International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-10 Issue-9, July 2021

Automatic Table Detection, Structure Recognition and Data Extraction from Document Images

Borra Vineetha, D. N. D. Harini, Ravi Yelesvarupu

Abstract: ***In the recent advancement, the extensive usage of electronic devices to photograph and upload documents, the requirement for extracting the information present in the unstructured document images is becoming progressively intense. The major obstacle to the objective is, these images often contain information in tabular form and extracting the data from table images presents a series of challenges due to the various layouts and encodings of the tables. It includes the accurate detection of the table present in an image and eventually recognizing the internal structure of the table and extracting the information from it. Although some progress has been made in table detection, obtaining the table contents is still a challenge since this involves more fine-grained table structure (rows and columns) recognition. The digitization of critical information has to be carried out automatically since there are millions of documents. Based on the motivation that AI-based solutions are automating many processors, this work comprises three different stages: First, the table detection using Faster R-CNN algorithm. Second, table internal structure recognition process using morphology operation and refine operation and last the table data extraction using contours algorithm. The dataset used in this work was taken from the UNLV dataset***

Keywords: Deep Learning, OCR, Scanned documents, Table detection, Structure recognition, Table data extraction

I. INTRODUCTION

Tables are widely used in many domains to present and communicate structured information to human readers since tables enable readers to search, compare and understand facts and draw conclusions rapidly. Hence, automatically detecting tables from documents and extracting the information contained in tables are of significant importance in the field of document recognition and analysis and have attracted a lot of research efforts in the past few decades. This paper focuses on the table detection, table internal structure recognition and data extraction in scanned documents. Scanned documents have become more and more popular. However, there is little or no structure information in scanned documents, which makes the information extraction and document understanding a challenging task.

As far as table detection is concerned, many researchers accomplished the task on scanned document images and web pages. Still, no work can handle all the tables well due to the diversity of table layouts and a variety of encodings. The existing methods for table recognition still have some challenging problems. For example, the image-based methods are prone to fail when directly carried out on scanned pages, while most of the rule-based methods are hardly able to recognize the tables without ruling lines or the tables that have complex layouts. More often, intersected vertical and horizontal lines are usually detected as faked tables in prior methods.

Meanwhile, in recent times, Deep learning techniques have improved the results of many computer vision tasks and information processing work to a great extent. In order to improve table detection performance and make up for the limitations of prior methods, this paper proposes a method of table detection based on deep learning techniques. This paper adopts the Faster R-CNN [1] algorithm to detect the tables and proposes the thresholding method to recognize the tabular internal structure. And then uses the adaptive thresholding method through OCR for data extraction from the tables.

In this paper the main focus is on the tabular data present in the images. Section 2 summarizes the relevant work on table detection, internal structure recognition and data extraction. Section 3 describes the framework of the proposed method in detail. Section 4 analyses the experimental results. The conclusion is discussed in Section 5.

II. RELATED WORK

In 1997, P. Pyreddy and W.B. Croft [2] was the first to propose a method for detecting tables using heuristics. The method was called TINTIN. To recognize table-like structures and their component fields they utilized the structural data present in the document. From the sequence of the words to the gap between them, the heuristics are categorized. Components are detected by visualization but not what they possibly mean. Doo soon Kim and Miao Fan [3] had developed a method for detecting the tabular structure from PDF documents by utilizing three classification algorithms namely Logistic Regression, Support Vector Machine, and Naïve Bayes.

Manuscript received on July 13, 2021.
Revised Manuscript received on July 17, 2021.
Manuscript published on July 30, 2021.
* Correspondence Author
Borra Vineetha*, Department of Computer Science and Engineering, GVP College of Engineering, Visakhapatnam (A.P.), India. E-mail: borravineetha97@gmail.com
D. N. D. Harini, Department of Computer Science and Engineering, GVP College of Engineering, Visakhapatnam (A.P.), India. Email: harinidhara@gvpce.ac.in
Ravi Yelesvarupu, CEO, Hallmark Solutions, Visakhapatnam (A.P.), India. Email: ravi@hallmark-solutions.com

In 2015, Florence Folake Babatunde et al. [4] presented a work to detect and extract table regions automatically by using Hidden Markov Model (HMM) from heterogeneous documents. The model was tested and trained with 526 self-generated tables. For the testing purpose, they have used the Viterbi algorithm. H.T.Ha et al. [5] introduced the OCR Miner system which is designed to extract the data present in the structured documents from scanned images.

Amir Riad et al. [6] has proposed the classification and information extraction methods for complex tabular structure in images. Their proposed document classification method uses document layout, and OCRed text information and, based on the combination of relative ROIs and RegEx the information extraction process is performed. Tuan Anh Tran et al. [7] implemented a method for detection of table regions by using a Random Rotation Bounding Box which is used for description and illustration of the table areas. Their system performed three different stages to recognize the table areas: classification of the non-text and text elements present in the input image, recognition of the un-ruling line tables and the ruling-line tables. Their algorithm has been evaluated on three publicly available datasets namely ICDAR2013 table competition, Diotek and UNLV.

Manabu Ohta et al. [8] proposed a cell detection method for table-structure recognition and automatic generation of graphs from the tables. They detected the cells by evaluating indirect rules to identify the table structure. They have used the ICDAR2013 table dataset for implementing their method. Rastan et al. [9] recently developed the TEXUS framework (2019), which identifies the structure of a table in an unconventional layout manner. This work is limited to born-digital PDFs only. In [10], the authors chose to form twenty rules to detect a table. These rules are applied to frames of web pages. The interpretation phase follows locating the tables. They again use visual rules to extract information from the tables. Elvis Koci et al. [11] proposed a framework that will be able to automatically infer the structure and extract the information from the documents in canonical form.

III. PROPOSED METHODOLOGY

The proposed method consists of three major modules: Table Detection, Table Structure Recognition and Table Data Extraction present in the table. This section will discuss each procedure in detail. The system architecture of the proposed work is shown in figure 1.

A. SYSTEM MODEL

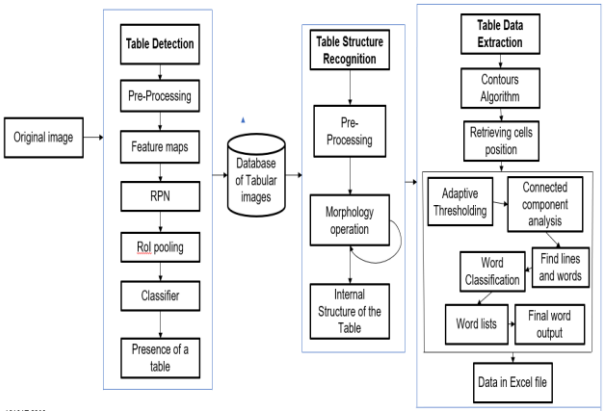


Figure 1. System Architecture

In the above figure 1 there are three stages in the proposed system which are as follows: first, table detection which includes pre-processing and faster R-CNN algorithm; second, the identification of the tabular structure including rows and columns using morphology operations; third, extracting the data present in the table from the document image by applying OCR.

B. Table Detection

In this, we will identify the presence and location of the table present in the image from the scanned document. For this purpose, a deep learning framework called Faster R-CNN is utilized for the detection process. The Faster R-CNN model consists of a detector which is Fast R-CNN and a Region Proposer which is RPN. This is an object detection algorithm. It is the improved version of Fast R-CNN. R-CNN and Fast R-CNN need an external region proposer called selective search whereas Faster R-CNN doesn't need any external Region proposers because it has an RPN model.

The architecture of table detection using Faster R-CNN is shown in the below figure 2. The very first step of the FRCNN algorithm is the Region Proposal Network. The task of RPN is to find out those areas in an image where there is a possibility of an object present. It means the area in the picture where the object can be possibly found. The process of RPN starts with the use of Anchor boxes. These are the set of pre-defined bounding boxes of some height and weight. After generating the anchor boxes the next step is to find out IoU. Under anchor box, with the higher IoU i.e., IoU>50% will be labelled as foreground labels otherwise labelled as background class. The output from RPN will be a feature map of those with anchor boxes labelled as foreground boxes.

Algorithm: Region Proposal Network Layer
Input:

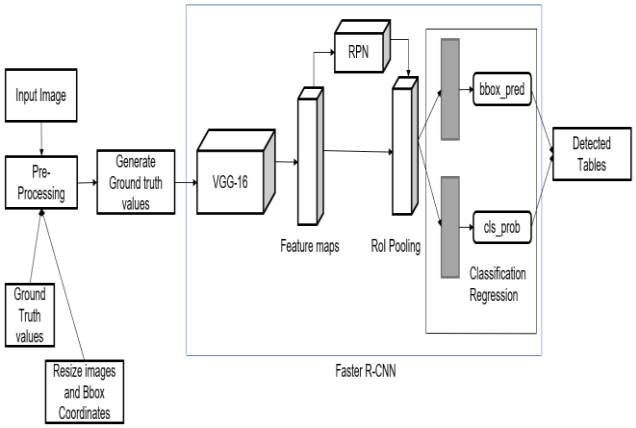
- Pre-processed scanned document image(S)
- Pre-processed template Document image(T)
- Annotation for template document (A_T)

Procedure:

- Image width- W_T ←Width(T)
- Image height- $H_T \leftarrow 1.4 * W_T$
- For each field in A_T do
  Obtain rectangular area of the field
  The area of the rectangle in all the directions should get increased slightly Crop the new area in the document image
  Find the area of the field in the given image
- End for

Output: region proposals for all the fields in the document images

Figure 2. Architecture of Table detection



Next is the RoI Layer; the data which the RoI layer receives is the output of the RPN layer. The input to this RoI is different sizes of feature maps. The task of RoI is to reduce all the feature maps to the same size. RoI pooling layer will produce fixed size feature maps from the different sized regions using max pool on it. The size of max pool is 7*7 and number of channels are 512. RoI pooling accepts two inputs. One is RoI which RPN creates and the other is feature maps from VGG. The figure 3 is the output image after performing the table detection process.

Algorithm: Area Selection

Input: Input scanned document Template Document Region proposals in input document Input document and template document dimensions

Procedure:

- for every field in region proposal do
- find the texts which are in common in the proposal area from the input image, and the corresponding area from the template image. ▪
- If same>0 then'
- Find the text which is nearer to the original value for the field from the template image
- Get a vector from the center of the approx. text and the original value.
- With the help of dimensions of the template image, normalize the vector. Using the dimensions of the input image, de-normalize the vector.
- Predict the center using the vector in the given image for which the value of the field is present.
- Acquire the rectangle in the given image by applying scaling appropriately.
- else acquire the rectangle at the center of the closest area by considering the value of the field in template image
- end if
- end for

Output: Final Bboxes for all the fields in the scanned document given as input.

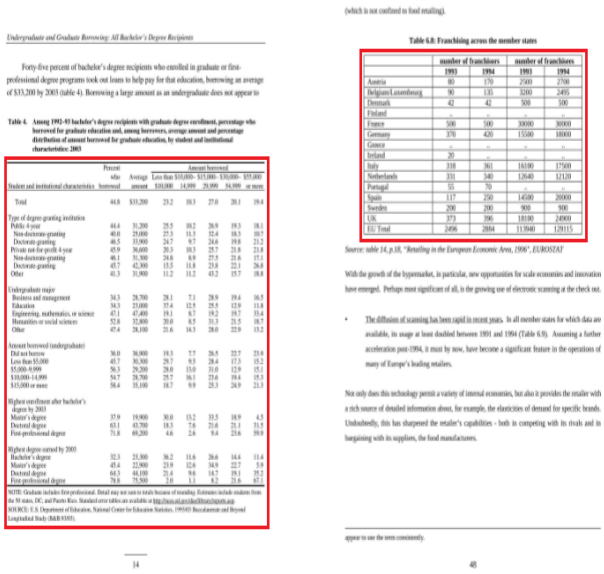


Figure 3. Detection of Tables

C. Table Structure Recognition

After the table presence and location is successfully detected from the image, the next step is to recognize the internal structure of the table. For recognizing the tabular structure, a morphology method is utilized. It is one of the most widely used techniques for depicting the shape of the region in the image. The overall process for Table Structure Recognition is shown in Figure 4. Here we apply dual morphological operations i.e., erosion and dilation with dynamic structuring elements the rows and columns of the table. In this work, the horizontal structuring element and vertical structuring element have been selected to detect the horizontal and vertical lines of the table.

Algorithm:

Input: Database of Tabular images

Procedure:

- List all data-lines within the table as x1, x2....xn, and traverse possible line pairs.

  Xij= {Ti, Tj}, 0<i<j<=n.
- If Ti and Tj are not vertically aligned, go to step 7
- If they are vertically aligned, record all their alignment types in Pij.
- Check whether Ti is already included in the existing column and check if intersection of PC and Pij is not empty.
- if yes, add Tj into C. And set the intersection of PC and Pij as new PC.
- if no, create a new column candidate Cnew= {Ti, Tj}. And set PCnew=Pij
- Continue with the next pair.

Output: Internal Structure of a Table

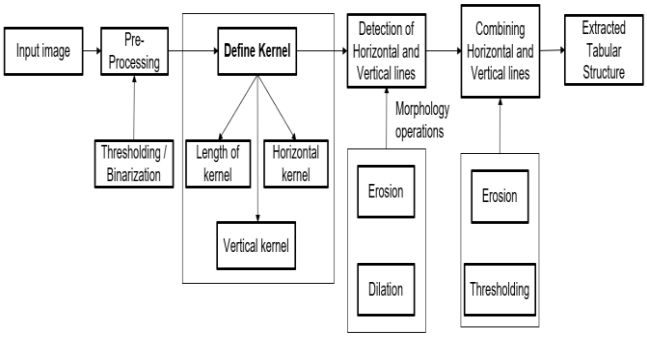


Figure 4: Architecture of Table Structure Recognition

The images from figures 5(a) to 5(e) are obtained after applying thresholding method and morphology operations on the original image. These images display the results of the proposed method for Table Structure Recognition where figure 5(a) is the original image taken from the dataset, figure 5(b) is the image obtained after applying the Otsu thresholding method, figure 5(c) and figure 5(d) are the images obtained after detecting the vertical lines and horizontal lines using kernel, and figure 5(e) is the final image obtained after recognizing the internal structure of the table

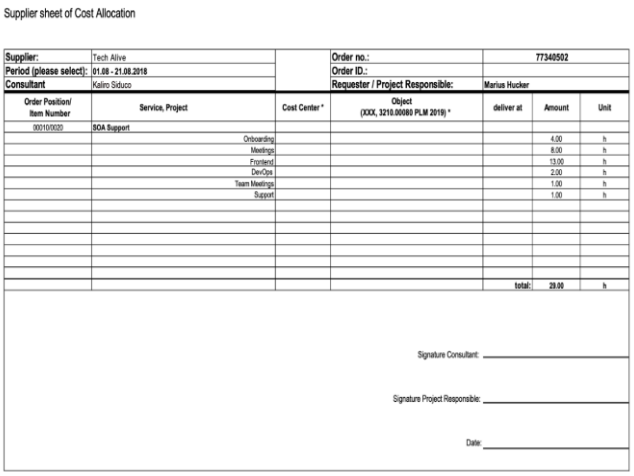


Figure 5(a)

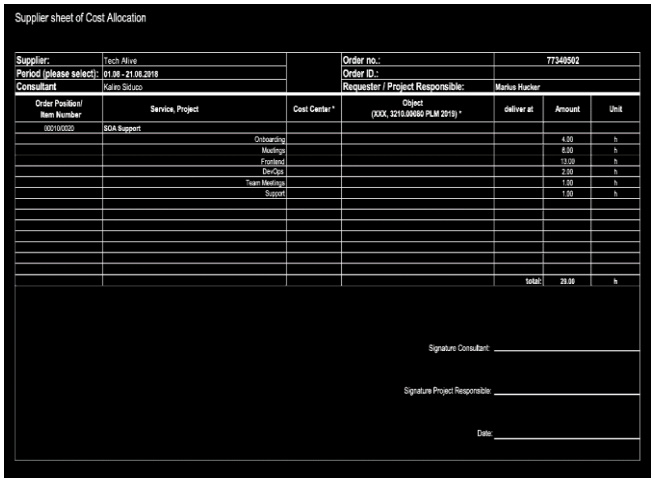


Figure 5(b)

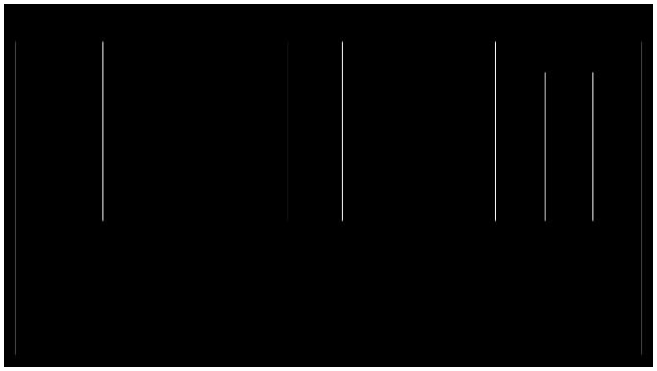


Figure 5(c)

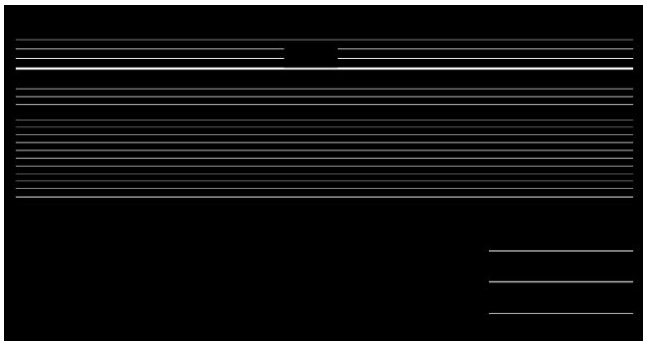


Figure 5(d)

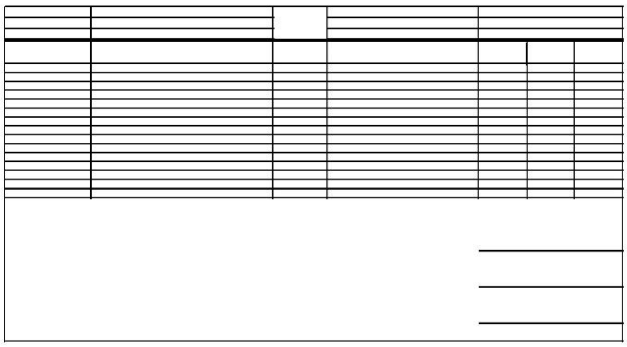


Figure 5(e)

D. Table Data Extraction

The architecture of Data Extraction is shown in Figure 6. For extracting the data, there are two different steps namely retrieving the cells position and extracting the values.

To retrieve the cells position; the height of each cell is to be retrieved and calculate the mean from the heights. Next retrieve the position, height and width of each contour and store it in the box list. Then draw the rectangles for all the boxes and plot the image. To get the right location of the cell within the table, the particular location of the cell should be known i.e.; in which row and column it is there.

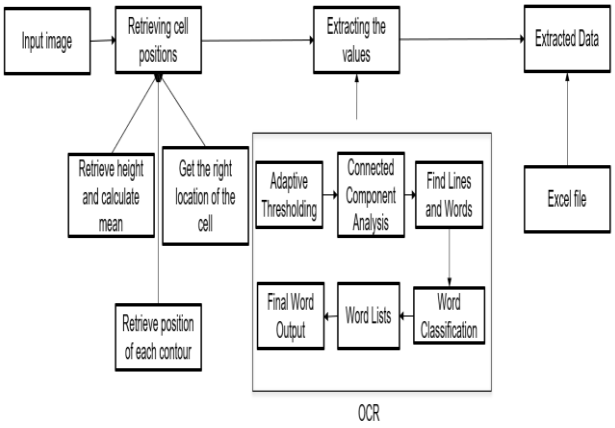


Figure 6: Architecture of Table Data Extraction

Here the box is in the same row if its own (height + mean/2) does not vary more than its original height. But if the height difference is more than the present (height + mean/2), a new row starts. Columns are relevantly arranged from left to right. To know the total number of columns a table contains, the maximum number of columns (meaning cells) should be calculated. After this, the midpoint of each column in a list is to be stored, create an array and sort the values. The proper sequence will be stored in the list final boxes.

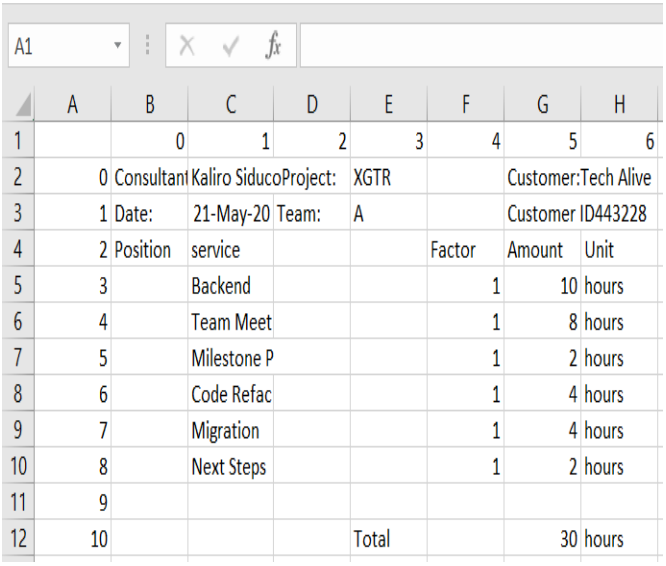
Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in

Algorithm:
Input: Database of Tabular images
Procedure:

- assign a confidence weight to every word
  - initialize the confidence for all words to 0
- check the previous word and immediate next word in the correct text using the location to see if it is a match
  - if match
    increase word confidence and previous word by 25
    store the location in the database
    move to next word (again from step 1)
  - if not a match
    decrease word confidence by 25
    if current confidence of word=5(match in dict)
    then
    decrease confidence of previous word by 25
    continue to next step
- begin to look for the word in a nearby location
  search for exact matches after the word for m words (at present m=20)
  - if match
    increase confidence by 25
    move to the next word
  - if not match
    search m words before the previous word
  - if match
    increase confidence by 1
    decrease confidence of the previous word by 4
    move to the next word
    else
    go to next step
- No match for words immediately after or in the neighbouring x words:
  Search for matches like in previous step for neighboring words
  Going to search for words that have an edit distance of Y.
  Iterate starting with 1 edit distance until Y
  Limit the size of Y to avoid false positives
  Set a maximum Y of 3
  Edit distance checked is less than Y
  - If matching candidate with Y distance is found
    Increase confidence by 1
    Move on to the next word
  - If not
    Increase the edit distance (Y++)
    Continue loop
- If the word is not found, search for the previous X words by allowing an edit distance until Y
- After all these steps if a match is not found
  Give a confidence of –1
  Store the location of the previous word plus
  Apply OCR

Output: Extraction of data from a table

Extracting the values; make use of list final boxes by taking every image-based box and prepare it for Optical Character Recognition by dilating and eroding it. Now let Pytesseract recognize the containing strings. The loop runs over every cell and stores the value in the outer list. The last step is the conversion of the list to a data frame and storing it into an excel-file. The table should now be in an excel-file and can be used for Natural Language Processing, for further analysis via statistics or just for editing it. Figure 7 is the image after data extraction is performed. And that image is in the form of an excel-file.



	A	B	C	D	E	F	G	H
1		0	1	2	3	4	5	6
2	0	Consultant	Kaliro Siduco	Project:	XGTR		Customer:Tech Alive	
3	1	Date:	21-May-20	Team:	A		Customer ID	443228
4	2	Position	service			Factor	Amount	Unit
5	3		Backend			1	10	hours
6	4		Team Meet			1	8	hours
7	5		Milestone P			1	2	hours
8	6		Code Refac			1	4	hours
9	7		Migration			1	4	hours
10	8		Next Steps			1	2	hours
11	9							
12	10				Total		30	hours

Figure 7. Data in excel file

IV. RESULTS AND ANALYSIS

The experimental results are shown below. It shows some detected table areas, internal structure of the table and the extracted data present in the table. The table will be in an excel-file and can be used for Natural Language Processing, for further analysis via statistics or just for editing it.

A. Evaluation Metrics

Various performance measures have been used for evaluation of table recognition system. These measures include Mean IoU, precision, recall and F1_score.

a) Mean IoU: The IoU is the ratio of the area of intersection and area of union of the ground truth bounding box and predicted bounding box.

$$\frac{Area\ of\ intersection\ of\ predicted\ and\ GT\ bounding\ box}{Area\ of\ union\ of\ predicted\ and\ GT\ bounding\ box}$$

b) Precision: This measure has been used for evaluating the overall performance of table detection method. It finds the percentage of detected tables that belong to table regions of ground truth document image.

$$\frac{Area\ of\ GT\ regions\ in\ detected\ regions}{Area\ of\ all\ detected\ table\ regions}$$

c) Recall: It is evaluated by finding the percentage of ground truth table regions that were marked as detected table regions.

$$\frac{Area\ of\ GT\ regions\ in\ detected\ regions}{Area\ of\ all\ GT\ table\ regions}$$

d) F1 Score: It is defined as the harmonic mean of the model's precision and recall. It considers both recall and precision to compute the accuracy of the methodology

$$2 \times \frac{Precision * Recall}{Precision + Recall}$$

B. Result and Discussions

Performance comparison between Florence Folake Babatunde et al. [4], Ranajit Saha et al. [12], Maik Rhiele et al. [11] and the proposed method is shown in table 1. The results exhibit that the proposed system has better performance by achieving Mean IoU as 88.7, Precision as 97.3, Recall as 92.4 and F1_score as 90.8. For classification, the mean IoU evaluation metric is used for predicting class labels where the model outputs a single label that is either present or absent. This type of classification makes computing accuracy straightforward. Any algorithm that provides predicting bounding boxes as output can be evaluated using IoU for better results. The work is designed and implemented in the windows operating system having 8GB ram, 2GB Graphic card and 1TB HDD.

Table1: Results Table

Evaluation Metric	Florence Folake Babatunde et al. [4]	Ranajit Saha et al. [12]	Maik Rhiele et al. [11]	Proposed Work
Mean IoU	81.8	-	-	88.7
Precision	96.8	92.8	65.0	97.3
Recall	91.7	92.1	63.0	92.4
F1_Measure	88.8	91.5	61.0	90.8

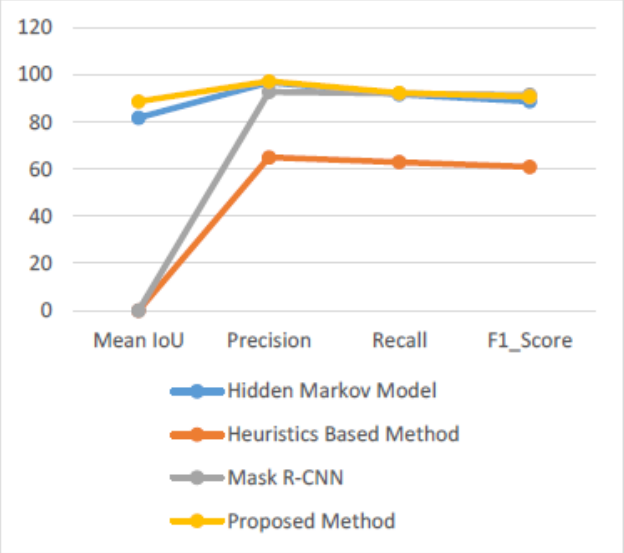


Figure 8: Comparison of Performance Measurements

V. CONCLUSION

In this work, a method for extracting data from tables has been developed. In the process of table detection, feature map extraction and classification are done using the Faster R-CNN algorithm. The refine morphological operations erosion and dilation with dynamic structuring elements employed to identify the internal structure of the table. The data from the recognized table is then extracted using the contours algorithm and OCR. In this research, the UNLV dataset is used to extract information from tables. Furthermore, the results of comparative experiments have demonstrated that the proposed method outperforms the existing system and the other approaches with an accuracy of 88%. Further research will focus on improving the system architecture and recognition accuracy for more imbalanced and multiple layout tables.

REFERENCES

1. S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards Real-time Object Detection with Region Proposal Networks," in Advances in Neural Information Processing Systems, pp. 91–99, v3, 2016.
2. P. Pyreddy and W. B. Croft. "Tintin: A System for Retrieval in Text Tables", in Proceedings of the Second ACM International Conference on Digital Libraries, pp.193-200, 1997.

3. Miao Fan and Doo Soon Kim, "Detecting Table Region in PDF Documents Using Distant Supervision", corpus ID: 14348894, Version 6, 2015.
4. Florence Folake Babatunde, Bolanle Adefowoke Ojokoh, Samuel Adebayo Oluwadare, "Automatic Table Recognition and Extraction from Heterogeneous Documents", Journal of Computer and Communications 03, pp 100-110, 2015
5. H. T. Ha, M. Medved, Z. Neverilova, and A. Horak, "Recognition of OCR Invoice Metadata Block Types", 21st International Conference, TSD 2018, Proceedings, Pp. 304-312.
6. Amir Riad, Christian Sporer, Syed Saqib Bukhari, Andreas Dengel, "Classification and Information Extraction for Complex and Nested Tabular Structures in Images", 14th IAPR International Conference on Document Analysis and Recognition, 2017.
7. Thong Huynh-Van, Khuong Nguyen-An, Trinh Le Ba Khanh, Hyung-Jeong Yang, Tuan Anh Tran, Soo-Hyung Kim, "Learning to Detect Tables in Document Images using Line and Text Information", ICMLSC, Pp 151-155, 2018.
8. Manabu Ohta, Ryoya Yamada, Teruhito Kanazawa, Atsuhiro Takasu, "A Cell-detection-based Table-structure Recognition Method", DocEng'19: Proceedings of the ACM Symposium on Document Engineering, Pp 1-4, 2019.
9. R. Rastan, H.-Y. Paik, and J. Shepherd, "Texas. A Unified Framework for Extracting and Understanding Tables in PDF Documents," Information Processing & Management, Vol.56, no. 3. Pp. 895-918, 2019.
10. J. W. Son, H. J. Song, J. A. Lee, S. J. Lee, S. B. Park, and S. Y. Park, "Discriminating Meaningful Web Tables from Decorative Tables using a Composite Kernel," in Proceedings - 2008 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2008, pp. 368–371, 2008.
11. Elvis Koci, Maik Thiele, Oscar Romero, and Wolfgang Lelner, "Table Identification and Reconstruction in Spreadsheets", International Conference on Advanced Information Systems Engineering,2017: Proceedings, Springer, pp 527-541, 2017.
12. Ranajit Saha, Ajoy Mondal, C V Jawahar, "Graphical Object Detection in Document Images", Institute of Electrical and Electronics Engineers (IEEE), Conferences, 2019.
13. Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, Zhoujun Li, "Table Bank: Table Benchmark for Image-based Table Detection and Recognition", Proceedings of the 12th conference on Language Resources and Evaluation (IREC 2020), pages 1918-1925.

AUTHORS PROFILE

Borra Vineetha, is currently pursuing her MTech in the dept of Computer Science and Engineering at GVP College of Engineering(A), Visakhapatnam. She had a degree of BTech in Computer Science and Engineering from Dadi Institute of Engineering and Technology, Visakhapatnam. Her main area of interests includes Machine Learning and Deep Learning.

D. N. D. Harini, completed her PhD in CSE, and working as Associate Professor in CSE Dept at GVP College of Engineering (A), Visakhapatnam. Her research areas include Computer Vision, Machine Learning and Deep learning. She had published good number of publications in reputed journals.

Ravi Yelesvarupu, completed his Masters in Mathematics from Osmania University, Hyderabad. He had more than 15 years of US experience working for AT&T. Founder of Hallmark solutions, Visakhapatnam. It is an India based startup focusing on cloud, Open source, ML/ AI based solutions in the areas of Data analytics, OCR, Workflow Automation.