated the approach with a cosine similarity score. Text recognition has been very challenging due to the diversification of data space and the medium through which the data was collected many of the document images can be very blurry or of poor resolution, for example, images that were collected through a low-configuration mobile device for this type of data samples various techniques of image transformation has been also added in the recognition process to make the performance of the model smoother like in [32] the authors discussed various approaches through which this type of data samples can also be recognized by the OCR model. In the table (II-D) we have listed the different approaches and mechanisms used to implement the text recognition model. In the Evaluation Metrics column, these are the detailed versions of the keywords used { ("Character Error Rate": "CER"), ("Word Error Rate": "WER")} other than this all of the column values are self-explanatory.

Ref	Dataset	Volume	Language	Architecture	Pre-Post Processing	Evaluation Metrices
Arnab et al. [52]	EMNIST Dataset	9000 Images	English	CNN	Binary Transform	Character Error Rate
Srinidhi et al. [23]	Medical Documents	×	English	Bidirectional Encoder	×	Word Error Rate
Najam et al. [39]	Handwritten Documents	25,888 Images	Arabic	CNN-LSTM-CTC	×	Word Error Rate, Character Error Rate
JIANG et al. [22]	Research Articles	100K Images	English Numeric	DCNN, CRNN	Dilation	Word Error Rate
Du et al. [4]	Financial Documents	97 Images	Chinese	CRNN	Rotation	Word Error Rate, Precision
Dipu et al. [10]	BanglaLekha -Isolated Dataset	1,66,105 Images	Bangla	VGG-16	Exposure, Blur	Accuracy
Zhang et al. [61]	ICDAR-2015 Dataset	1,134 Images	English	CNN	Padding	Accuracy
Rawls et al. [49]	DARPA Dataset	268K Images	Arabic	bi-directional LSTM	Gaussian Noise	Character Error Rate, Word Error Rate
Anuradha et al. [4]	UCSC Dataset	400K Images	Hindi	Tesseract	×	Accuracy
Namysl et al. [40]	Newspapers	19K Images	English	CNN-LSTM	×	Word Error Rate

TABLE 2. Transformative Text Recognition Approaches

E. DATA EXTRACTION APPROACHES FROM TABLE IMAGES

Most of the current approaches in information extraction from table images after detecting the table region rely upon horizontal and vertical line detection or nearest pixel segmentation approach which relies upon an optimal threshold that is obtained from a certain group of image layout data so when the layout spaces varies from the original training data these approaches struggle to identify table cell values and to group them accordingly to the input table image, [25]. Moreover, if the layout even matches the training data then the attributes in the table image that are not related to any of the columns cause inappropriate formation of the table cells. Also in the case of any missing cell value even if the image is fully bordered there occurs a mismatch if the table image is border-less then the intensity of the formation of the table cell increases. For example in [44] the author implemented an efficient model to detect bordered and border-less tables in document images but after the detection, the techniques utilized horizontal and vertical line detection to extract and group the cell values of the columns in the image which struggles with the problems mentioned above. Another approach in [63] is the table column and row values are grouped based on the HTML tag of the web tables but in the document data domain all tables are not in web format so this approach won't be applicable where we have only the table image data as input. In [45] the author attempted to divide the table cells by detecting the column regions of the table but this approach also fell short as many cells that are not related to columns are considered as its cell value as there is no textual understanding of the cell text when forming the column cell groups. These are the challenges we have addressed in our work by combining Textual and positional features to extract the data from the table in a structured format.

III. METHODOLOGY

In this section, we'll discuss the mechanism of table detection and elaborate on the different phases of the proposed work. The first section (III-A), describes the target dataset and the layout along with the challenges that the dataset brings in working with. In section (III-C), the first phase of the pipeline is table region Detection along with techniques like Transfer Learning, Augmentation, and Model Training. In the next section (III-D), we discussed the mechanism of the table cell detection model which is similar to the table region detection task also listed the techniques used to improve the performance of the cell detection block like augmentation and synthetic data generation. In section (III-E), we highlighted the text recognition task which is about extracting text from detected table cells, and finally in section , we concluded with the structured information extraction task from table images where we combined textual and visual features to group the cell values in proper JSON structure so that it follows the

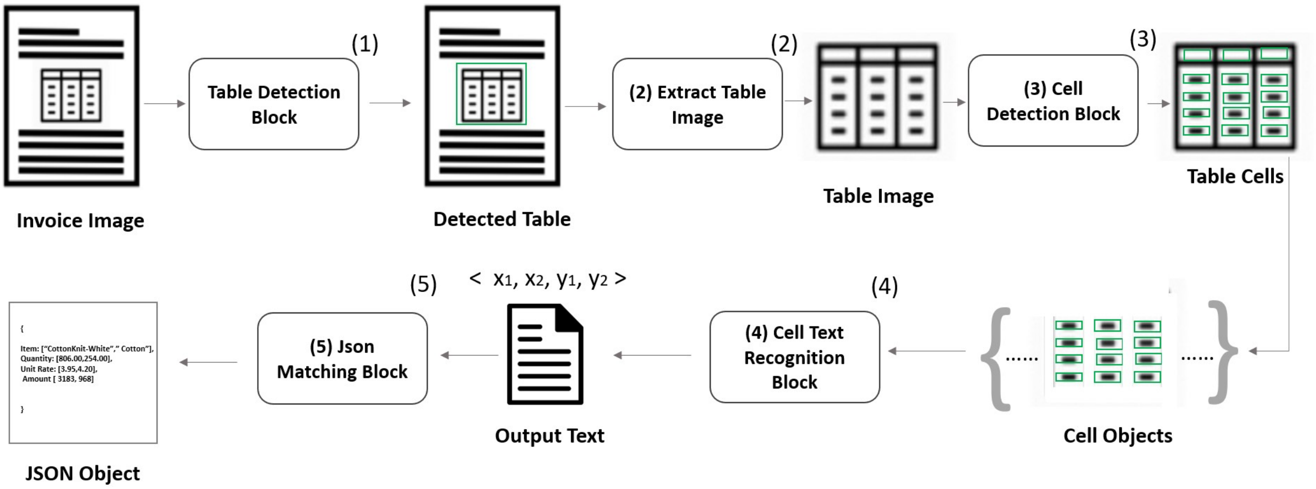


FIGURE 1. Architecture of Proposed Information Extraction Methodology

relationship depicted in the original table image. This sequence is also drawn in Figure-1 where the process starts with extracting the table image from the original invoice image in the next phase the table image is forwarded into the table cell detection block and after extracting the detected cells they are fed into the text extraction block and combining the table cells textual embeddings and visual positional embedding the final structured information of the table image is mapped into a JSON object.

A. DATASET OVERVIEW

We collected the dataset from a private bank and the dataset consists of Invoice images. This type of invoice is exchanged between companies worldwide for international trade. The sample size of the dataset is not very vast, to mention the dataset has only 4000 invoice images. The layout of the images inside the dataset varies and also there have been a variety of languages used in the images, English, Chinese, Russian, etc. To our advantage, the dataset images have the important piece of information in English to make the trade details understandable by all. All of the invoice images do not have a table figure as some of the invoice's lengths are big so the total invoice has been distributed in two images. So, the first image might not always include table figures and the original format of the dataset samples is in pdf. As we previously mentioned the size of the dataset is very insignificant for the extraction models to learn so we would like to leverage publicly available datasets to make all of the module's predictive models generalize better. The next segments will shed light on all of the distinctive modules for the information extraction process:

B. BENCHMARK DATASETS OVERVIEW

Below is the dataset description list that we have leveraged to conduct transfer learning to our predictive model. We believe the samples inside the below-mentioned dataset will provide better generalization features to our model which will impact the performance in the target dataset:

- **Marmot Dataset** [58], which contains approximately 2000 image samples in pdf format, where the sample images are the pages of research articles including tabular structures.
- **UNLV Table Dataset** [51], which consists of scanned document images collected from different domains like newspapers, Financial Reports, and magazines. In total, the dataset has 427 samples.
- **ICDAR 2013 [17] and 2019 [62],** is the Robust Reading Challenge on reading competition dataset which has almost half a million images in total, the dataset has a variety of types of documents hence we would only use the ones that have tabular structure figures. Although the dataset has Chinese language documents we are interested in detecting the structure of tabular structures therefore it won't impact the model's learning ability.
- **DeepFigures Dataset** [53], dataset has document images of tables and graphical figures. The dataset is for both figure and table detection and for our work we will focus on only the table image containing samples.
- **PubTabNet** [63], is the publicly available dataset that has more than 5,00,000 document images from scholarly article documents. One more aspect that the dataset is useful to us is that it contains the annotation of the table structures in HTML format meaning all the text of the table cells and the hierarchy of the table structure, [51]. The problem is the dataset has only an image of the table boundary hence there is no background for the table to be detected. To utilize the images we placed the tables on white background images using the height and width of the table boundary. We generated random coordinates for the table in the white blank images within the range of the height and width of the blank image. Having generated the random bounding box we placed the image

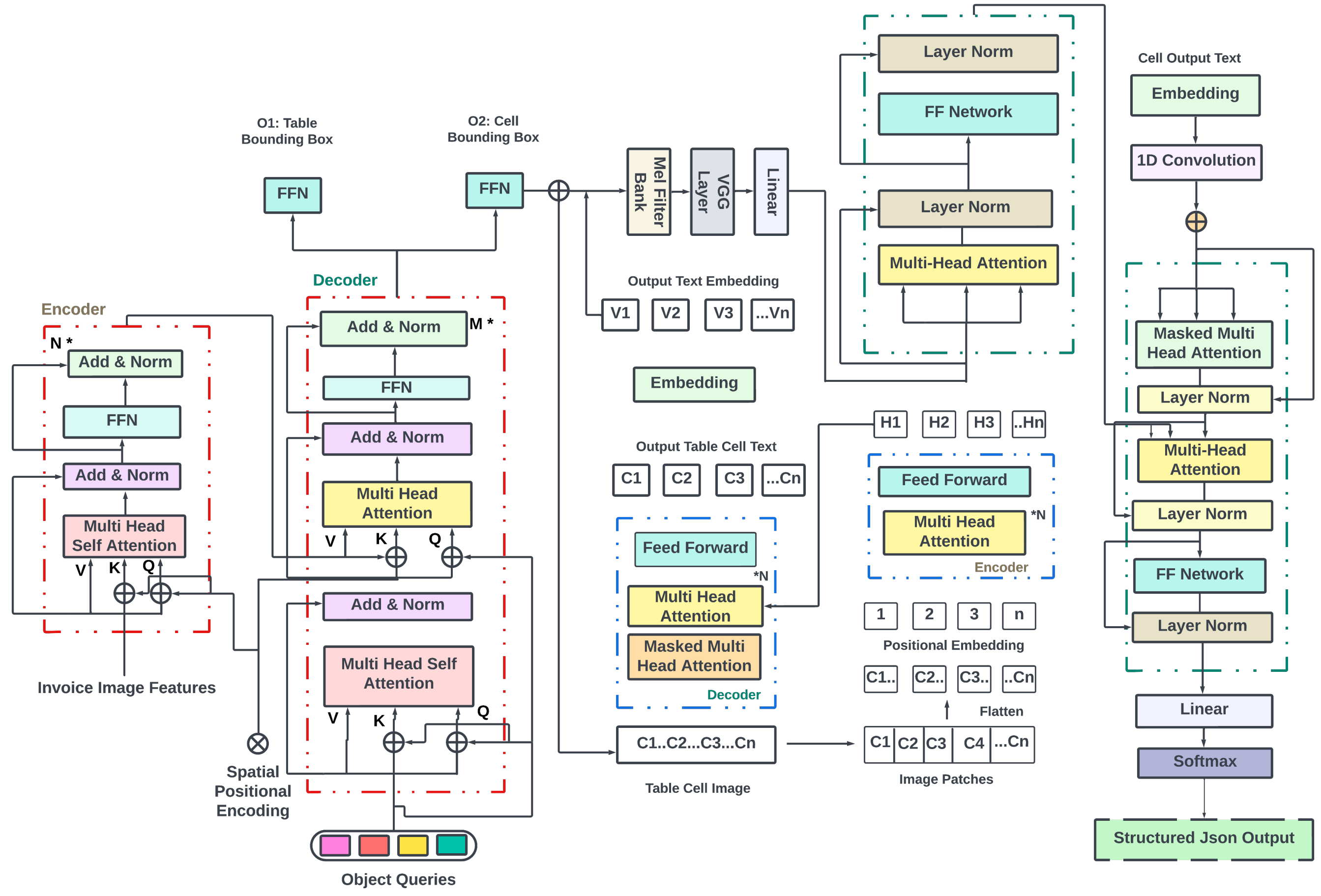


FIGURE 2. Table Information Extraction Block Diagram

in that part of the blank image to construct a full-fledged sample out of the table image and recorded the randomly generated bbox (bounding box) in a JSON object to load it during the training phase.

C. DETECTING TABLE REGIONS

The first module detects the table region so that using the coordinates of the table we can extract the table image from the invoice image and then forward the table image to the next extraction modules. As we mentioned before the volume of our target dataset is very small so the probability of the predictive model gaining a decent classification performance is very low, [36]. Hence we collected some publicly available document data consisting of table figures. The dataset we collected is pre-annotated with table regions which will save the manual annotation effort and time in our task. In the next section, we briefly describe the publicly available datasets that we have used to train the model.

1) Training Procedures for Table Region Detection Model

For training the table region detection model we used the DETR: End-to-End Object Detection with Transformers model [6], which was proposed by the Meta research team. The model consists of a CNN layer which usually extracts features from the input image and it is the Backbone of the model. Other backbone models like RestNet-50, ResNet-101, and U-net can also be used in this layer to act as the backbone of the model. The final output consists of an Encoder-Decoder transformer architecture where the model evaluates the predicted region with the ground truth through bipartite matching. In the final layers the coordinate of the target object is determined through the Relu activation function and to assign a label to the detected object Softmax function is used. Compared to other popular object detection architectures like RCNN, FR-CNN, and VGG the DETR model is much faster in detecting target objects, and also in terms of detecting smaller objects in images the performance is remarkable, [19]. The model is pre-trained on the Imagenet dataset which means the model has some generalization and classification capabilities and it's always prescribed to use architectures that are already trained on some sort of dataset cause it has a performance advantage compared to blank weight models.

2) Leveraging Transfer Learning with Benchmark Datasets

In the first stage, we started the Transfer Learning procedure which uses all the existing dataset samples to train the model

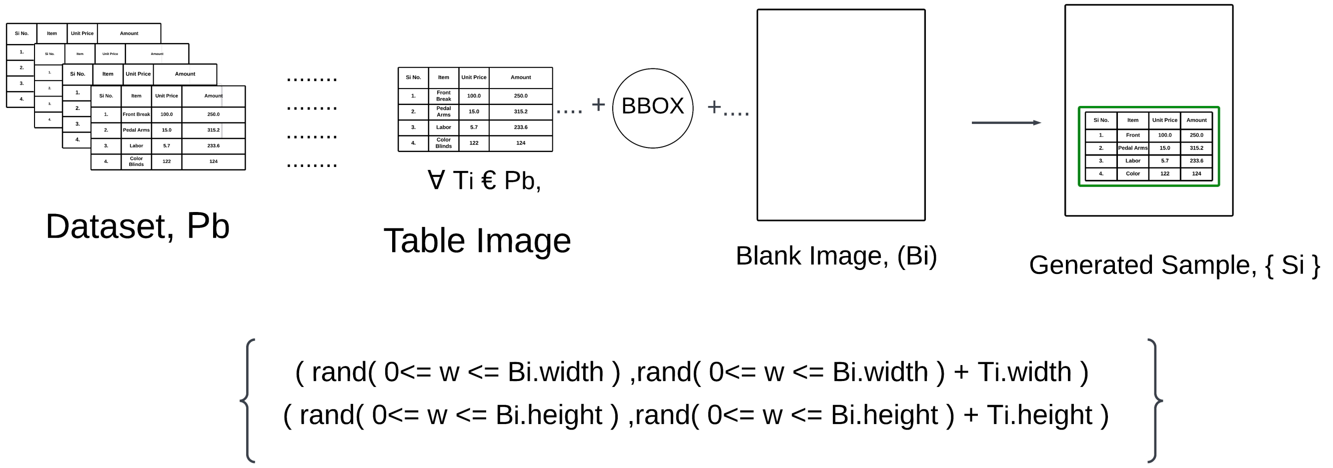
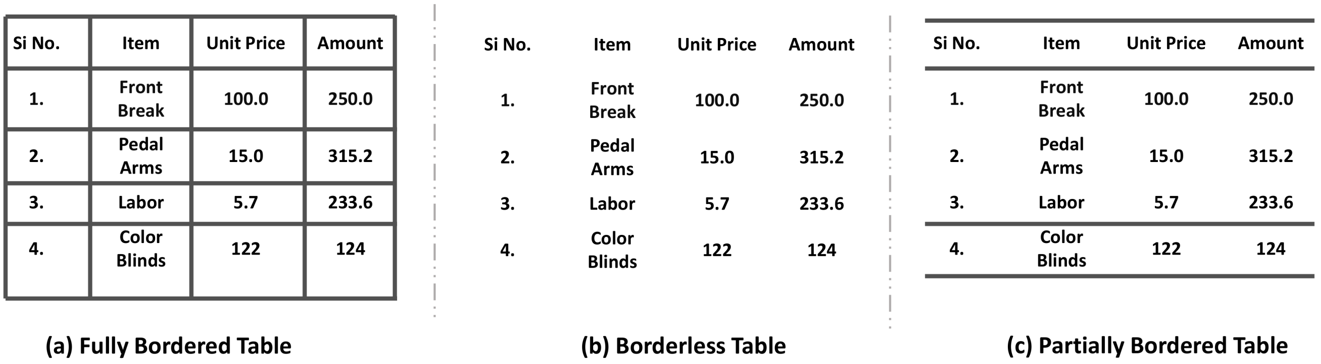

Si No.
Item
Unit Price
Amount
....
∀ Ti € Pb,
Dataset, Pb
Table Image
.... +
BBOX
+....
Blank Image, (Bi)
Generated Sample, { Si }
(rand(0<= w <= Bi.width) ,rand(0<= w <= Bi.width) + Ti.width)
(rand(0<= w <= Bi.height) ,rand(0<= w <= Bi.height) + Ti.height)

FIGURE 3. Training Sample Generation from Table Dataset



Si No.	Item	Unit Price	Amount
1.	Front Break	100.0	250.0
2.	Pedal Arms	15.0	315.2
3.	Labor	5.7	233.6
4.	Color Blinds	122	124

(a) Fully Bordered Table

Si No.	Item	Unit Price	Amount
1.	Front Break	100.0	250.0
2.	Pedal Arms	15.0	315.2
3.	Labor	5.7	233.6
4.	Color Blinds	122	124

(b) Borderless Table

Si No.	Item	Unit Price	Amount
1.	Front Break	100.0	250.0
2.	Pedal Arms	15.0	315.2
3.	Labor	5.7	233.6
4.	Color Blinds	122	124

(c) Partially Bordered Table

FIGURE 4. Types of Table Layouts

$$
\left\{
\begin{array}{ll}
\sum_0^i D_t \longrightarrow & \{\ (D_i \subseteq TD_M) \equiv (D_i \subseteq TD_I) \equiv (D_i \subseteq TD_U) \equiv (D_i \subseteq TD_D) \equiv (D_i \subseteq TD_P)\ \} \\
\sum_0^j D_v \longrightarrow & \{\subseteq TD_M \backslash D_i) \equiv (D_i \subseteq TD_I \backslash D_i) \equiv (D_i \subseteq TD_U \backslash D_i) \equiv (D_i \subseteq TD_D \backslash D_i) \equiv (D_i \subseteq TD_P \backslash D_i)\ \} \\
\sum_0^k D_e \longrightarrow & \{\ \subseteq TD_M \backslash (D_i \cup D_v)) \equiv (D_i \subseteq TD_I \backslash (D_i \cup D_v)) \equiv (D_i \subseteq TD_U \backslash (D_i \cup D_v)) \\
 & \quad \equiv (D_i \subseteq TD_D \backslash (D_i \cup D_v)) \equiv (D_i \subseteq TD_P \backslash (D_i \cup D_v))\ \} \\
\text{where, } D_t \longrightarrow \ train \ , \ D_v \longrightarrow \ validation \ , \ D_e \longrightarrow \ test &
\end{array}
\right. \tag{1}
$$

initially, and later on, we'll fine-tune the model on the target dataset, [13]. The annotation format of all the existing datasets Marmot (TD$_M$), ICDAR(TD$_I$), UNLV Dataset(TD$_U$), DeepFigure Dataset(TD$_F$) and PubTabnet Dataset(TD$_P$) varies from one another hence we wrote the script for all the individual dataset annotations and converted them into JSON format which is desired by the DETR architecture. The target dataset is annotated by us according to the desired format from the beginning of the processing the first stage, we started the transfer learning procedure which uses all the existing dataset samples to train the model initially, and later on, we'll fine-tune the model on the target dataset. The annotation format of all the existing datasets Marmot (TD$_M$), ICDAR(TD$_I$), UNLV Dataset(TD$_U$), DeepFigure Dataset(TD$_D$) and PubTabnet Dataset(TD$_P$) varies from one another hence we wrote the script for all the individual dataset annotations and converted them into JSON format which is desired by the DETR architecture. The target dataset is annotated by us according to the desired format from the beginning of the process. In the transfer learning phase for the train, validation, and test set we picked an equal number of samples from all of the different datasets and split them into train, test, and validation sets, [35]. This way all the data subspace consisted of samples from all the different datasets, some of the datasets are bigger in volume compared to the other datasets for those we adopted the bagging technique to create similar subsets of the data using unpicked samples given in the equation 1 above.

3) Optimizing Table Detection with Data Augmentation

During the training phase, we included some augmentation techniques to feed the predictive model not because we believe the number of data samples is low but to introduce the model to a different form of the training data so that it can extract some meaningful features, [37]. As we are working with document image data the rotation technique which is quite popular in object detection tasks won't be beneficial in the learning phase rather it might confuse the model. Therefore we used some of the basic augmentation techniques like dilation, erosion, and smudge. Additionally, we introduce a new augmentation technique for document images like shifting the location of the table from the original image using the annotated coordinates of the tabular structure given that the document has some empty space to accommodate. Another technique we applied is swapping tables, [3] within two randomly picked images if the areas of the two tables are compatible with one another. we included all these transformed data samples in all the training folds for the model to encounter. some of the sample examples of data augmentation are shown below:

D. MODEL TRAINING FOR TABLE CELL IDENTIFICATION

The second task is to detect all the different cell that has the attribute values of the table across the structure, the cell object is the text that is sufficient to convey the information it bears. The cell object can be a value that may be a heading of a column or a respective value of any specific column. we won't be focusing on the relation between cells for now in this phase we will only focus on detecting all the cell objects within the table. Some other cell values don't belong to any column this type of value can be a summary of the tabular contents or any type of relevant information related to the invoice. There are different types of table layouts some tables have grid lines others might be completely borderless. In grid-style tables [46], the cells are quite visible and easy to detect but in the borderless table domain, the model has to find an optimal coordinate so that the structure pattern of column row format can be found or data can be grouped to that format. The architecture of the cell detection model is the Tranformer-based DETR predictive model. This architecture will be forked to the table region detection model where the first model will provide the detected table coordinates and this model will try to separate, group, or detect the cell object within the detected table image. After training our model with the Pubtabnet dataset we fine-tuned the model on the target data. The target dataset table images were labeled with the cell coordinates for the model to fit. As the volume of the dataset is small the annotation didn't require much time and effort. the main learning materials were provided by the Pubtabnet dataset and the final finetuning made the model able to comprehend the target data sample structure.

1) Enhancing Table Cell Detection Through Data Augmentation

The dataset we will utilize for training the cell detection model is the Pubtabnet dataset [63], The reason for selecting this dataset is that this is the only dataset that has been annotated with the coordinates of the cell object within every table image in the dataset. The other publicly available dataset has only annotated table structure coordinates so for this task these datasets are not feasible to use as it will consume a lot of time and effort to annotate the data. The Pubtabnet dataset [63], has more than 5 million table samples with all the cell coordinates and values annotated in HTML structure. Though the volume of the data is pretty massive we did introduce an augmentation block to amplify the dataset more which will play an impactful aspect in teaching the model the insight patterns. The augmentation has been categorized into two different techniques which are randomly deleting any of the columns (C_i) from the table image (T) which could be up to (n-1) number of columns. The second augmentation technique is randomly deleting any of the cell objects (O_j) from randomly selected columns (C_i). Using these two techniques will introduce the model to some different versions of the data samples which will help the model to comprehend the important patterns and will reduce the risk of overfitting the model.

E. BUILDING A TEXT RECOGNITION MODEL FOR TABLE CELLS

After getting all the cell objects of the table image we have to recognize the text within the cell objects. We have trained a custom model for detecting the text of the cell objects cause if we use open-source OCR software like Tesseract, EasyOCR, PaddleOCR, etc. the accuracy of recognizing text is very poor. The reason for this behavior is that the cell object images are of very small dimensions and most of the open source OCRS are trained on full-length a4 size document images hence the model fails to recognize patterns from such small text images. In this task, we have used the TR-OCR architecture [30], to train the model and it is based on the encoder-decoder architecture where ViTransformer acts as the encoder and Roberta is the decoder counterpart. Every transformer layer consists of a fully connected feed-forward network and a multi-head self-attention mechanism. The advantage of the TR-OCR model is that it comes with pre-trained computer vision and NLP transformers which makes the text recognition part more precise. The model is comparatively faster as it is a convolution-free model and doesn't require any complicated pre or post-processing mechanisms. Before training the model on the cell object data we pre-trained it on some of the publicly available OCR datasets like FunSD [21], DocBank [28], IIIT-AR-13K [38] to push the text recognition model's generalization capability.

To feed the text recognizer model more data so that the model can detect text in smaller region image blocks we collected some publicly available textual datasets. Using these text datasets especially financial textual datasets like [33],

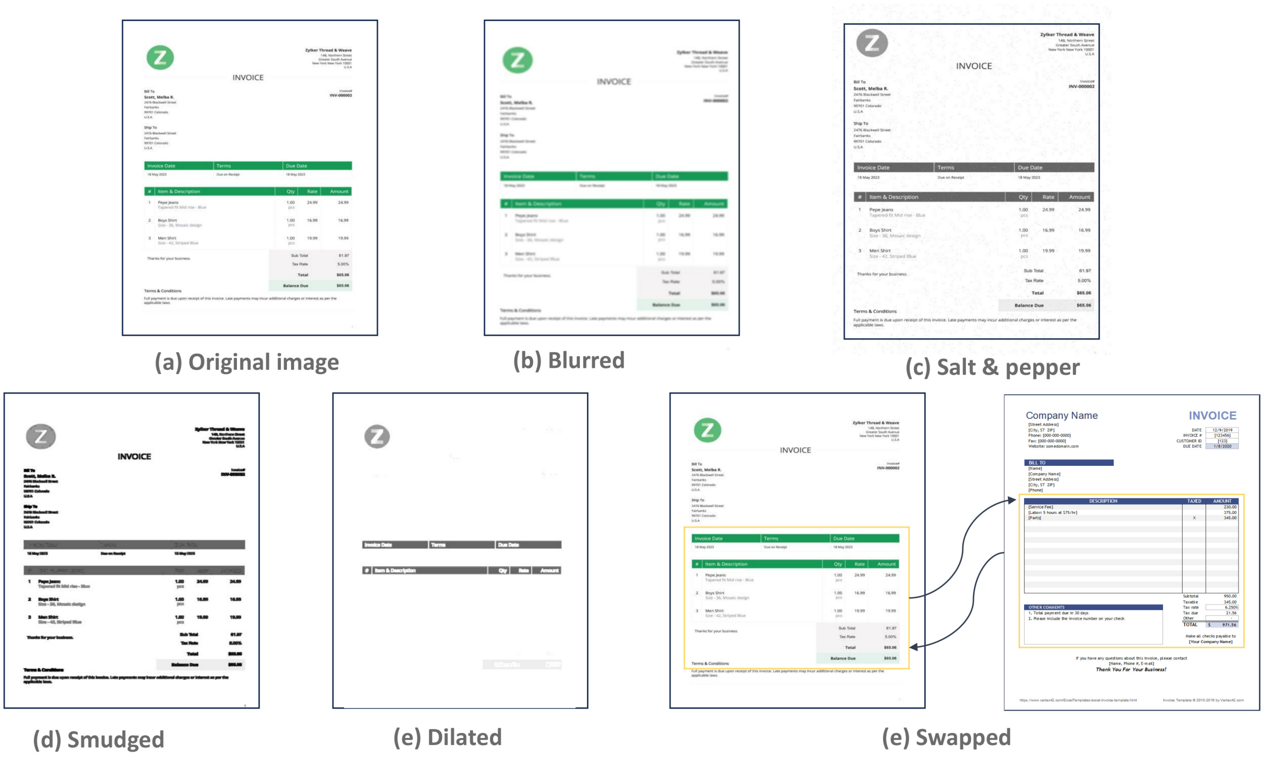
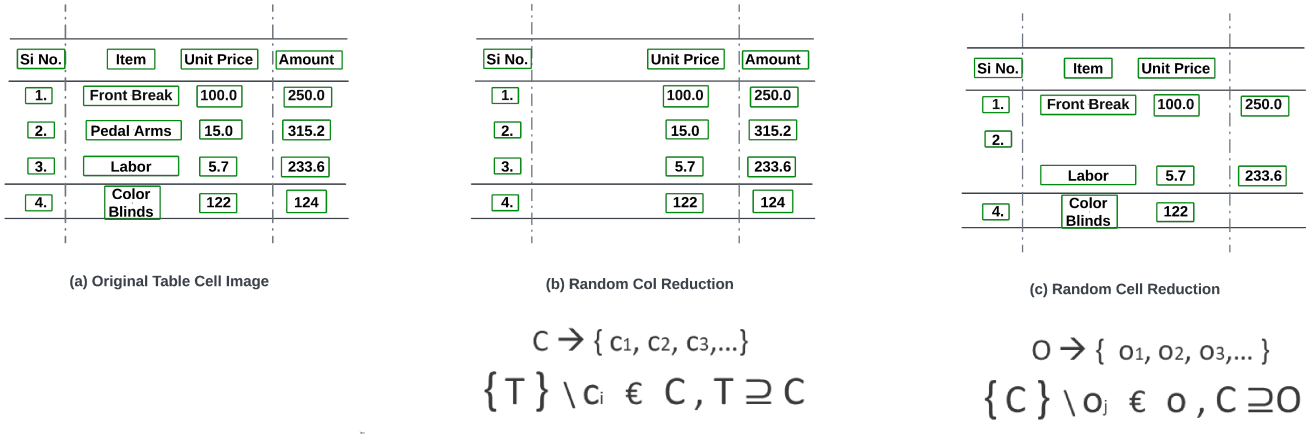


FIGURE 5. Augmented Training Samples for Table Detection Model



Si No.	Item	Unit Price	Amount
1.	Front Break	100.0	250.0
2.	Pedal Arms	15.0	315.2
3.	Labor	5.7	233.6
4.	Color Blinds	122	124

(a) Original Table Cell Image

Si No.		Unit Price	Amount
1.		100.0	250.0
2.		15.0	315.2
3.		5.7	233.6
4.		122	124

(b) Random Col Reduction

Si No.	Item	Unit Price	
1.	Front Break	100.0	250.0
2.			
	Labor	5.7	233.6
4.	Color Blinds	122	

(c) Random Cell Reduction

FIGURE 6. Augmented Training Samples for Table Cell Detection Model

[14] etc. as our focus target dataset is from the financial domain. we will generate some small dimension text image blocks and train the model on such generated samples. At first, for any collected text dataset we applied tokenization on the textual body of the document (D_n) and used these tokens to integrate with the cell object images. The cell object images (O_j) of the tables (T_p) in Pubtabnet [63], and the target dataset were selected randomly to construct a cell object set. From this set, we picked one cell object image at a time and emptied the content of the cell image (O_j), and using the OpenCV [5], tool we put the text inside the blank cell image. Therefore we generated small cell object samples (S_i) where we have the newly inclined text image and the ground truth (GT_i) of the image is the selected token (T_i). After training the model on the synthesized data we fine-tuned the model on the target dataset which was annotated manually with the textual content within the cell object images of the tables.

F. EXTRACTING STRUCTURED INFORMATION FROM TABLE IMAGES

This is the main contribution block of this work where to overcome the current problems discussed in (II-E), in the

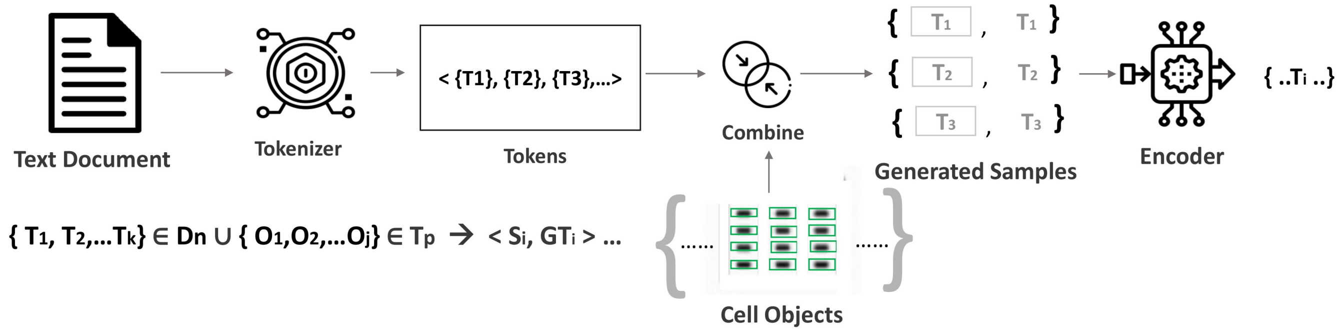


FIGURE 7. Textual Cell Image Generation from Related Document Dataset

extraction process we have proposed to combine the positional and textual features of the tables cell blocks to generate structured JSON object where the key will be the value of the column and the pointed object by the key is a list of all the values of that column and all these tuples will be confined in a single JSON object, [18]. There are some values in the table that have no relation with any of the existing columns within the table and these single values affect the optimal formation of the extracted information. Therefore as a solution to this problem, this type of single value won't be injected in any of the columns value lists rather it will reside in the JSON as a stand-alone object with a predefined key "single_object", so it might be convenient to search for any information within this single object and keeping the column values within their related space.

In the first phase, we implemented an HTML to JSON generation block for all the table images in the PubTabnet dataset [63], as the annotation of the table values was in HTML format and the target Invoice image dataset was annotated in the desired strucured JSON format. After transforming the annotation we used the Bidirectional and Auto-Regressive Transformers model to train the Table image to Structured Information Extraction model. The Bidirectional and Auto-Regressive Transformers model utilizes a masked language model" (MLM) to generate some masks from the randomly selected token from the input and tries to predict those masks. The model is a transformed-based architecture with multiple self-attention layers, [2] and a feed-forward neural net that encodes the input sequence. The model employs the input encoding and currently generated sequence to predict the next tokens and conclude the generation process with the complete target sequence.

To provide the model with the contextual and task-oriented feature samples we generated synthesized data from the annotation so the model can comprehend the output pattern based on the different forms of input. The original annotation of the PubTabnet dataset [63], was in HTML form thus when concerting the HTML structure into JSON format we randomly deleted some of the cell values informed of the (<td>) tag and also in some cases the whole column has been removed to generate augmented sample which removes the <tr> - > </tr> tag. Based on the modified HTML table code the data transformation block generated the JSON counterpart keeping all the table values in the HTML. This will enable the model to confound over-fitting and form any static rule to generate the output hence the model can apply the attention block to the different embedding to utilize the positional and textual embeddings most rationally.

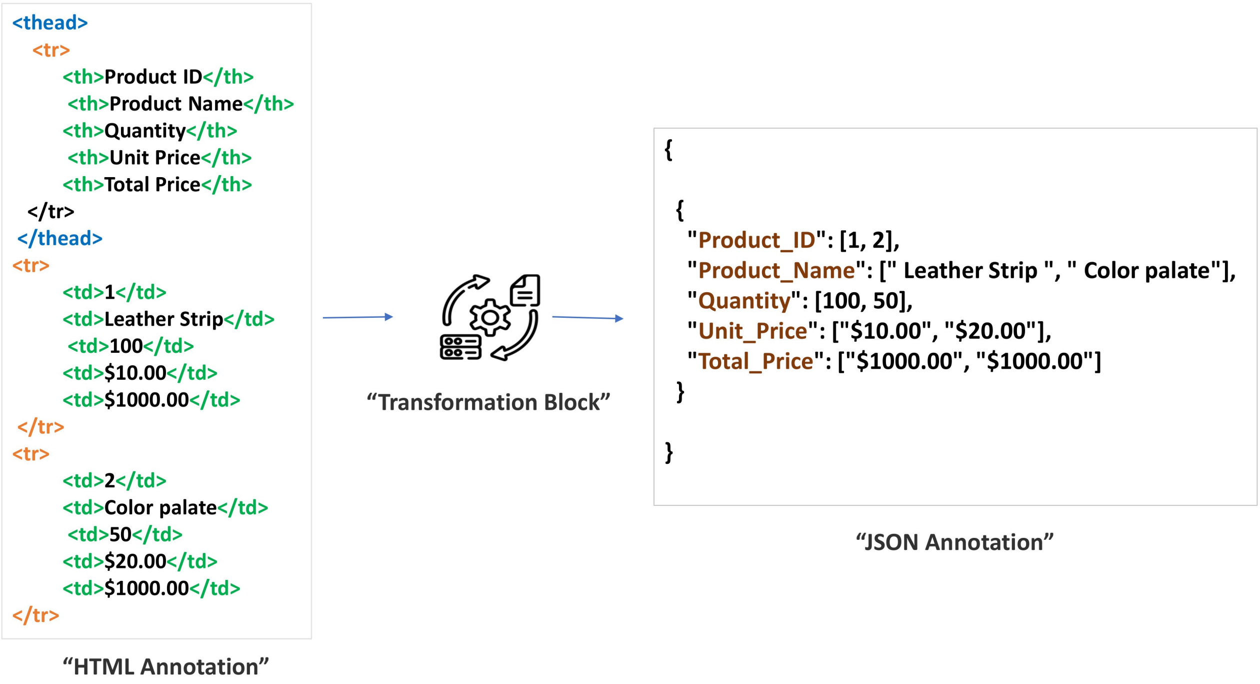
IV. PERFORMANCE METRICS AND ANALYSIS

The performance evaluation section has been divided into four subsections and each subsection is about the efficiency of all the individual models in our proposed pipeline. The first section (IV-A), highlights the performance of the table region detection block following the second subsection (IV-B), which focuses on the Table cell detection block module performance evaluation. Likewise, the third subsection (IV-C), is the Text recognition blocks evaluation and in the final subsection (IV-D), the table image to JSON matching blocks performance evaluation is documented.

A. EVALUATING TABLE REGION DETECTION MODEL

We evaluated the Table detection model using the Intersection over Union metric which is a widely popular metric in the object detection domain. The IoU score is the ratio of the ground truth which is the area of the actual target object and the predicted region of the target object. The dividend part is the intersection of the ground truth region { $X1_g$, $Y1_g$,$X2_g$,$Y2_g$ }and the predicted region { $X1_p$, $Y1_p$,$X2_p$,$Y2_p$ } and the divisor is the total area of the predicted and ground truth region combined which is the union opinion.

The result we have shown in the evaluation section is only based on the prediction that has a confidence score over 90% hence the predicted objects that have confidence above the threshold (0.9) are considered a perfect match and deemed misclassified otherwise. Based on these criteria we have evaluated all the documents within the datasets and have calculated the effectiveness with three matrices which are (Recall, Precision, and F1-Score). We have also included the other public dataset evaluation scores along with the target Invoice Dataset in the below table. The True Positive (TP) value is the model (M_t) predicted a table region (D_t) with 90%



```
<thead>
   <tr>
        <th>Product ID</th>
         <th>Product Name</th>
        <th>Quantity</th>
         <th>Unit Price</th>
        <th>Total Price</th>
  </tr>
 </thead>
<tr>
        <td>1</td>
        <td>Leather Strip</td>
         <td>100</td>
        <td>$10.00</td>
        <td>$1000.00</td>
 </tr>
<tr>
        <td>2</td>
        <td>Color palate</td>
         <td>50</td>
        <td>$20.00</td>
        <td>$1000.00</td>
</tr>
```

"HTML Annotation"

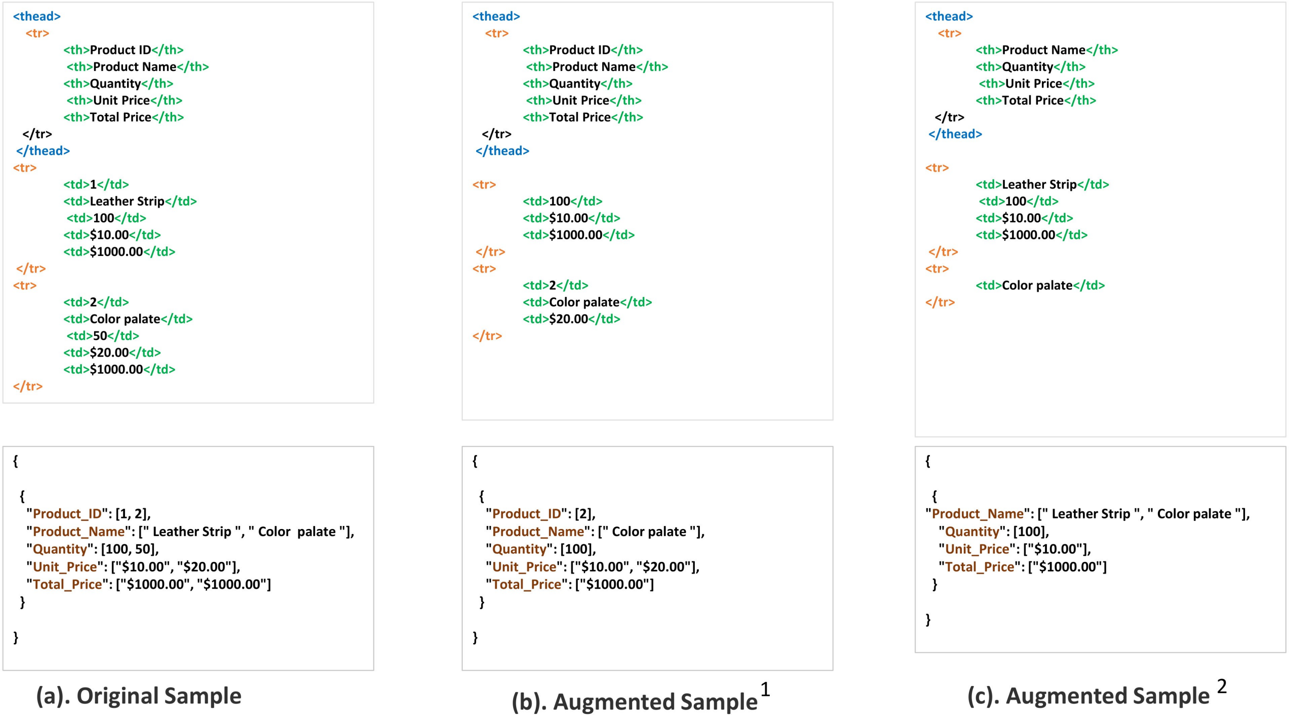
```
{

  {
   "Product_ID": [1, 2],
   "Product_Name": [" Leather Strip ", " Color palate"],
   "Quantity": [100, 50],
   "Unit_Price": ["$10.00", "$20.00"],
   "Total_Price": ["$1000.00", "$1000.00"]
  }

}
```

"JSON Annotation"

FIGURE 8. Generating JSON Object from HTML Annotation Structures



```
<thead>
   <tr>
        <th>Product ID</th>
         <th>Product Name</th>
        <th>Quantity</th>
         <th>Unit Price</th>
        <th>Total Price</th>
  </tr>
 </thead>
<tr>
        <td>1</td>
        <td>Leather Strip</td>
         <td>100</td>
        <td>$10.00</td>
        <td>$1000.00</td>
 </tr>
<tr>
        <td>2</td>
        <td>Color palate</td>
         <td>50</td>
        <td>$20.00</td>
        <td>$1000.00</td>
</tr>
```

```
{

 {
  "Product_ID": [1, 2],
  "Product_Name": [" Leather Strip ", " Color  palate "],
  "Quantity": [100, 50],
  "Unit_Price": ["$10.00", "$20.00"],
  "Total_Price": ["$1000.00", "$1000.00"]
 }

}
```

(a). Original Sample

```
<thead>
   <tr>
        <th>Product ID</th>
         <th>Product Name</th>
        <th>Quantity</th>
         <th>Unit Price</th>
        <th>Total Price</th>
  </tr>
 </thead>

<tr>
        <td>100</td>
        <td>$10.00</td>
        <td>$1000.00</td>
 </tr>
<tr>
        <td>2</td>
        <td>Color palate</td>
        <td>$20.00</td>
</tr>
```

```
{

 {
  "Product_ID": [2],
  "Product_Name": [" Color palate "],
  "Quantity": [100],
  "Unit_Price": ["$10.00", "$20.00"],
  "Total_Price": ["$1000.00"]
 }

}
```

(b). Augmented Sample 1

```
<thead>
   <tr>
        <th>Product Name</th>
        <th>Quantity</th>
         <th>Unit Price</th>
        <th>Total Price</th>
  </tr>
 </thead>

<tr>
        <td>Leather Strip</td>
         <td>100</td>
        <td>$10.00</td>
        <td>$1000.00</td>
 </tr>
<tr>
        <td>Color palate</td>
</tr>
```

```
{

 {
"Product_Name": [" Leather Strip ", " Color palate "],
  "Quantity": [100],
  "Unit_Price": ["$10.00"],
  "Total_Price": ["$1000.00"]
 }

}
```

(c). Augmented Sample 2

FIGURE 9. Augmented Training Samples Generation for Table to JSON Matching Model

confidence and the ground truth (G_t)contains a table object in that region, on the other hand, the True Negative (TN) is where the model (M_t) predicted no table and the ground truth (G_t) doesn't contain any table object. False positive (FP) is when the model (M_t) predicts a table object (D_t)but it's not in the Ground Truth (G_t) set and the False Negative (FN) is

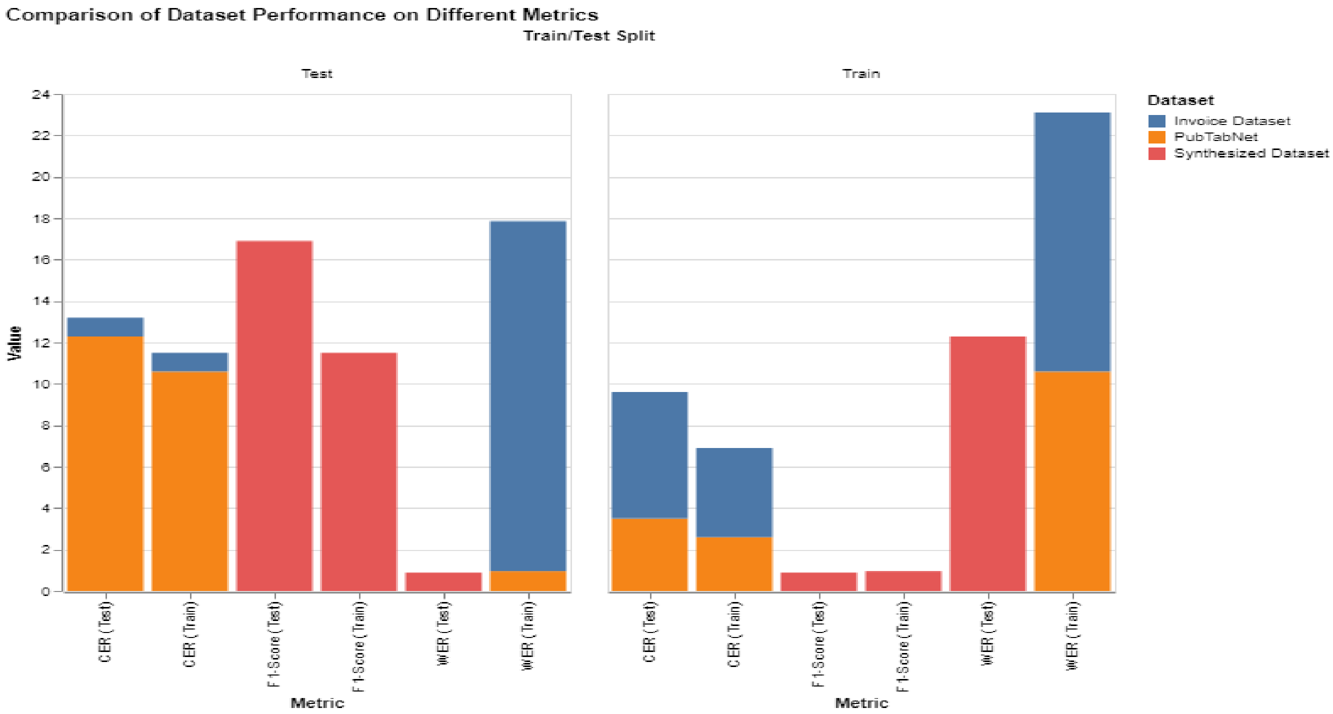


FIGURE 10. Text Recognition Model Evaluation on Different Evaluation Metrics.

tf the model predicted no table object but the Ground Truth (G_t) contains table object (T_i). The expressions are given in the equation below:

Analyzing the results we observed that the FR-CNN model acquired a higher performance score in both the test and train set. however, in this experiment, we couldn't include all the data samples available within the dataset hence the results achieved from this initial experiment can not be assured for generalization. In the next experiment, we would like to amplify the test set by carefully selecting different layout table images that will indicate the actual impact of the predictive model.

Dataset	Recall		Precision		F1-Score	
	train	test	train	test	train	test
Marmot	0.9121	0.8921	0.9180	0.8879	0.9150	0.8876
UNLV	0.9409	0.9123	0.9512	0.9079	0.9460	0.8979
ICDAR	0.7998	0.6940	0.9748	0.8279	0.8786	0.8879
DeepFigures	0.9145	0.8878	0.8527	0.9018	0.8829	0.8921
PubTabNet	0.9594	0.9135	0.9458	0.9199	0.9425	0.8979
***Invoice Dataset**	**0.9822**	**0.9615**	**0.9729**	**0.9740**	**0.9539**	**0.9677**

TABLE 3. Table Detection Model Evaluation on Different Datasets

B. EVALUATING MODEL PERFORMANCE IN TABLE CELL BLOCK DETECTION

The Table cell is similar to the table detection task which fundamentally falls under the object detection domain, hence the evaluation process for the cell detection model is identical to the table detection model evaluation. To detect the performance of the model we used the IoU score also mentioned in the above equation which will indicate if the model can draw optimal boundaries between the different cell objects which is also like separating them from one another and maintaining all the related pixel values of the cell object within one boundary. In the table below we have listed the IoU scores for different thresholds acquired in the three datasets that were used to train the model:

We further evaluated the model with Recall, Precision, and F1-Score just like the Table region evaluation process. The following evaluation has been conducted using the (0.9) IoU threshold and attributes used to calculate the metrics have been described in the table region detection evaluation section which are True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The evaluation was done on the three training datasets and the table below consists of the performance record acquired by the model.

C. EVALUATION OF TEXT RECOGNITION BLOCKS

The text recognition block is one of the important key blocks as the information extraction percentage is highly dependent on it. We elected cosine similarity ($\cos(\varphi)$) as the evaluation metric for the text recognizer block. The threshold or similarity score we have set for the evaluation is (1.0) which means a 100% match between the predicted (X_i) and ground truth vector (Y_i). The equation of the cosine similarity metric ($\cos(\varphi)$) is given below where the dot product of the predicted word vector (X_i) and ground truth word vector (Y_i) is divided by the norm of those two vectors. The more similar the two vectors are the more the similarity score ($\cos(\varphi)$) will be close to ($\cos(\varphi) \cong 1.0$).b The test object is all the potential text inscribed cell objects and if the similarity score ($\cos(\varphi)$)

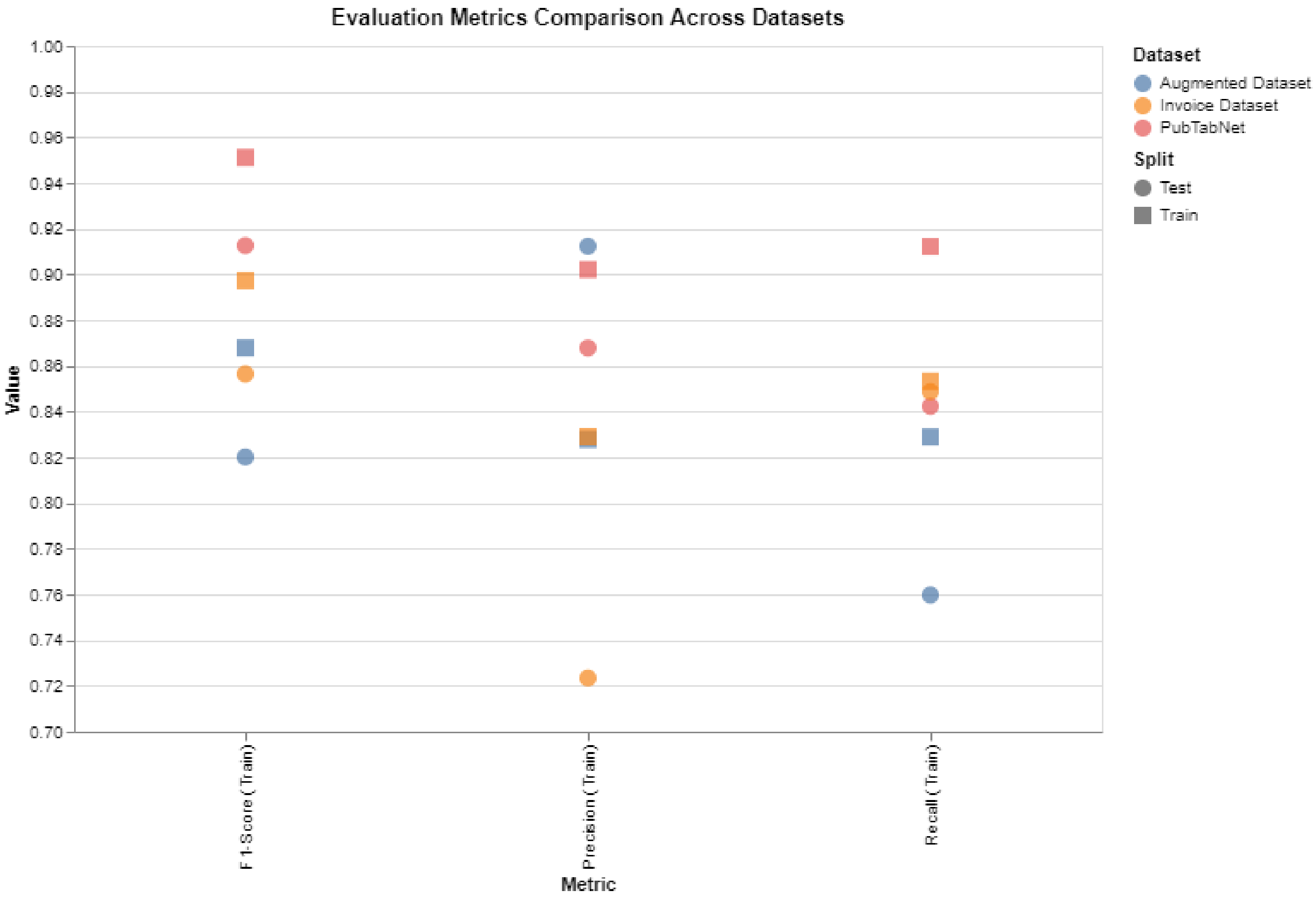
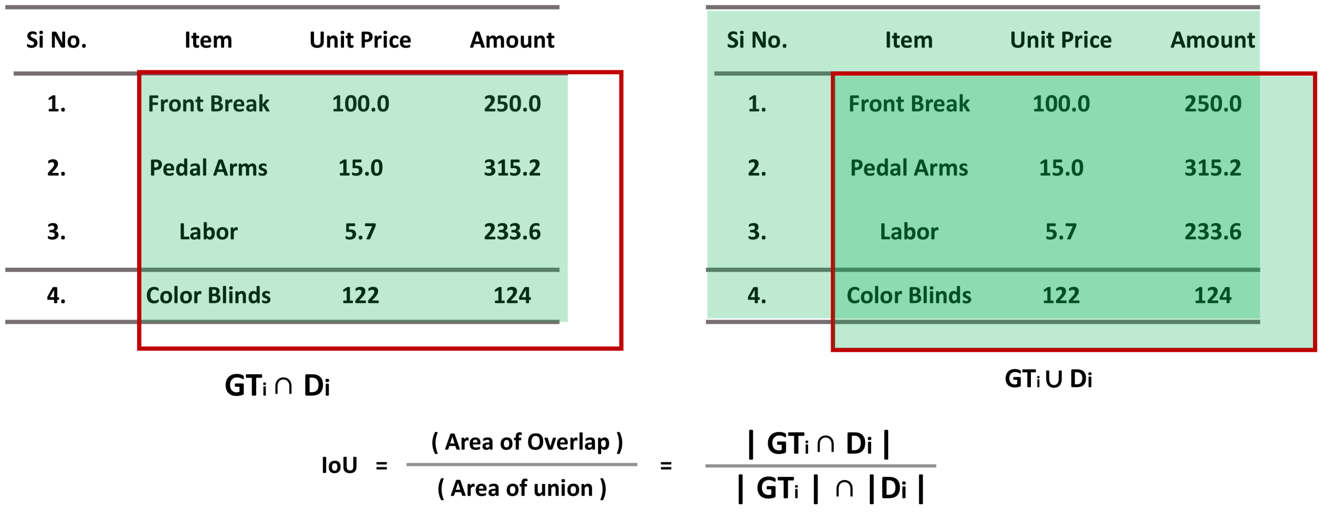


FIGURE 11. Text Recognition Model Evaluation on Different Evaluation Metrics.



Si No.	Item	Unit Price	Amount
1.	Front Break	100.0	250.0
2.	Pedal Arms	15.0	315.2
3.	Labor	5.7	233.6
4.	Color Blinds	122	124

$GT_i \cap D_i$

Si No.	Item	Unit Price	Amount
1.	Front Break	100.0	250.0
2.	Pedal Arms	15.0	315.2
3.	Labor	5.7	233.6
4.	Color Blinds	122	124

$GT_i \cup D_i$

$$\text{IoU} = \frac{(\text{Area of Overlap})}{(\text{Area of union})} = \frac{|GT_i \cap D_i|}{|GT_i| \cap |D_i|}$$

FIGURE 12. Table Detection Model Evaluation Metric with Intersection over Union (IoU)

for the predicted text equals ($\cos(\varphi) \cong 1.0$) then we have considered it as a successful recognition otherwise if the similarity score ($\cos(\varphi)$) is less than the threshold ($\cos(\varphi) \leq 1.0$) then the case will fall under a direct miss which means the accuracy is (0%) for that cell object.

Having set up the criteria we have listed the performance score in the below table with the following three metrics which are Character Error Rate (CER%), Word Error Rate

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}} \qquad \text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive (FP)}}$$

$$\text{F1-Score} = \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

where,

$$\text{TP: } M_t \rightarrow D_t,\ D_t \in G_{t}\{Ti..\} \ \mid\ \text{TN: } M_t \rightarrow \emptyset,\ \emptyset \in G_t\{\emptyset\} \ \mid\ \text{FP: } M_t \rightarrow D_t,\ D_t \notin G_t\{\emptyset\} \ \mid\ \text{FN: } M_t \rightarrow \emptyset,\ \emptyset \notin G_{t}\{Ti..\}$$

Dataset	0.6		0.7		0.8		0.9		Weighted Avg.	
2-11	train	test	train	test	train	test	train	test	train	test
PubTabNet	0.969	0.9131	0.957	0.9079	0.9512	0.8976	0.897	0.8321	0.94355	0.8876
Augmented Dataset	0.979	0.9157	0.966	0.9234	0.9391	0.88246	0.850	0.8378	0.9335	0.8898
***Invoice Dataset**	0.943	0.8976	0.9342	0.9072	0.9258	0.9124	0.9011	0.8693	**0.9260**	**0.8966**

TABLE 4. Table Cell Detection Model Evaluation with Different Thresholds

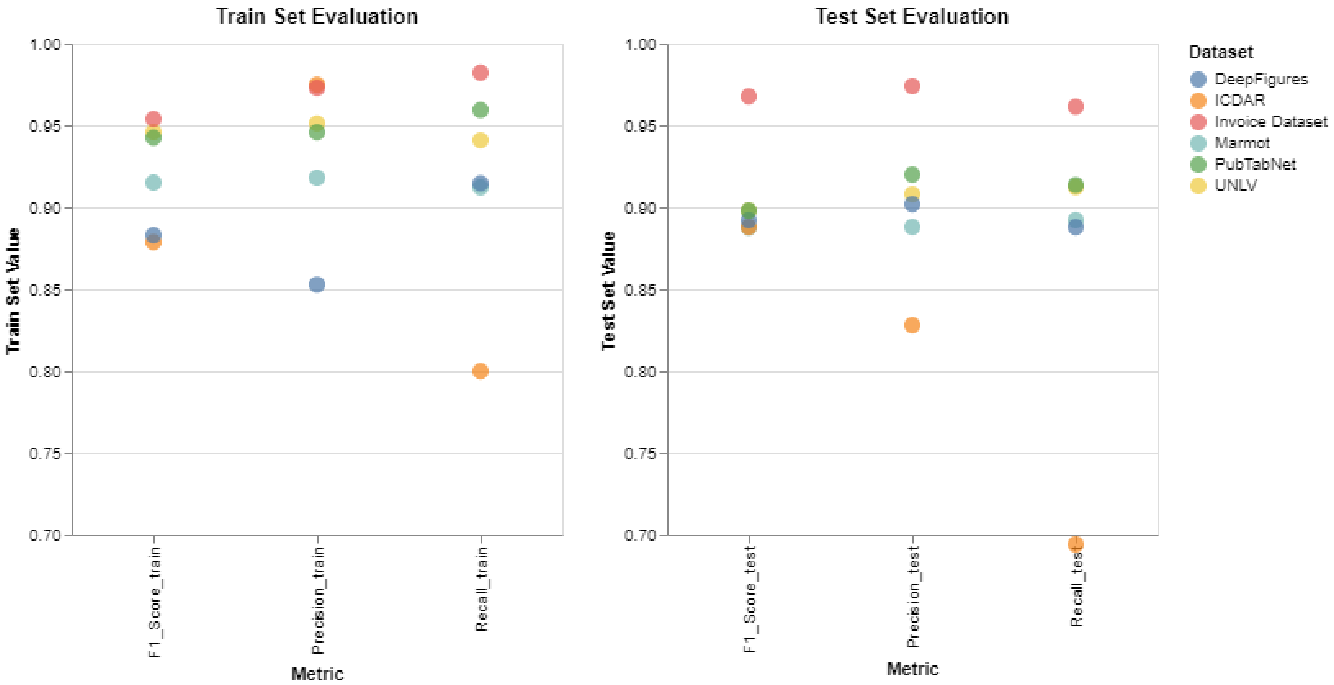


FIGURE 13. Table Detection Model Evaluation on Different Datasets

TABLE 5. Table Detection Model Evaluation on Different Datasets

Dataset	Recall		Precision		F1-Score	
2-7	train	test	train	test	train	test
PubTabNet	0.9044	0.8735	0.9593	0.8999	0.9311	0.8724
Augmented Dataset	0.8895	0.8164	0.9672	0.9234	0.9267	0.8913
***Invoice Dataset**	**0.9571**	**0.9476**	**0.9299**	**0.8865**	**0.9433**	**0.9139**

(WER%), and F1-Score for the PubTabNet and target invoice dataset also for more insight we evaluated the model on the synthetic data that was generated to train the model. The character error rate (CER%), focuses on evaluating the model in terms of the number of characters it can successfully classify and the word character rate (WER%), does the same in terms

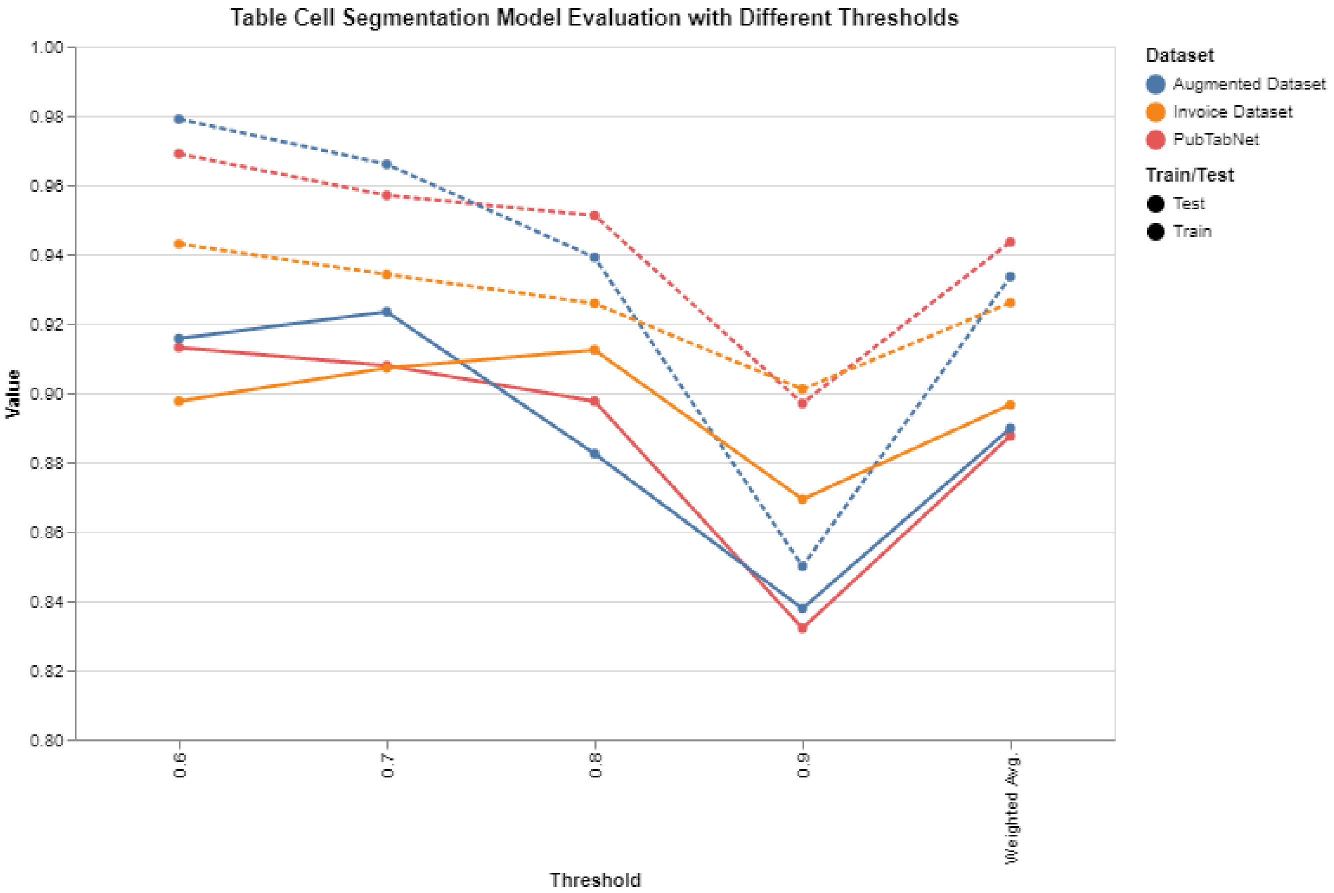


FIGURE 14. Table Cell Detection Model Evaluation with Different Thresholds

$$\cos(\varphi) = \begin{cases} \frac{\vec{X}\cdot\vec{Y}}{\|\vec{X}\|\|\vec{Y}\|} = \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2}\sqrt{\sum_{i=1}^{n} Y_i^2}} \\ \text{where } X_i \text{ denotes the predicted word vector } \{W_i, \ldots, W_n\} \\ \text{where } Y_i \text{ denotes the ground truth word vector } \{W_j, \ldots, W_g\} \\ \text{and } \vec{X}\cdot\vec{Y} \text{ is the dot product of vectors } \vec{X} \text{ and } \vec{Y} \end{cases} \tag{2}$$

of precise classified words. The f1-Score metric is calculated using the cosine similarity threshold ($\cos(\varphi) \cong 1.0$) which is listed in the table below:

D. ANALYZING THE EFFECTIVENESS OF INFORMATION EXTRACTION MODELS

To evaluate the information extraction block we used the object matching technique which is based on the similar concept of string matching. The calculation of the evaluation metric is based on matching all the attribute values of the JSON structure of the Ground truth and the generated sequence. The success case is when all of the entries on the ground truth and predicted JSON match completely which will be counted as (1) meaning it is correctly classified and in the other scenario even if the model could generate a fair amount of target output still it would be considered as a failed prediction resulting in (0) in the accuracy. In the below table, we have listed the Recall, Precision, and F1-Score of the model in the three datasets based on the accuracy calculation technique mentioned above.

Analyzing the impact of the experiment we observed that our proposed method managed to extract approximately 85% data in all of the different datasets. The percentage is solely based on the test set performance score. These depict the efficiency of the proposed model if we had trained the model on the target data the model would have struggled in comprehending novel test samples. Due to the different insight features provided by the synthesized data, the model could learn the jist of the task and learn to optimally extract information based on the textual and political features of the table. This also shows that our approach outperforms the segmentation and threshold-based techniques where the original structure of the table is missing in the extracted information which our approach manages to preserve.

$$CER = \begin{cases} \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \\ \text{where,} \\ \quad S \longrightarrow \text{Number of substitutions,} \\ \quad D \longrightarrow \text{Number of deletions,} \\ \quad I \longrightarrow \text{Number of insertions,} \\ \quad C \longrightarrow \text{Number of correct characters,} \\ \quad N \longrightarrow \text{Number of characters in the document} \end{cases} \tag{3}$$

$$WER = \begin{cases} \frac{R+A+E}{V} \\ \text{where, } R \longrightarrow \text{Number of words to be replaced,} \\ \text{where, } A \longrightarrow \text{Number of words to be added,} \\ \text{where, } E \longrightarrow \text{Number of words to be eliminated,} \\ \text{where, } V \longrightarrow \text{Total vocab size} \end{cases} \tag{4}$$

TABLE 6. Text Recognition Model Evaluation on Different Datasets

Dataset	CER (%)		WER (%)		F1-Score	
2-7	train	test	train	test	train	test
PubTabNet	2.6%	3.5%	10.6%	12.3%	0.9534	0.8936
Synthesized Dataset	4.3%	6.1%	12.5%	16.9%	0.8921	0.8498
***Invoice Dataset**	**2.7%**	**4.2%**	**11.5%**	**16.8%**	**0.9009**	**0.8623**

TABLE 7. Table Information Extraction Model Evaluation

Dataset	Recall		Precision		F1-Score	
2-7	train	test	train	test	train	test
PubTabNet	0.9122	0.8423	0.9021	0.8678	0.9511	0.9126
Augmented Dataset	0.8289	0.7597	0.8278	0.9123	0.8679	0.8201
***Invoice Dataset**	**0.8532**	**0.8487**	**0.8290**	**0.7234**	**0.8973**	**0.8564**

V. CONCLUSION

To summarize the contribution of this work, we started with 4,000 samples of Financial Image Datasets and started the information extraction task by adopting the transfer learning technique where we utilized publicly available datasets to train our information extraction pipeline. We introduced various augmentation techniques to generate synthesized data which led the model to comprehend insightful patterns and boosted the accuracy of the extracted information from the table images. The transfer learning and synthesized data generation approach played a vital role in the effectiveness of our work as the target data volume samples are not sufficient for the model to converge at an optimal performance edge. The current extraction approaches struggled to extract information from different layout tables and relied upon threshold values for grouping or clustering table cell data. Our proposed approach overcame the stated problem by utilizing positional, and textual features to generate a structured format of the extracted data using a sequence-to-sequence generation mechanism. We also demonstrated that our proposed approach manages to extract 85% of exact information despite different layouts.

References

[1] Rakesh Achanta and Trevor Hastie. "Telugu OCR framework using deep learning". In: *arXiv preprint arXiv:1509.05962* (2015).

[2] Md Rakibul Alama, Md Imran Hossaina, and Jannatul Ferdousa. "Analytical Device Model of Graphene Nanoribbon Field Effect Transistor". In: *Analytical Device Model of Graphene Nanoribbon Field Effect Transistor* 14.1 (2018), pp. 14–14.

[3] Adib Ahmed Anik. "An Experimental Approach to Identify Lifestyle Indicators in Diabetic Patients' Daily Episodic Notes Using UMLS and NLP With Reduced Ambiguity". MA thesis. Marquette University, 2023.

[4] Isuri Anuradha et al. "Deep learning based sinhala optical character recognition (ocr)". In: *2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer)*. IEEE. 2020, pp. 298–299.

[5] G. Bradski. "The OpenCV Library". In: *Dr. Dobb's Journal of Software Tools* (2000).

[6] Nicolas Carion et al. "End-to-end object detection with transformers". In: *European conference on computer vision*. Springer. 2020, pp. 213–229.

[7] E Roy Davies. *Computer vision: principles, algorithms, applications, learning*. Academic Press, 2017.

[8] Brijeshwar Dessai and Amit Patil. "A deep learning approach for optical character recognition of handwritten Devanagari script". In: *2019 2nd International conference on intelligent computing, instrumentation and control technologies (ICICICT)*. Vol. 1. IEEE. 2019, pp. 1160–1165.

[9] Srinivasa Rao Dhanikonda et al. "An efficient deep learning model with interrelated tagging prototype with segmentation for telugu optical character recognition". In: *Scientific Programming* 2022 (2022).

[10] Nadim Mahmud Dipu, Sifatul Alam Shohan, and KMA Salam. "Bangla optical character recognition (ocr) using deep learning based image classification algorithms". In: *2021 24th International Conference on Computer and Information Technology (ICCIT)*. IEEE. 2021, pp. 1–5.

[11] Y Du et al. "PP-OCR: A practical ultra lightweight OCR system. arXiv 2020". In: *arXiv preprint arXiv:2009.09941* ().

[12] Jing Fang et al. "A table detection method for multipage pdf documents via visual seperators and tabular structures". In: *2011 International Conference on Document Analysis and Recognition*. IEEE. 2011, pp. 779–783.

[13] Omar Farghaly and Priya Deshpande. "Texture-Based Classification to Overcome Uncertainty between COVID-19 and Viral Pneumonia Using Machine Learning and Deep Learning Techniques". In: *Diagnostics* 14.10 (2024), p. 1017.

[14] Filippo Galgani. *Legal Case Reports*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5ZS41. 2012.

[15] Basilios Gatos et al. "Automatic table detection in document images". In: *Pattern Recognition and Data Mining: Third International Conference on Advances in Pattern Recognition, ICAPR 2005, Bath, UK, August 22-25, 2005, Proceedings, Part I 3*. Springer. 2005, pp. 609–618.

[16] Wolfgang Gatterbauer et al. "Towards domain-independent information extraction from web tables". In: *Proceedings of the 16th international conference on World Wide Web*. 2007, pp. 71–80.

[17] Max Göbel et al. "ICDAR 2013 table competition". In: *2013 12th International Conference on Document Analysis and Recognition*. IEEE. 2013, pp. 1449–1453.

[18] Md Imran Hossain, Md Rakibul Alam, and Khondakar Abdullah Al Mamun. "A review of locomotion mechanism for wireless capsule endoscopy". In: *The International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering* 7.7 (2019), pp. 2321–5526.

[19] Jianying Hu et al. "Evaluating the performance of table processing algorithms". In: *International Journal on Document Analysis and Recognition* 4 (2002), pp. 140–153.

[20] Jianying Hu et al. "Table structure recognition and its evaluation". In: *Document Recognition and Retrieval VIII*. Vol. 4307. SPIE. 2000, pp. 44–55.

[21] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. "Funsd: A dataset for form understanding in noisy scanned documents". In: *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. Vol. 2. IEEE. 2019, pp. 1–6.

[22] Yuxiang Jiang, Haiwei Dong, and Abdulmotaleb El Saddik. "Baidu Meizu deep learning competition: Arithmetic operation recognition using end-to-end learning OCR technologies". In: *IEEE Access* 6 (2018), pp. 60128–60136.

[23] Srinidhi Karthikeyan et al. "An ocr post-correction approach using deep learning for processing medical reports". In: *IEEE Transactions on Circuits and Systems for Video Technology* 32.5 (2021), pp. 2574–2581.

[24] Thotreingam Kasar et al. "Learning to detect tables in scanned document images using line information". In: *2013 12th International Conference on Document Analysis and Recognition*. IEEE. 2013, pp. 1185–1189.

[25] Iftakhar Khandokar et al. "Towards Precision Diagnosis: Integrating Lexical Analysis and Deep Learning for Uncertainty Detection and Quantification in Clinical Reports". In: *2024 IEEE 37th International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE. 2024, pp. 267–272.

[26] Thomas Kieninger and Andreas Dengel. "The t-recs table recognition and analysis system". In: *Document Analysis Systems: Theory and Practice: Third IAPR Workshop, DAS'98 Nagano, Japan, November 4–6, 1998 Selected Papers 3*. Springer. 1999, pp. 255–270.

[27] Thomas G Kieninger. "Table structure recognition based on robust block segmentation". In: *Document Recognition V*. Vol. 3305. SPIE. 1998, pp. 22–32.

[28] Minghao Li et al. "DocBank: A benchmark dataset for document layout analysis". In: *arXiv preprint arXiv:2006.01038* (2020).

[29] Minghao Li et al. "Tablebank: Table benchmark for image-based table detection and recognition". In: *Proceedings of the Twelfth Language Resources and Evaluation Conference*. 2020, pp. 1918–1925.

[30] Minghao Li et al. "Trocr: Transformer-based optical character recognition with pre-trained models". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 11. 2023, pp. 13094–13102.

[31] KLND Liyanage. "Improving sinhala ocr using deep learning". PhD thesis. 2021.

[32] Binh Quang Long Mai, Tue Huu Huynh, and Anh Dong Doan. "A study about the reconstruction of remote, low resolution mobile captured text images for OCR". In: *2014 International Conference on Advanced Technologies for Communications (ATC 2014)*. IEEE. 2014, pp. 286–291.

[33] P. Malo et al. "Good debt or bad debt: Detecting semantic orientations in economic texts". In: *Journal of the Association for Information Science and Technology* 65 (2014).

[34] Sekhar Mandal et al. "A simple and effective table detection system from document images". In: *International Journal of Document Analysis and Recognition (IJDAR)* 8.2-3 (2006), pp. 172–182.

[35] Shamiha Binta Manir and Priya Deshpande. "Critical Risk Assessment, Diagnosis, and Survival Analysis of Breast Cancer". In: *Diagnostics* 14.10 (2024), p. 984.

[36] Shamiha Binta Manir and Priya Deshpande. "Student success analysis for minority students in higher education". In: *2023 IEEE International Conference on Big Data (BigData)*. IEEE. 2023, pp. 5292–5298.

[37] Shamiha Binta Manir, Mahima Karim, and Md Adnan Kiber. "Assessment of lung diseases from features extraction of breath sounds using digital signal processing methods". In: *2020 Emerging Technology in Computing, Communication and Electronics (ETCCE)*. IEEE. 2020, pp. 1–6.

[38] Ajoy Mondal, Peter Lipps, and CV Jawahar. "IIIT-AR-13K: A new dataset for graphical object detection in documents". In: *Document Analysis Systems: 14th IAPR International Workshop, DAS 2020, Wuhan, China, July 26–29, 2020, Proceedings 14*. Springer. 2020, pp. 216–230.

[39] Rayyan Najam and Safiullah Faizullah. "Analysis of recent deep learning techniques for Arabic handwritten-text OCR and Post-OCR correction". In: *Applied Sciences* 13.13 (2023), p. 7568.

[40] Marcin Namysl and Iuliu Konya. "Efficient, lexicon-free OCR using deep learning". In: *2019 international conference on document analysis and recognition (ICDAR)*. IEEE. 2019, pp. 295–301.

[41] Shubham Singh Paliwal et al. "Tablenet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images". In: *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE. 2019, pp. 128–133.

[42] Martha O Perez-Arriaga, Trilce Estrada, and Soraya Abad-Mota. "TAO: system for table detection and extraction from PDF documents". In: *The twenty-ninth international flairs conference*. 2016.

[43] David Pinto et al. "Table extraction using conditional random fields". In: *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. 2003, pp. 235–242.

[44] Devashish Prasad et al. "CascadeTabNet: An approach for end to end table detection and structure recognition from image-based documents". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2020, pp. 572–573.

[45] Shah Rukh Qasim, Hassan Mahmood, and Faisal Shafait. "Rethinking table recognition using graph neural networks". In: *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE. 2019, pp. 142–147.

[46] Masud Rabbani et al. "Towards developing a voice-activated self-monitoring application (VoiS) for adults with diabetes and hypertension". In: *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE. 2022, pp. 512–519.

[47] J-Y Ramel et al. "Detection, extraction and representation of tables". In: *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings*. IEEE. 2003, pp. 374–378.

[48] Sheikh Faisal Rashid et al. "Table recognition in heterogeneous documents using machine learning". In: *2017 14th IAPR International conference on document analysis and recognition (ICDAR)*. Vol. 1. IEEE. 2017, pp. 777–782.

[49] Stephen Rawls et al. "Combining deep learning and language modeling for segmentation-free OCR from raw pixels". In: *2017 1st international workshop on Arabic script analysis and recognition (ASAR)*. IEEE. 2017, pp. 119–123.

[50] Dennis Rösch et al. "VirtualSubstation: An IEC 61850 framework for a Containernet based virtual substation". In: *2022 57th International Universities Power Engineering Conference (UPEC)*. IEEE. 2022, pp. 1–6.

[51] Asif Shahab et al. "An open approach towards the benchmarking of table structure recognition systems". In: *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*. 2010, pp. 113–120.

[52] Arnab Sen Sharma et al. "A deep cnn model for student learning pedagogy detection data collection using ocr". In: *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*. IEEE. 2018, pp. 1–6.

[53] Noah Siegel et al. "Extracting scientific figures with distantly supervised neural networks". In: *Proceedings of the 18th ACM/IEEE on joint conference on digital libraries*. 2018, pp. 223–232.

[54] N Subramani et al. "A survey of deep learning approaches for ocr and document understanding. arXiv 2020". In: *arXiv preprint arXiv:2011.13534* ().

[55] Ashwin Tengli, Yiming Yang, and Nian Li Ma. "Learning table extraction from examples". In: *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*. 2004, pp. 987–993.

[56] Yalin Wang and Jianying Hu. "A machine learning based approach for table detection on the web". In: *Proceedings of the 11th international conference on World Wide Web*. 2002, pp. 242–250.

[57] Poonam A Wankhede and Sudhir W Mohod. "A different image content-based retrievals using OCR techniques". In: *2017 international conference of electronics, communication and aerospace technology (ICECA)*. Vol. 2. IEEE. 2017, pp. 155–161.

[58] Patrick Y Wu and Walter R Mebane Jr. "MARMOT: A deep learning framework for constructing multimodal representations for vision-and-language tasks". In: *Computational Communication Research* 4.1 (2022).

[59] Burcu Yildiz, Katharina Kaiser, and Silvia Miksch. "pdf2table: A method to extract table information from pdf files". In: *IICAI*. Vol. 2005. Citeseer. 2005, pp. 1773–1785.

[60] Richard Zanibbi, Dorothea Blostein, and James R Cordy. "A survey of table recognition: Models, observations, transformations, and inferences". In: *Document Analysis and Recognition* 7 (2004), pp. 1–16.

[61] Haochen Zhang, Dong Liu, and Zhiwei Xiong. "Cnn-based text image super-resolution tailored for ocr". In: *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE. 2017, pp. 1–4.

[62] Rui Zhang et al. "Icdar 2019 robust reading challenge on reading chinese text on signboard". In: *2019 international conference on document analysis and recognition (ICDAR)*. IEEE. 2019, pp. 1577–1581.

[63] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. "Image-based table recognition: data, model, and evaluation". In: *European conference on computer vision*. Springer. 2020, pp. 564–580.