# Table Recognition and Content Extraction

*Submitted by*

Hire Vikram Umaji (2015CS10227)

B.TECH PROJECT REPORT

&

Amar Agnihotri (2012CS50277 )

M.TECH PROJECT REPORT PART 1

*Under the guidance of*

Prof. M. Balakrishnan & Prof. Volker Sorge

*in partial fulfilment of the requirements*
*for the award of the degree of*

Bachelor of Technology


Department Of Computer Science and Engineering
INDIAN INSTITUTE OF TECHNOLOGY DELHI

# ACKNOWLEDGEMENTS

First and foremost, we would like to thank **Prof. M Balakrishnan** and **Prof. Volker Sorge** (University of Birmingham) for giving us the opportunity to work on this amazing project and providing their guidance throughout the project.

A very special thanks to Himanshu Garg and Neha Jadhav for their guidance and discussions throughout the project.

We would also like to thank our family and friends for their moral support and continuous encouragement.

# Contents

# Chapter 1

## 1.1 INTRODUCTION

This document provides an analysis of Table recognition, content extraction and their placement in output HTML which is a sub-task of a currently ongoing project RAVI (Reading Assistant for Visually Impaired) which focuses on making digital textbooks accessible to readers by accessing, analysing and interpreting the data in some form for visually impaired. One of the work under this is to detect tables and extract its information as tabular cell structures. The motivation behind this work comes from making tables accessible to Visually Impaired Peoples.

## 1.2 Objective

The objective of our project is to make tables accessible in a document. The table coming inside a document is classified into 2 areas in this project. The table structure comes in the background as image glyph and table data in the foreground as span in the RAVI project. Our task is to identify the table structure in the background and match every table cell with correct data in the foreground and finally place the table in the HTML output.

## 1.3 Prior Work

Prior work on table accessibility was done previously in the RAVI project. But they had done work only on Bordered Tables, Borderless tables were undetected. In bordered table work the analysis we did used about 20 documents of samples folder, which consisted of about total 50 tables. Their work indicated the following results:

| Docs | Tables | Correctly Detected | Accuracy |
|---|---|---|---|
| 20 | 50 | 21 | 42% |

As we can see, many errors were occurring.

We have utilised the matchSpans() function from prior works to detect and match data of table in the foreground to the table cell structure in the background. The table structure was detected by taking image glyphs from backgroundAnalysis() function as input.

The classification of table was done into two types as stated above i.e. Bordered Tables and Borderless Tables. The bordered tables were tables which were coming with complete table cell line structure. Similarly borderless tables were tables with partial/Incomplete borders of cell structure. The examples are shown in Figures below:

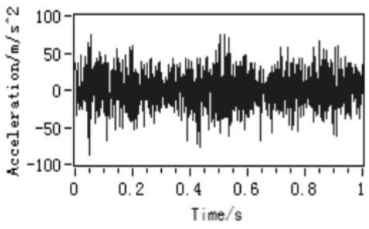
simulation signal after de-noising, we can find 100Hz and 10Hz frequency band. And from the envelope spectrum, we can identify the modulation frequency of 10 Hz. Through a simple test, we can confirm that the wavelet-mathematical morphology de-noising can restore the original signal and verify the capability envelope spectrum in the extraction of modulated signal.

## V ENGINEERING APPLICATION

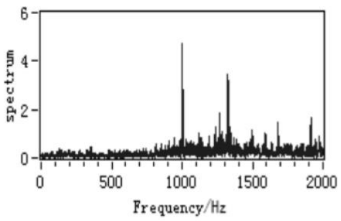
TABLE I BASIC KINEMATIC DATA OF THE WIND TURBINE UNDER CONSIDERATION

| | Frequency / Hz | |
|---|---|---|
| **Rotor shaft** | **9.9 rpm** | **17.3 rpm** |
| Spindle shaft | 0.165 | 0.288 |
| Sun shaft | 0.95 | 1.65 |
| Intermediate shaft | 3.71 | 6.48 |
| High speed shaft | 17.2 | 30.06 |
| Line star-level | 17.16 | 30 |
| Intermediate-level | 96.46 | 168.48 |
| High- level | 430 | 751.5 |

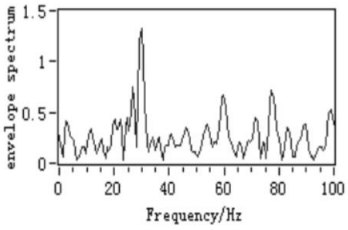
We use the gearbox shown in Fig. 1 to monitor wind turbines in a wind farm. The low speed shaft which drives from the planetary gear can be power split used for split of power and the internal gear is used reasonably at the same time. The last two stages are parallel shaft cylindrical gears, and can assign speed ratio reasonably, improve the transmission efficiency. The following results are shown in Fig. 3.



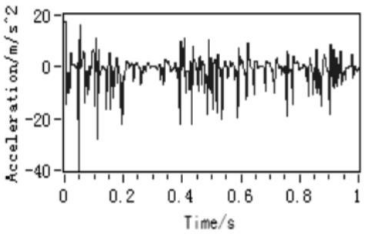

(a)Original signal of Gearbox in time domain



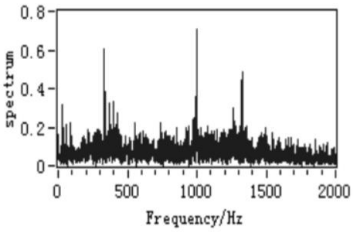

(b)Original signal of Gearbox in frequency spectrum



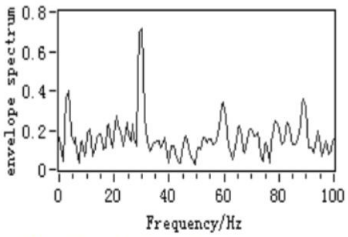

(c)Original signal of Gearbox in envelope spectrum




(d)De-noising signal of Gearbox in time domain




(e)De-noising signal of Gearbox in frequency spectrum




(f)De-noising signal of Gearbox in envelope spectrum

Figure 3. Fault detection of gearbox for wind turbine in a wind farm

From the original frequency spectrum of the vibration signal of the gearbox, many peaks can be found in high frequency domains while the useful frequency is not easy to be found in low frequency domains. After the de-noising using





Fig. Document page with Bordered Table

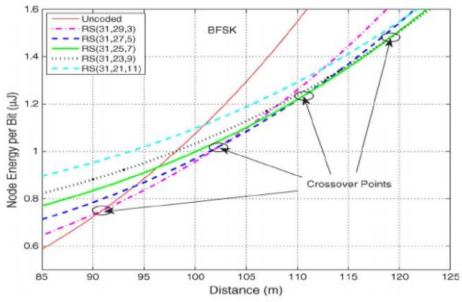



Fig. 4. Energy-optimal RS codes.

TABLE II
TYPICAL BERs FOR CLASSES OF APPLICATIONS [19]

| Application class | BER |
|---|---|
| Speech telephony | $10^{-4}$ |
| Voice band data | $10^{-6}$ |
| E-mail, Electronics newspaper | $10^{-6}$ |
| Internet access | $10^{-6}$ |
| Video telephony | $10^{-7}$ |
| High speed computing | $10^{-7}$ |

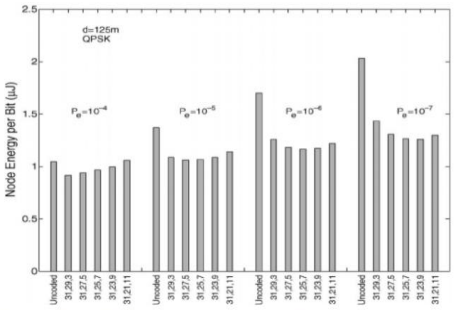



Fig. 5. Effect of BER on energy consumption of RS codes.

With the change in the surrounding environment from open air to urban area, the channel behavior changes. The choice of energy-optimal scheme also changes with the change in surroundings. Path loss exponent $n$ models the change in channel conditions. In general, the value of path loss exponent typically varies from two to six. In Table III, the values of path loss exponents for various environments are listed.

From (5), $P_{\text{sig}}$ is proportional to $d^n$. As $n$ decreases, for the same distance, signal transmit power decreases exponentially (Fig. 6). At low value of $n$, computation energy dominates.

TABLE III
PATH LOSS EXPONENTS FOR DIFFERENT ENVIRONMENTS [20]

| Environment | Path loss exponent |
|---|---|
| Free space | 2 |
| Urban area cellular radio | 2.7 to 3.5 |
| Shadowed urban cellular radio | 3 to 5 |
| In building line-of-sight | 1.6 to 1.8 |
| Obstructed in building | 4 to 6 |
| Obstructed in factories | 2 to 3 |

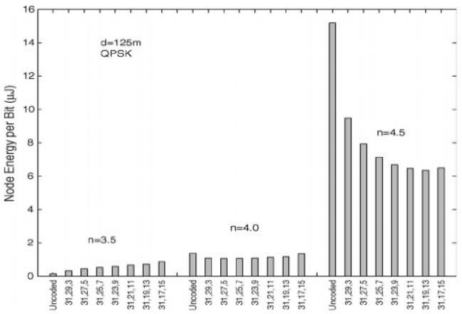



Fig. 6. Energy consumption of various RS codes for different path loss exponents.

This implies that the coding gain is not sufficient enough to overcome the computation overhead. As evident from Fig. 6, at a distance of 125 m for $n = 3.5$, sending the uncoded data is beneficial. However, when $n = 4$, sending the encoded data is beneficial.

### C. Node Energy Variations With Modulation and ECC Parameters

*1) Constellation Size:* Different commercial sensor nodes may use different modulation schemes, depending on the operating frequency and communication standards used. Our framework takes into account different modulation schemes and their constellation sizes. Node energy consumption depends on the constellation size of a modulation scheme in two ways. On increasing the constellation size of the uncoded signal, the required SNR per bit decreases. On the other hand, for the coded signal for an increased constellation size, a large SNR is required to maintain the desired BER. This, in turn, affects the coding gain of ECC. Therefore, on changing the constellation size, the node energy with the same ECC changes. Fig. 7 shows the normalized node energy with respect to BPSK at different distances for RS (31, 29, 3) with various PSK constellation sizes. The node energy is optimal for a particular constellation size. On changing the internode distance from 30 to 60 m, the optimal ECC changes from 8-PSK to QPSK.

*2) ECC Parameters:* The user can choose from many available ECCs and their variants. Different RS codes are designed by varying code word length $N$ and error-correcting



Fig. Document page with Borderless Table

**Table 1. Summary of Notation**

| Symbol | Meaning |
|---|---|
| $k$ | The number of loci that influence a quantitative trait |
| $\ell$ | The ploidy of the individuals being considered |
| $M$ | An individual's population membership; takes values A and B |
| $L_{ij}$ | An individual's allelic type at the $j$th allele at the $i$th locus; takes values 0 and 1 |
| $p_i$ | The frequency of the 1 allele at locus $i$ in population A (Eq. 1) |
| $q_i$ | The frequency of the 1 allele at locus $i$ in population B (Eq. 1) |
| $\overline{p}$ | The mean frequency of the 1 allele across loci in population A (Eq. 2) |
| $\overline{q}$ | The mean frequency of the 1 allele across loci in population B (Eq. 2) |
| $s_p^2$ | The variance across loci in the frequency of the 1 allele in population A (Eq. 3) |
| $s_q^2$ | The variance across loci in the frequency of the 1 allele in population B (Eq. 3) |
| $V_{ij}$ | An indicator for whether an individual's $j$th allele at the $i$th locus is a + allele (Eq. 4) |
| $T$ | An individual's value for a quantitative trait (Eq. 4) |
| $X_i$ | An indicator for whether the 0 or the 1 allele is also the + allele at locus $i$ (Eq. 5) |
| $S$ | The number of 1 alleles an individual carries (Eq. 6) |
| $\delta_i$ | The difference between populations in the frequency of the 1 allele at locus $i$ (Eq. 8) |
| $\overline{\delta}$ | The mean allele-frequency difference between populations, or $\overline{q} - \overline{p}$ (Eq. 9) |
| $\overline{\delta^2}$ | The mean squared difference between populations in the frequency of the 1 allele (Eq. 10) |
| $F_{ST}^k$ | The ratio of the mean (across $k$ loci) within-population variance in an allelic indicator variable to the mean (across $k$ loci) total variance in an allelic indicator variable (Eq. 14) |
| $D_L^2$ | A function of $F_{ST}^k$ that bears the same relationship to $F_{ST}^k$ as the square of Cohen's $d$ does to $r^2$ (Eq. 15) |
| $F_{ST(\ell)}^k$ | A generalization of $F_{ST}^k$ for the sum of $\ell$ independent allelic indicator variables (Eq. 16) |
| $D_{L(\ell)}^2$ | A function of $F_{ST(\ell)}^k$ that bears the same relationship to $F_{ST(\ell)}^k$ as the square of Cohen's $d$ does to $r^2$ (Eq. 17) |
| $U_i$ | A transformation of the $X_i$: if $X_i = 1$, then $U_i = 1$; if $X_i = 0$, then $U_i = -1$ (Eq. 18) |
| $D_T$ | The standardized difference between populations A and B on the trait (Eqs. 25, 34) |
| $\rho_T^2$ | The proportion of the total variance in the trait attributable to between-population difference on the trait (Eqs. 26, 27, 41) |
| $Q_{ST}$ | A quantitative-trait analogue of $F_{ST}$: if $\ell = 1$ (haploid organisms), then $Q_{ST} = \rho_T^2$ (Eq. 28) |
| $W_S$ | A quantity that equals 1 if an individual is classified into the wrong population on the basis of its value of $S$, and that equals 0 otherwise (Eq. 55) |
| $W_T$ | A quantity that equals 1 if an individual is classified into the wrong population on the basis of its value of $T$, and that equals 0 otherwise (Eq. 56) |

haploids. Here, we extend our earlier model to allow arbitrary allele-frequency distributions and arbitrary ploidy. Our results provide another way of establishing the result that between-group differentiation on a neutral trait mirrors between-group genetic differentiation at a neutral locus, one that makes minimal evolutionary assumptions. In Section 2, we describe our extended model. In Section 3, we define several measurements of between-group genetic differentiation. In Sections 4 and 5, we describe properties of two statistics that summarize the degree of difference between two populations on a quantitative trait. In Section 6, we introduce two simplifying assumptions that allow us to analyze the problem of inferring an individual's population of origin using either genetic or phenotypic information. Finally, we discuss the results with respect to the interpretation of population differences in disease phenotypes. Figure 1 provides a conceptual map of the structure of the article.

## 2. Preliminaries

### 2.1 Model

Our extended model is parallel to our previously reported model (Edge and Rosenberg 2015) and is similar to models used by Risch et al. (2002), Edwards (2003), and especially Tal (2012) to



Fig. Document page with Borderless Table

# Chapter 2

## 2.1 ASSUMPTIONS

The Following Assumptions were made in our Algorithm:

- Table Full Cell Structure is coming in Background Correctly
- No other Signs/Symbols in Background are included in Table Structure
- In bordered table line detection, it is assumed that the grayscale value of line pixel is higher compared to background pixel.
- Table data is coming in the foreground correctly

## 2.2 Criteria

Our project goal is to put the table element and its data into output HTML at their correct position.

# Chapter 3

## 3.1 ALGORITHM (Bordered Table)

Algorithm Input:

- Background Connected Component Glyphs
- Foreground Table Data Spans

Algorithm Output:

- HTML Table Structure with Data

### 3.1.1 Vertical Line Detection

Algorithm pseudocode:

- For each BG (background glyph):
    - Initialise a 1D array of size of BG bbox width as grey Hough parameter counting space (GHPCS)
    - Compute grey value for each pixel in BG bbox (grayscale = 0.2989 * red + 0.5870 * green + 0.1140 * blue)
    - For each pixel having (grayscale > line_pixel_grayscale_threshold):
        - Increment this pixel in GHPCS
    - Initialise a list VL
    - For a point in GHPCS having (value > verticle_line_threshold):
        - Push this point in VL

### 3.1.2 Horizontal Line Detection

Algorithm pseudocode:

- For each BG (background glyph):
  - Initialize a 1D array of size of BG bbox height as gray Hough parameter counting space (GHPCS)
  - Compute gray value for each pixel in BG bbox (grayscale = 0.2989 * red + 0.5870 * green + 0.1140 * blue)
  - For each pixel having (grayscale > line_pixel_grayscale_threshold):
    - Increment this pixel in GHPCS
  - Initialize a list HL
  - For a point in GHPCS having (value > horizontal_line_threshold):
    - Push this point in HL

### 3.1.3 Vertical/Horizontal Lines start and end pixel point

Algorithm pseudocode:

- For each data in VL:
  - Check for each pixel between two consecutive data in HL having (grayscale > line_pixel_threshold):
    - We got a vertical line between these two horizontal lines.
  - Merge the cells row wise for all consecutive discontinuous vertical line between two consecutive same data in HL.
- For each data in HL:
  - Check for each pixel between two consecutive data in VL having (grayscale > line_pixel_threshold):
    - We got a horizontal line between these two horizontal lines.
  - Merge the cells column wise for all consecutive discontinuous horizontal line between two consecutive same data in HL.

### 3.1.4 Cell Data Mapping

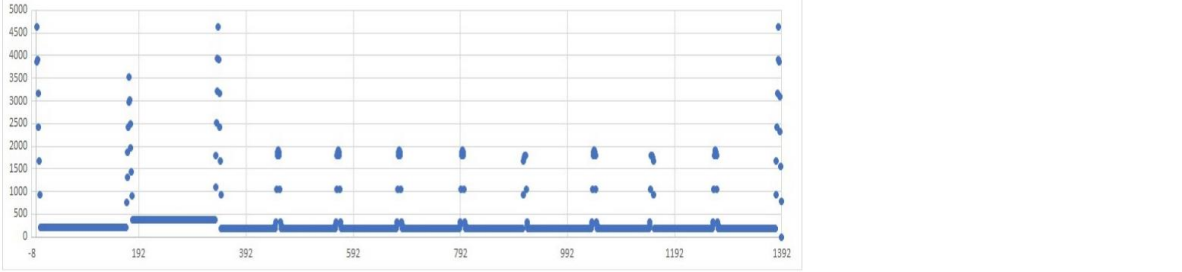
Now, using two consecutive horizontal and vertical lines, we get all the cell bbox.

Algorithm pseudocode:

- Input:
  - Cells bbox
  - Spans in BG region in a Hash table
- Output:
  - Cells' individual spans
- For each Span having space in innerText:
  - Make a new span corresponding to each split part
    - Compute bbox, innertext, glyphs of new span
  - Add new span to hash table
  - Add space span if required
  - Remove this span from hash table
- For each cell, Map Spans having its bbox inside the cell bbox and put them in a span list.
- For each cell, order the spans inside the span list according to 'inherent language order' (for English, it is left to right going vertically from top to bottom).

# 3.2 Bordered Table Algorithm Explanation with an Example

| Propriety | | a) | b) | c) | d) | e) | f) | g) | h) | i) |
|---|---|---|---|---|---|---|---|---|---|---|
| Zoom (0-100) | | 0,0001 | - | - | - | 0,0088 | - | 0,0005 | 0,0237 | - |
| Text Size (0-100) | | - | 0,0001 | 0,0008 | - | - | - | 0,0001 | - | 0,0001 |
| Link Enhance-ment | Box (0-1) | - | - | - | - | - | - | - | - | - |
| | Underline (0-1) | - | - | - | - | - | - | - | - | - |
| | Colour | - | - | - | - | - | - | 0,0108 | - | - |
| | Border Size (0-100) | - | 0,0001 | - | - | - | - | - | - | 0,0001 |
| Button | Enhanced (0-1) | - | - | - | - | - | - | - | - | - |
| | Colour | - | - | - | - | - | 0,0003 | - | - | - |
| | Border Size | - | - | - | 0,0003 | - | - | - | - | - |




We have plotted the sum of grayscale values row wise and column wise on the right side and the bottom in the above figure. We can see the graph shows local maxima (spike) corresponding to each line. In the figure, we have 12 spikes in the bottom graph and 11 spikes in the right graph. This corresponds to 12 vertical lines and 11 horizontal lines. So, now, we have got 11*12 cells structure table. In addition, we have 12 data in VL and 11 data in HL corresponding to the point of spike.

Now, we check the continuity of each horizontal line between consecutive vertical lines and merge the cells column wise.

Similarly, we check the continuity of each vertical line between consecutive horizontal lines and merge the cells row wise.

Finally, get the data by intersecting common box areas in background to foreground and ordering them in language inherent order.

# 3.3 ALGORITHM (Borderless Table)

We have utilised the CascadeTabNet[1] model for detection of the bounding box of tables in a document. It is a Cascade mask Region-based CNN High-Resolution Network (Cascade mask R-CNN HRNet) based model. It has shown best F1-Score for ICDAR 2013 Table database and was ranked 3rd in ICDAR 2019 Table detection competition.

- Model Input:
    - Document page Image(PNG file)
- Model output:
    - Bbox (Bounding Box) list with a score and a label {‘bordered’, ‘cell’, ‘borderless’’} corresponding to each box.

# Chapter 4

## 4.1 RESULTS (Bordered Table)

### 4.1.1 Dataset:

About 20 documents of samples folder were utilised in the analysis, which consisted of about total 50 tables. The results of our new Algorithm improvements give us the following results:

| Docs | Tables | Correctly Detected | Accuracy |
|---|---|---|---|
| 20 | 50 | 46 | 92% |

### 4.1.2 Thresholds

The Prior work already utilised certain thresholds for table structure detection. Vertical Line and horizontal line detection utilised a certain distance threshold for line detection. We changed these static thresholds to dynamic thresholds which depend on height and width of the document page. Similarly the line thickness thresholds utilised before was giving many ghost cells, as certain lines in the table were much thicker relative to other lines. To prune this error we set the line thickness threshold = 1 as static and introduced a dynamic minimum line distance threshold which depends on height and width of the page. This minimum line distance threshold never considers lines detected in close proximity which was happening before due to line thickness threshold.

## 4.1.3 Errors Removed

### 4.1.3.1 Cell Structure Error:

- Prior work was producing wrong colspan and rowspan value and Separating certain whole tables into 2 parts
- Explanation:
  - Due to variation in thickness of line in single table cells, extra ghost cells were created leading to colspan/rowspan error
  - Due to separate Background Glyphs of single Table its structure was coming partially in final HTML output
- Solution:
  - Introduced new threshold ‘minimum line distance’ to collapse certain ghost cells which are occurring due to Thickness threshold Edge/Line Detection
  - Utilised the above threshold to group together separate Background Glyphs of Single Table

### 4.1.3.2 Cell Data Error:

- Prior code was producing tables with missing space in cell data and Hyphenated words, Some data was missing altogether.
- Explanation:
  - In current RAVI, span having space as innerText has undefined bbox, unable to position it.
  - Span ids were not coming in ordered manner of cells
- Observation:
  - All span ids have spatial locality
- Solution:
  - Add space span between two non-space innerText span,(having id s1 and s2), if s1+1 or s2-1 is space span.
  - Join two data if there is a hyphen at end of s1 and s2 bbox is below s1 bbox
  - Search for all spans which are occurring inside a cell Bbox.

## 4.1.3.3 Multi Cells Span Error:

- Prior code was producing Multi cells span in starting cell leaving subsequent cell data blank
- Explanation:
  - Some span bbox in foreground is mapped to many cells in background table
- Observation:
  - All multi cells span have space in foreground at line pixel in background
- Solution:
  - Added split span with space
- Exception:
  - Only 1 table with ‘s’ char at line border pixel.

## 4.1.3.4 False Cell Error:

- Prior code was producing false cells
- Explanation:
  - Prior code is detecting lines if a single pixel value is above line pixel.
  - Some ‘_’ , ‘/’ symbols are coming in the background as horizontal lines.
- Solutions:
  - Added Hline Remove Function which removes Such Symbols from background
  - Check for continuity of horizontal line between consecutive two vertical lines
- Exception:
  - Certain NCERT Sample Docs are having these symbols in background which are not recognised by background Analyser Function as Separate Glyphs leading to not detecting them by our function
  - Due to Blue background of the Table Line Continuity test is passed by the above example due to which a false cell is detected.

# 4.2 RESULTS (Borderless Table)

For borderless table detection we are utilising CascadeTabNet[1]. We did an analysis on the working of the CascadeTabNet tool on our Sample Document of RAVI project. It consisted of 7 models for Bounding box detection of tables, here a table could be of bordered type or borderless type. Similarly one of the model i.e. epoch_36 also had a table classifier which classified the table into bordered or borderless type. This model also had a cell detector for borderless type tables for better table cell structure recognition.

## 4.2.1 Dataset:

About 250 pages of samples folder were utilised in the analysis, which consisted of about total 194 tables. Of the 194 tables, 47 were bordered tables and 147 were borderless tables.

## 4.2.2 Thresholds

We have assumed that score of 85% as threshold value for box detection for each bordered, cell and borderless.

The analysis of these models on our Sample Dataset is given below:

- Analysis of epoch_13 model:

| No. of Pages | Total Tables | True Tables | False Tables | % Accuracy | % False Tables |
|---|---|---|---|---|---|
| 250 | 194 | 148 | 17 | 76.28% | 10.30% |

- Analysis of epoch_14_B model:

| No. of Pages | Total Tables | True Tables | False Tables | % Accuracy | % False Tables |
|---|---|---|---|---|---|
| 250 | 194 | 156 | 12 | 80.41% | 7.14% |

- Analysis of epoch_17 model:

| No. of Pages | Total Tables | True Tables | False Tables | % Accuracy | % False Tables |
|---|---|---|---|---|---|
| 250 | 194 | 156 | 9 | 80.41% | 5.45% |

- Analysis of epoch_14_A model:

| No. of Pages | Total Tables | True Tables | False Tables | % Accuracy | % False Tables |
|---|---|---|---|---|---|
| 250 | 194 | 183 | 36 | 94.32% | 16.43% |

- Analysis of epoch_24 model:

| No. of Pages | Total Tables | True Tables | False Tables | % Accuracy | % False Tables |
|---|---|---|---|---|---|
| 250 | 194 | 182 | 25 | 93.81% | 12.08% |

- Analysis of epoch_1 model:

| No. of Pages | Total Tables | True Tables | False Tables | % Accuracy | % False Tables |
|---|---|---|---|---|---|
| 250 | 194 | 183 | 25 | 94.32% | 12.01% |

- Analysis of epoch_36 model:

| Total Pages | Total Tables | Bordered Tables | Borderless Tables | True Table Boundary | False Table Boundary | False Cell Pages |
|---|---|---|---|---|---|---|
| 250 | 194 | 47 | 147 | 168 (86.6%) | 14 | 45 (18%) |

- Analysis of table classifier of epoch_36:

| True Table | Correct Class | Conflict Class | Incorrect Class |
|---|---|---|---|
| 168 | 146 | 15 | 7 |

Here the % false tables is calculated as false tables detected divided by total tables detected (i.e. false + true tables). For the epoch_36 model there is an additional table showing the result of the table classifier which classifies the table into a bordered table or borderless table.

- Runtime Analysis:

| File Name | epoch_1 | epoch_13 | epoch_14_A | epoch_14_B | epoch_17 | epoch_24 | epoch_36 |
|---|---|---|---|---|---|---|---|
| Runtime/ Page (Sec) | 0.759 | 0.748 | 0.766 | 0.736 | 0.754 | 0.758 | 1.436 |

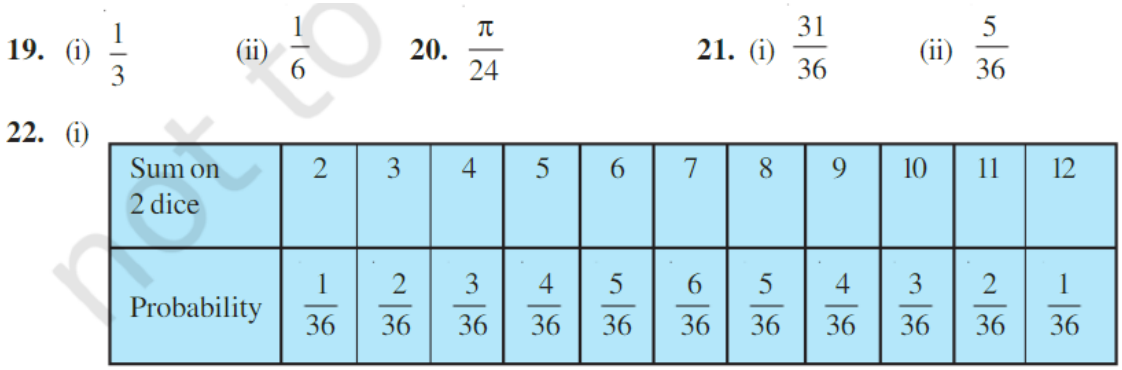
Runtime analysis shows that there is not much difference in average runtime of the first 6 models but the model epoch_36 takes about twice the amount of time than others. This may be because of the additional Table Classifier and Cell Structure Detector present in epoch_36 model.

# Chapter 5

## 5.1 FAILURE ANALYSIS (Bordered Table)

The algorithm failed mainly in the following case:

1. We failed to remove the False Cell Error in the NCERT book tables with blue background colour in the table structure.
    a. This happened because the blue colour background was also passing the line continuity test of our algorithm which led to detection of line due to the Divide symbol.
    b. Similarly removal of the divide symbol using RAVI tools was not possible as RAVI backgroundAnalysis() function was not capable of recognising the divide symbol coming inside table structure in the background due to the blue colour table.



**19.** (i) $\frac{1}{3}$ (ii) $\frac{1}{6}$ **20.** $\frac{\pi}{24}$ **21.** (i) $\frac{31}{36}$ (ii) $\frac{5}{36}$

**22.** (i)

| Sum on 2 dice | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Probability | $\frac{1}{36}$ | $\frac{2}{36}$ | $\frac{3}{36}$ | $\frac{4}{36}$ | $\frac{5}{36}$ | $\frac{6}{36}$ | $\frac{5}{36}$ | $\frac{4}{36}$ | $\frac{3}{36}$ | $\frac{2}{36}$ | $\frac{1}{36}$ |

Fig 1. Table in a NCERT with divide symbol and blue background colour.

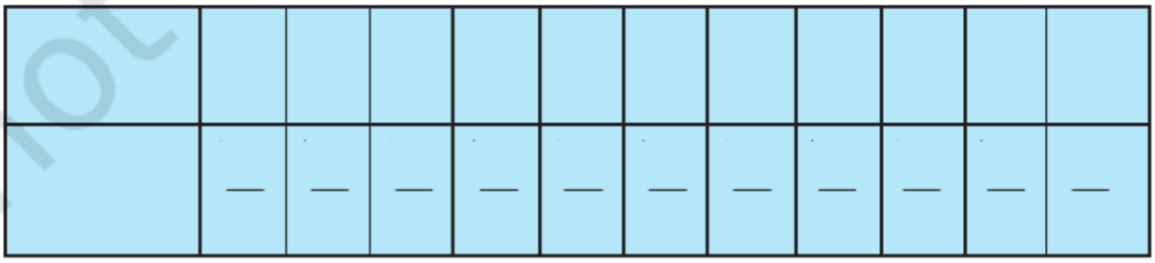


Fig 2. Tables RAVI background in a NCERT with divide symbol and blue background colour.

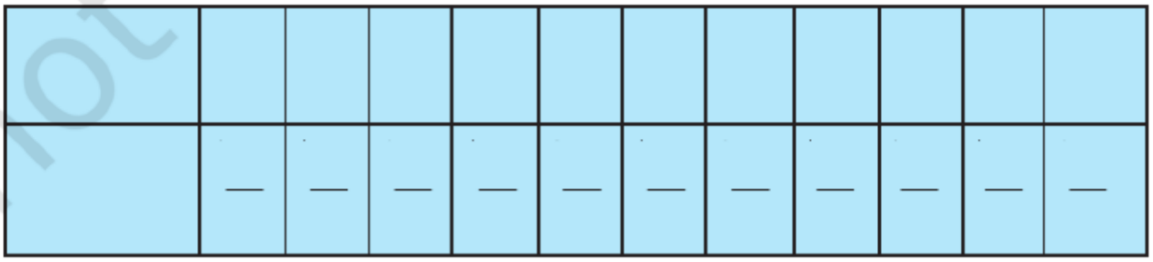


Fig 3. Tables RAVI background in a NCERT with divide symbol and blue background colour after symbol removal.

You can see from Figure 1 to Figure 3 the divide symbols of white background in the page are removed using RAVI tools and our RemoveHlines() function. But the symbols coming inside the table with the blue background are not detected by RAVI.

# Chapter 6

## 6.1 Integration with RAVI

## Workflow:

- let d = new pdf2charinfo.document();
- d.matchSpans();
  - Clusters textual glyphs coming in the foreground into spans.
- d.backgroundAnalysis();
  - Detects the background Glyphs
- d.bgcheck();
  - Removes the horizontal line symbols interfering with table line detection.
- d.tableAnalysis();
  - Detects the table structure from the background of RAVI and table data from the foreground of RAVI. Then it maps the cell structure with the correct cell data and places it in the document as HTML output.

# Chapter 7

## 7.1 Conclusion

We have tested our Bordered Table Algorithm on a total of 50 tables from Sample Documents in the RAVI project which contain diverse documents for better analysis. In our analysis we got 100% accuracy in detecting bordered table bounding boxes and 92% accuracy results in bordered table cells accessibility.

In CascadeTabNet, the best models epoch_24 and epoch_1 showed 93.81% and 94.32% accuracy in detecting table bounding boxes with relatively lower false positives accuracy.

## 7.2 Future Work

We have also found the following topic for future work:

- Our bordered table algorithm needs a color conversion for better final output as black pixel for table structure line and table background colour as white pixel.
- We have only got 30% accuracy in cell bbox detection in the borderless table. We need to find an alternate high accuracy model for this in future work.
- We propose to integrate one of the models from epoch_24 or epoch_1 for table bounding box detection and to integrate epoch_36 model for table classification and cell structure recognition.

# References

[1] CascadeTabNet: An approach for end to end table detection and structure recognition from image-based documents

- Author : Devashish Prasad, Ayan Gadpal, Kshitij Kapadni, Manish Visave, Kavita Sultanpure [CoRR , abs/2004.12629]
- Year={2020}, eprint={2004.12629}, archivePrefix={arXiv}, primaryClass={cs.CV}

[2] Gray-scale hough transform for thick line detection in gray-scale images

- Author : Rong-chin Lo,  Wen-Hsiang Tsai
- Publisher : Elsevier BV
- Citation Data : Pattern Recognition, ISSN: 0031-3203, Vol: 28, Issue: 5, Page: 647-661
- Publication Year : 1995