Article Segmentation in Newspapers and Forms

A Project Report
submitted by

SAI VARUN SEEMAKURTHI
EE16B159

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

DUAL DEGREE

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS

June 2021

CERTIFICATE

This is to certify that the thesis entitled "**Article Segmentation in Newspapers and Forms**", submitted by **Sai Varun** (EE16B159), to the Department of Electrical Engineering, Indian Institute of Technology, Madras, for the award of the **Dual Degree** (B.Tech. and M.Tech.), is a bonafide record of research work carried out by him under my supervision. The content of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any other degree or diploma.

Prof. Srinivasa Chakravarthy V

Project Guide

Professor

Computational Neuroscience Laboratory

Department of Biotechnology

Indian Institute of Technology Madras

Chennai – 600 036.

Prof. S. Umesh

Project Co-Guide

Professor

Department of Electrical Engineering

Indian Institute of Technology Madras

Chennai – 600 036.

Date: 25^{th} June, 2021

Place: Chennai.

ACKNOWLEDGEMENTS

My first and foremost gratitude goes to **Prof. Srinivasa Chakravarthy**, who guided me to carry on this work. He allowed me to be part of his group and work under his supervision. He is the source of steady encouragement, guidance and motivation.

I would also like to thank my co-advisor **Prof. S Umesh** for his support during this project.

My heartfelt gratitude to **Vigneswaran** who helped and guided me throughout in the successful completion of this work.

I would heartily extend my gratitude for my team members: Hareesh, Sai Charan and Roshan for their support and encouragement.

Finally I would like to thank my parents, friends, all researchers of this world and also the Almighty God, without them this project would not have become possible.

ABSTRACT

KEYWORDS: You Only Look Once (YOLO), Character-Region Awareness For Text Detection (CRAFT), OpenCV, Dilation, Erosion, Infor -mation Retrieval (IR), Segmentation

Document analysis and recognition is increasingly used to digitise collections of historical books, newspapers and other periodicals. In the digital humanities, it is often the goal to apply Information Retrieval (IR) and Natural Language Processing (NLP) techniques to help researchers analyse and navigate these digitised archives. Though we have powerful text recognition algorithms, the lack of article segmentation is impairing many information retrieval systems, which assume text is split into ordered documents. In this report, we define a document analysis and image processing task for segmenting digitised newspapers into articles and other content. This report also discusses the use of YOLOv3 in detecting the required segments from Telugu newspaper. Considering we deal with a lot of forms in our daily life especially in India where many people are not conversant enough with English, form-processing system on Indic languages has a great value in office automation in India. This report also discusses a novel idea for identifying the fields like name, signature etc. in the forms. In this way this report considers and discusses three use cases to elaborate the concept of Segmentation.

List of Figures

Contents

INTRODUCTION

Nowadays every kind of information is going digital and internet oriented. In fact the analog life tends to be a memory of our past. Especially newspaper archives are a valuable record of culture and history from recent centuries. Large-scale digitisation, through commercial and government projects, promise their ready access to current and future generations. Hence we must think for the huge archives of many kind of documents to be digitized with minimum cost and the most accurate automatic way. Applying optical character recognition (OCR) to microfiche and complex layouts does not yield individual, error-free documents. This not only impairs user navigation of historical archives, but greatly impedes the application of many IR and NLP systems.

Newspapers are documents which are made of articles, news items, advertisements, photos. They are not meant to be read iteratively to reach the desired topic but the reader can pick his item in any order he wishes. Most of the digitized newspaper archives only offer them by issue ignoring this structural property.

Most business and government organizations in India use forms as the major means in collecting information. As the huge quantities of forms make manual processing a labor intensive task, so the automation of this procedure attracts intensive research interest

In this thesis three examples of segmentation are given. Detection of paragraphs from Telugu newspaper, Form Processing and Article Segmentation of Historic English Newspapers ignoring the non articles text.

LITERATURE SURVEY

Digitising historical text is very difficult due to complex layouts, deteriorated microfiche, inconsistent fonts, bleed through, faded text, and distorted scans [1].Most of the earlier work on layout analysis is done on journal pages which does not have complex layout where text and graphic regions are placed in a random fashion. The problem of layout analysis for newspaper images is addressed by few authors. Gatos et al [7] proposed an integrated methodology for segmenting newspaper page and identifying newspaper article. In the first stage, various regions are extracted using smearing and connected component labeling. A rule based approach is applied in the second stage to extract various regions and newspaper articles. Liu et al [8] presented a component based bottom-up algorithm for analyzing newspaper layout. This algorithm is based on a distance measure and layout rules which are designed heuristically.Wang et al [13] classified newspaper image block using textual features. The technique proposed assumes homogeneous rectangular blocks extracted using RLSA and Recursive X-Y cuts. The blocks are then classified based on statistical textual features and space decision techniques. Learning based methods have also been reported in literature for block labeling, layout analysis and other document processing tasks. Bukhari et al [4] proposed a layout analysis technique for Arabic historical document images. They extracted and generated feature vector in a connected component level. Multi-layer perceptron classifier was used to classify connected components to the relevant classes of text. A voting step is applied for the final classification and refinement. By carefully defining the article segmentation task, and describing our new state-of-the-art approach, we aim to encourage further progress in this challenging task.

BACKGROUND THEORY

In this chapter a brief background details are provided on neural network models, tools and methods that are used in this work.

3.1 Darknet

Darknet[9] is a Neural Network framework written in CUDA and C. With its help we can build, train and run the neural networks. It supports computation with both CPU and GPU. Darknet is an opensource and is available on GitHub.

3.1.1 Architecture

3.2 YOLOv3: You Only Look Once

YOLOv3[10] is a real time object detection algorithm that identifies specific objects in videos, images.

YOLO is a fully Convolutional Network. CNNs (Convolutional Neural Networks) are class of feed forward deep neural networks, mostly used in analysis of images. They process the images as structured arrays of data and identify the relationships between them. The advantage of YOLO is that it is faster than other architectures maintaining the accuracy.

3.2.1 Algorithm

It separates out an image in many grids. Each grid cell predicts number of anchor boxes around the objects that score highly with predefined classes. The boundary boxes are generated by clustering the dimensions of the ground truth boxes from the original dataset to find the most common shapes and sizes.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 3.1: Darknet-53 Architecture

Darknet 53 layers – the head: Darknet 53 is a classifier network and is used as the first part of the complete architecture.

Bounding box layers – the tail: The second part of YOLOv3 analyzes the output of Darknet-53 and first searches for what it believes to be reasonable bounding boxes and then classifies what is inside the bounding boxes. The final result of the network is a 3D tensor containing information about found bounding boxes and what class of object it predicts is inside it.

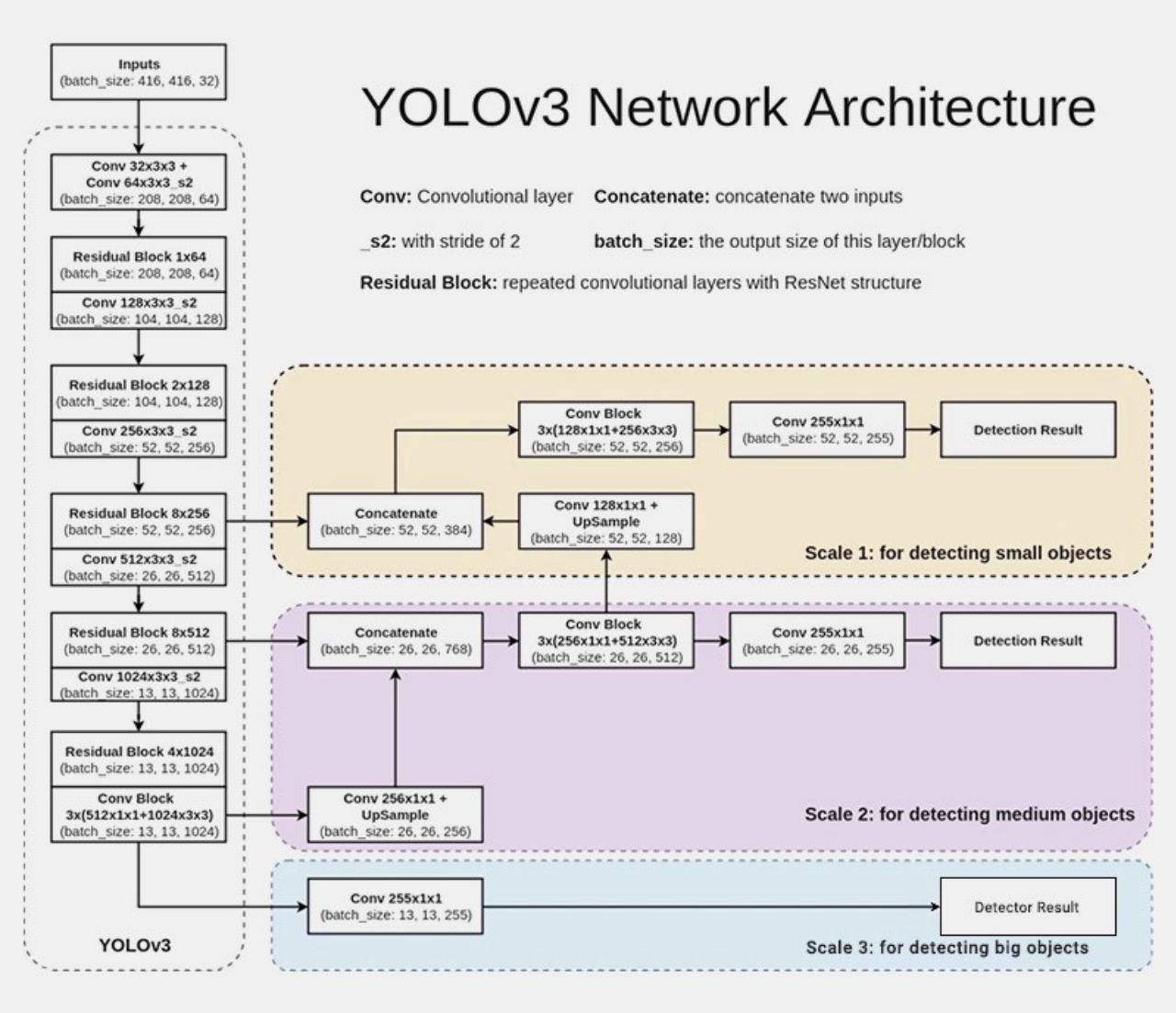


Figure 3.2: YOLOv3 Architecture

3.3 CRAFT: Character-Region Awareness For Text detection

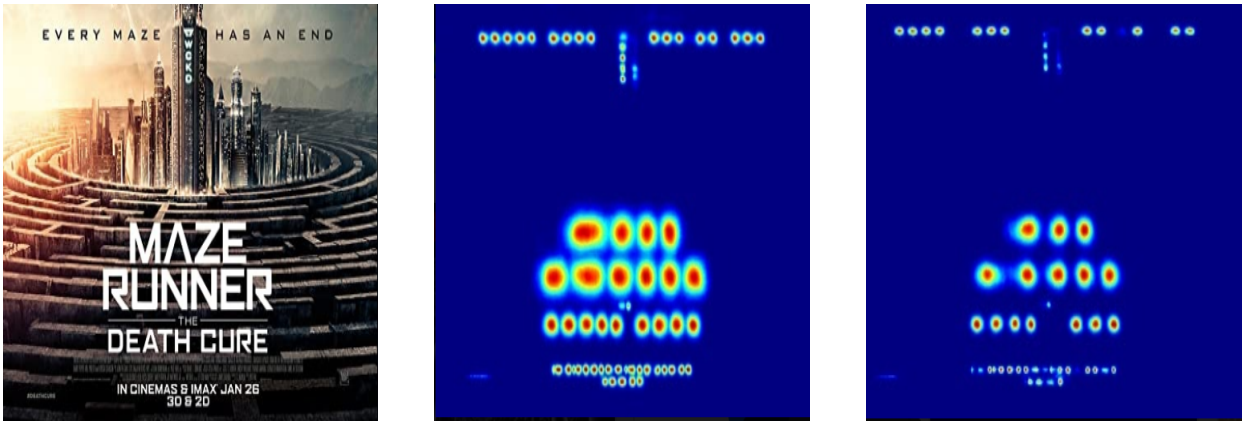
The primary goal of CRAFT: Character-Region Awareness For Text detection is to pinpoint individual character areas and connect the detected ones to a text instance.[2]

The backbone of CRAFT is VGG-16, which is a fully convolutional network

architecture. To put it another way, VGG16 is the feature extraction architecture that is utilised to encode the network's input into a feature representation. The CRAFT network's decoding section is comparable to UNet's. It has skip connections, which group low-level characteristics together.

For each character, CRAFT forecasts two scores:
Region score: As the name implies, it specifies the character's geographical location. It localizes the character.

Affinity score: The degree to which one material prefers to mix with another is referred to as "affinity". As a result, an affinity score combines many characters into a single instance.
Both scores combine to give the bounding boxes.
CRAFT generates two maps: Region Level Map and Affinity Map.



(a) Input image (b) Region Map (c) Affinity Map

Figure 3.3: CRAFT

3.3.1 Architecture

Following is the architecture of CRAFT -

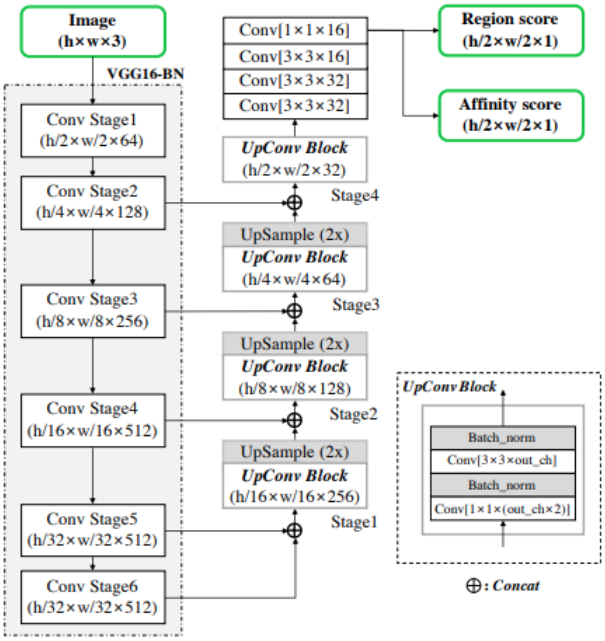


Figure 3.4: CRAFT Architecture

3.4 PIL: Python Imaging Library

PIL[12] library adds image processing capabilities to the python interpreter. It has many modules like ImageDraw, ImageEnhance etc.

3.5 OpenCV: Open Source Computer Vision Library

OpenCV[3] is a free software library for computer vision and machine learning. It was created to offer a standardised infrastructure for computer vision applications and to let commercial goods incorporate machine perception quickly. It has over 2500 optimised algorithms and has python, C++, Java interfaces.

3.5.1 Template matching

Template matching is a technique for searching and determining the location of a smaller template image in a larger one. For this we have a function called cv2.matchTemplate(). Just like in 2D convolution, it merely slides the template picture over the input picture and then compares the template and the patch of input picture under the template picture. This function returns a gray scale image, where bounding boxes around the matched region.

3.5.2 Find Contours

The contours are defined as a line that joins all the points that have the same intensity on the boundaries of a picture. In shape analysis, contours are helpful to discover the size of the item, and to detect objects.

OpenCV has the findContour() method to extract the contours from the picture. It works better with binary images, thus we first have to apply thresholding, sobel edges, etc.

3.6 Deskewing

The technique of eliminating skew from images photographs is known as deskewing. Skew can occur in scans due to camera misalignment, scanning defects, or the newspaper not being positioned totally level when scanned.
When an there is a skew in the image, optical character recognition (OCR) is more difficult and becomes slower and less accurate. Deskewing the documents beforehand can make the OCR process faster and more accurate.

3.7 Morphological operations

They are a series of procedures that process images based on shapes. Morphological operations create an output picture by applying a structuring element to an input picture. Dilation and erosion are two very important morphological operations.

3.7.1 Erosion

The basic concept of erosion is similar to that of soil erosion. It erodes away the borders of foreground object. The kernel slides through the image similar to that case of 2D convolution. A pixel in the original image which can either be 1 or 0, will be considered 1 if all the pixels under the kernel are 1, else it is made 0.

As a result, depending on the size of the kernel, all the pixels near the boundary will be discarded. As a result, the foreground object's thickness or size shrinks. It is very useful in removing small white noises, separate two connected items etc.

3.7.2 Dilation

It is just opposite to erosion. If there is at least one pixel under the kernel as 1, the corresponding pixel is made as 1. So, this effectively increases the size of foreground object.

Generally, in cases where noise has to be removed, erosion is followed by dilation. Erosion removed the noises, but also decreases the size of our object. So, we dilate, which results in increase in area of object.

3.7.3 Opening

Opening is a technique in which first erosion operation is executed and then dilation operation is executed. It removes the thin protrusions of the acquired image is used for effectively reducing noise of the resulting image.

3.7.4 Closing

Closing is a technique in which first dilation is carried out and then erosion is carried out. It removes the little troughs from the image. Closing is used to smooth outline and fuse small fractures.

3.8 Sobel filter

It is used for edge detection. An edge is a location in the image where there is sudden change of intensity or the color of pixels. It operates by computing the gradient of image at each pixel. It determines the direction of the most significant growth from light to dark and rate of change in that direction. The result indicates how suddenly or smoothly the image changes at each pixel, and hence how probable that pixel is an edge.

3.9 Binarization

Image binarization converts any image to a black and white image. The way to do this is to choose a threshold value, and classify all pixels with values below this threshold as black and all other pixels as white.
Choosing the threshold is very challenging. Choosing a single threshold for the entire image may be very difficult in many cases, therefore adaptive image binarization is used in those cases, wherein optimal threshold is chosen for each local area.

3.10 Light correction

Surface relief or book curvature cause light variance, which can be corrected using lighting adjustment. We implemented a method described in the paper [5]. This method achieves light homogeneity and removes shadows, whether they are vertical or horizontal. The technique suggested can achieve a higher performance in terms of both balancing the light variance and also achieving a high degree of text recognition.

3.11 AMPD: Automatic Multi scale Peak Detection

AMPD[6] is a signal processing algorithm used for finding peaks in noisy periodic and quasi-periodic signals. This method is quite robust and efficient against low and high frequency noise. There is a parameter called scale(say k) which indicates the algorithm considers windows up to +-k either side of the peak candidates.

EXPERIMENTATION AND RESULTS

Three experiments have been performed related to Segmentation and Locating the fields in various kinds of newspapers.

4.1 Paragraph detection and word segmentation in Telugu newspapers

4.1.1 Objective

To locate and segment the paragraphs in the Telugu newspapers altering YOLOv3[3.2]. To segment upto word level using CRAFT[3.3].

4.1.2 Methodology

Dataset: A set of 10 Telugu newspaper images are taken. Refer Figure 4.1a

Annotation: Each image will be associated with a .txt file having the details of object class and corresponding coordinates of the bounding box around that object.
In this case, paragraphs are annotated using a annotation tool called Make Sense [11] and the coordinates are noted in the following syntax -

<object_class> <x_center_norm> <y_center_norm> <width_norm> <height_norm>

```
x_center_norm = x_center_absolute/image_width
y_center_norm = y_center_absolute/image_height
width_norm = width_of_label_absolute/image_width
height_norm = height_of_label_absolute/image_height
```

(a) Telugu Newspaper

0 0.096667 0.320039 0.146667 0.235409
0 0.880000 0.449416 0.233333 0.077821
0 0.275000 0.325875 0.176667 0.138132
0 0.761667 0.550584 0.436667 0.077821
0 0.696667 0.329767 0.260000 0.138132
0 0.496667 0.679961 0.213333 0.064202
0 0.918333 0.318093 0.156667 0.165370
0 0.628333 0.456226 0.236667 0.087549

(b) Annotated coordinates

Figure 4.1: Sample of Telugu newspaper Training data with annotated coordinates

There is only 1 class in our case which we call as "Text" class.Refer Figure 4.1b

Augmented Dataset: Since any deep learning model requires large amount of data for training, in an attempt to generate more data, random Gaussian noise is added to each image to get 100 images.
Finally, we get a dataset of 1000 images which we will use for training our YOLO model. The text files are also replicated correspondingly to get 1000 text files.

Training YOLO: **Hardware**: The training process was done in Google colab

using NVIDIA Tesla K80 GPU.

Next, we created 4 files so that YOLOv3 knows how and what to train. They are obj.names.txt, train.txt and test.txt, obj.data.txt.

object.names.txt: It contains the names of our object classes in the format

object_1_classname

object_2_classname

. . . . . .

object_n_classname

Since in our case we have only 1 class i.e Text, we enter the same and save the file.

train.txt: It contains all the file paths of the training images.

test.txt: It contains all the file paths of the test images.

These two files share the same syntax for paths – data/ image_path.

obj.data.txt: This includes the number of classes, paths to train.txt, test.txt, obj.names.txt and the backup folder Backup folder is where we want to store the weights.

In short the dataset which we will be using is:

|- newspaper.obj/

| |-1.jpg

| |-1.txt

| |- . . . . .

|- object.names

|- train.txt

|- test.txt

Training framework setup: Training was done using the neural network framework Darknet. Configuration files for Darknet describing YOLOv3 were taken from the github project Darknet. The most important file to be downloaded in yolov3.cfg which has the architecture related details of the network and also the hyperparameters.

Hyperparameters: For training, we chose the batch size (number of samples in one batch) to be 64, subdivisions (number of mini batches in one batch) to be 16 and learning rate to be 0.001, max_batches =4000(number of training iterations)

Altering the architecture of neural network:

The default setting is to detect 80 classes. It is changed to detect 1 class. The setting for number of filters in convolutional layer will be

filters=(classes + 5)*3

Hence number of filters will 18. Each instance of yolo layer along with its immediate preceding convolutional layer were edited. See example -

```
[convolutional]
size=1
stride=1
pad=1
filters=255 <- WAS SET TO 18
activation=linear

[yolo]
mask = 6,7,8
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,..
classes=80 <- WAS SET TO 1
```

```
num=9
jitter=.3
ignore_thresh = .7
truth_thresh = 1
random=1
```

Training is carried out then for 3k iterations.

4.1.3 Results

Training Loss curve:

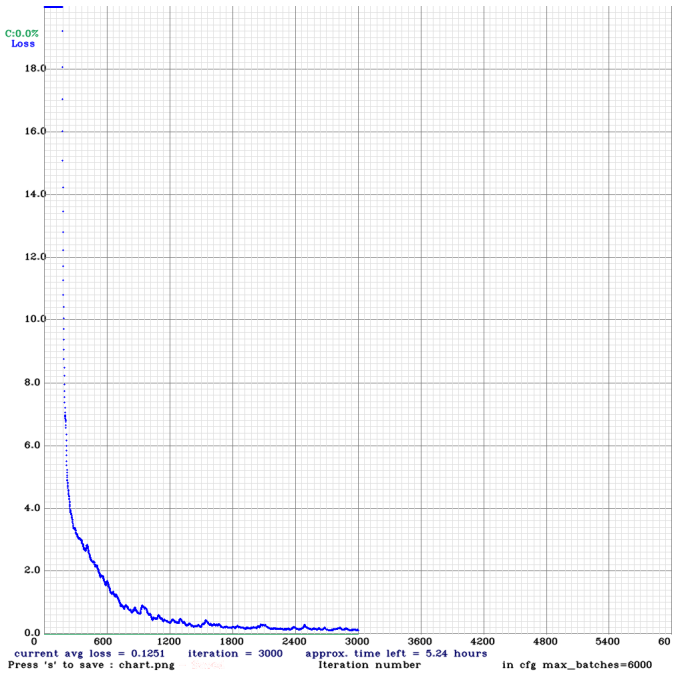


Figure 4.2: Training Loss Chart of YOLO on Telugu newspaper

Test results:

With even poor training data i.e with just 10 distinct images, we get almost

Figure 4.4: Test results of Telugu newspaper paragraph segmentation

good results. Hence this architecture is robust in detecting the required articles in a newspaper.

Word level Segmentation: Pytorch implementation of CRAFT is adopted

and implemented on a segmented paragraph from the above result.

Results-

Figure 4.5: Test results of Word Segmentation of Segmented Paragraph

These segmented and cropped words can be sent to the Telugu OCR which can recognize the words.

4.2 Form Processing

4.2.1 Detecting Fields in Custom images

Objective: To create a dataset of forms and test the performance of YOLOv3 in detecting various fields of that image.

Image description: Each image will consist of 3 fields or classes – Name, Father's name and Place (in Telugu). Around 300 names and places are collected by scraping various Wikipedia pages.
On a plain image, 3 random positions are selected to insert the corresponding fields making sure that they don't overlap. The 3 coordinates are stored in a

text file. Rest of image was also filled with random text again scraped from random Wikipedia pages. Care was taken that no words overlap and words don't cut into half at the end of line. Also, all words were having random gaps between them and also random colors.

The ImageDraw Module of PIL[3.4] was used to write the Telugu words on the image when coordinates are specified with various colors.

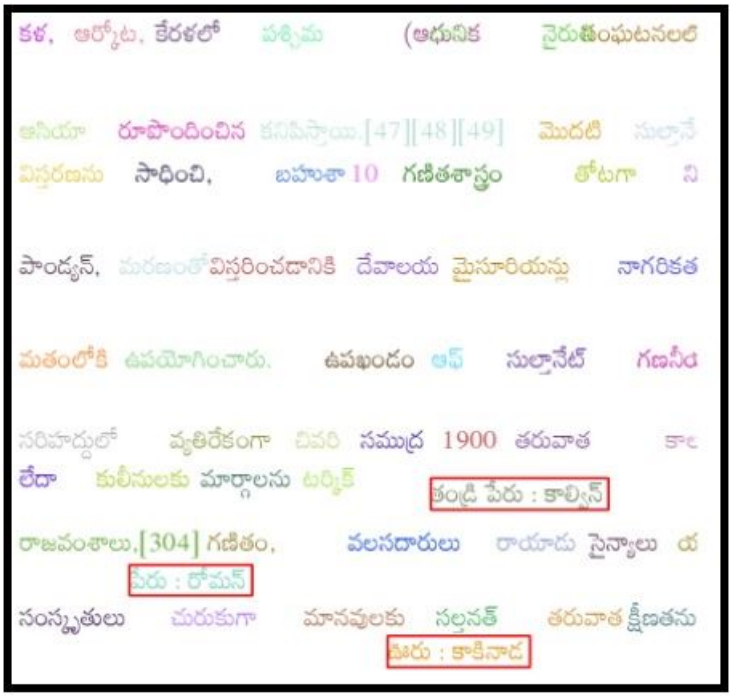


Figure 4.6: Data sample of Custom created form

Dataset: 1000 images are generated for training. Corresponding text files are also generated. Same procedure is followed as in the previous experiment.

object.names.txt: It will have three entries. They are 'Name', 'Father's name' and 'Place'.

train.txt and **test.txt** will contain the paths to the train images and the test images.

obj.data.txt: This includes the number of classes, paths to train.txt, test.txt, obj.names.txt and the backup folder.

Hyperparameters will be the same as chosen in the previous case.

Altering the architecture of neural network: Since there are 3 classes here, number of filters in the convolutional layer are choosen as 24.

Training is carried out then for 3k iterations.

Results

Training Loss curve:

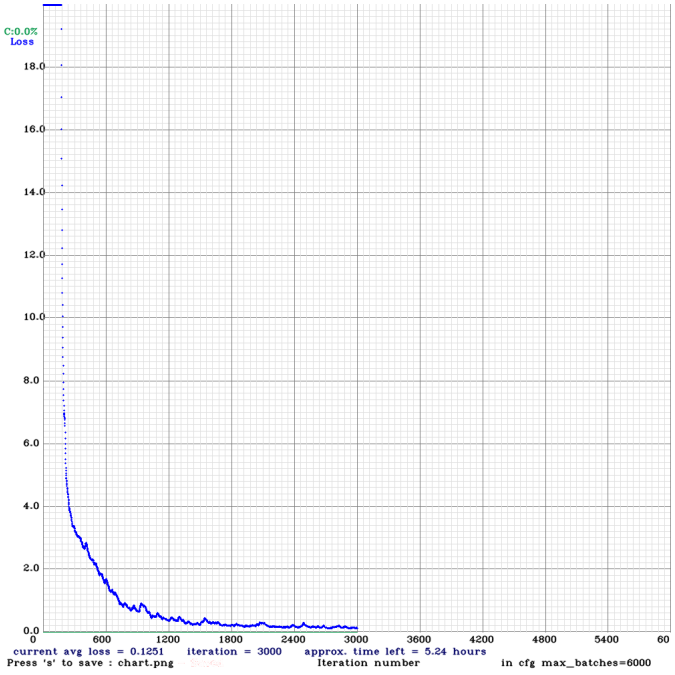


Figure 4.7: Training Loss Chart of YOLO on Custom telugu forms

Test results:

Detected fields are cropped. Using the Telugu OCR, these segmented and cropped fields can be recognized.

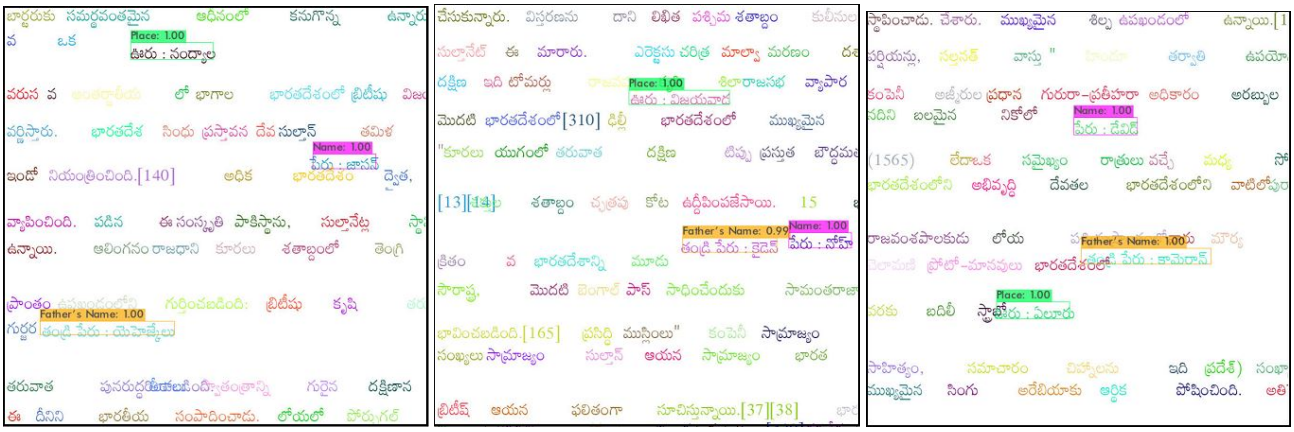


Figure 4.8: Test results of Segmented fields in Custom created forms

4.2.2 Detecting Fields in Certificates

Objective: To develop a generalised model for identifying the fields such name, signature etc. in the certificates.

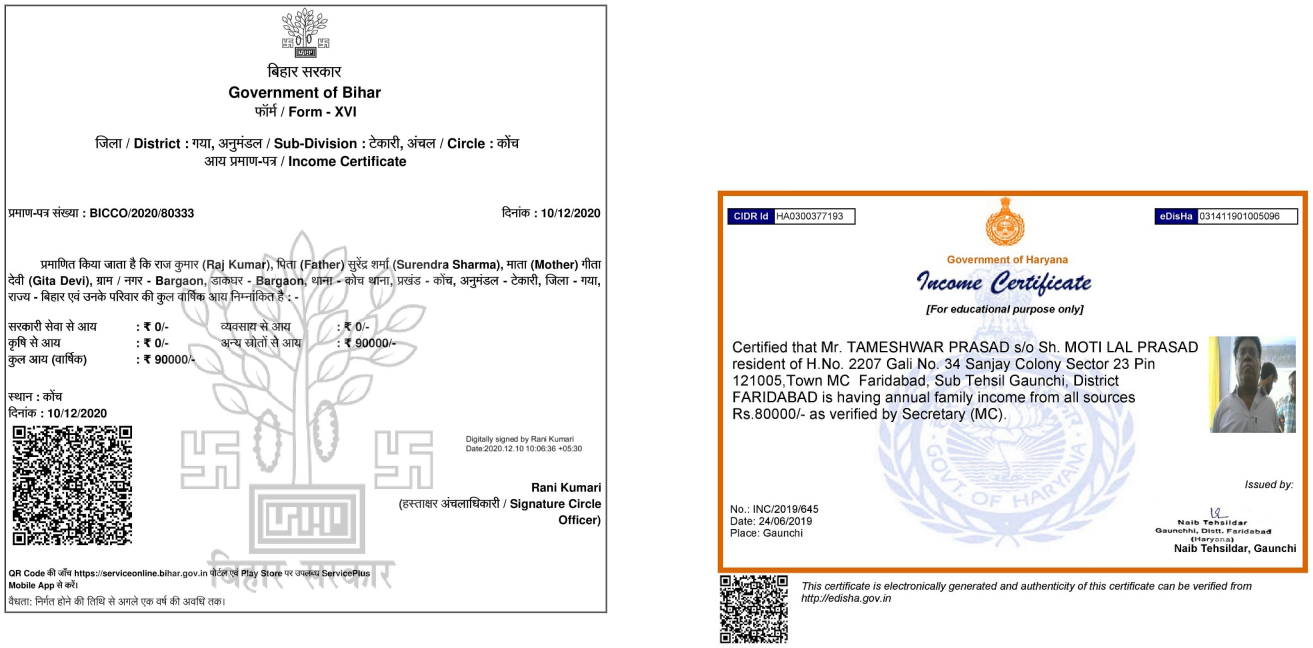


Figure 4.9: Certificates

Methodology: As we know to train any Deep Learning model, we need lot of data. In this case it was very difficult for us to obtain such huge training data and to annotate them requires so much manual endeavour.

One approach suggested is to replicate the certificates in Word. While training the model to detect the name field, one can annotate the name, its preceding word, next word, labels etc. for various layouts of certificates. This enables the model to learn the surroundings of name field.

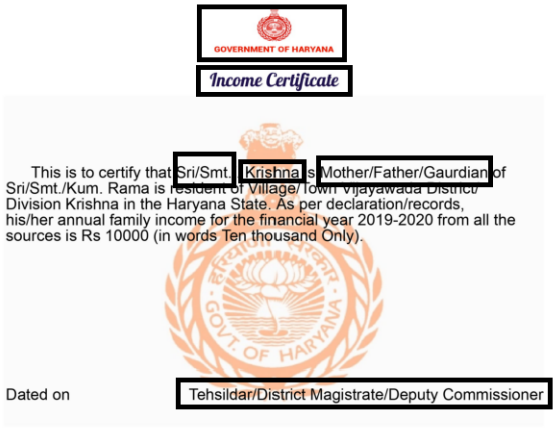


Figure 4.10: Annotated Duplicated Certificate

4.3 Segmentation of Historic English newspapers

4.3.1 Objective

To segment the newspaper into various articles which is an important step in the digitization of historic newspapers.

4.3.2 Methodology

Data: Newspapers which dated to 19th century. Figure 4.11

Algorithm proposed: On observing the various layouts of newspapers given, we devised this algorithm to segment the newspaper into articles.

1. Deskewing [3.6]

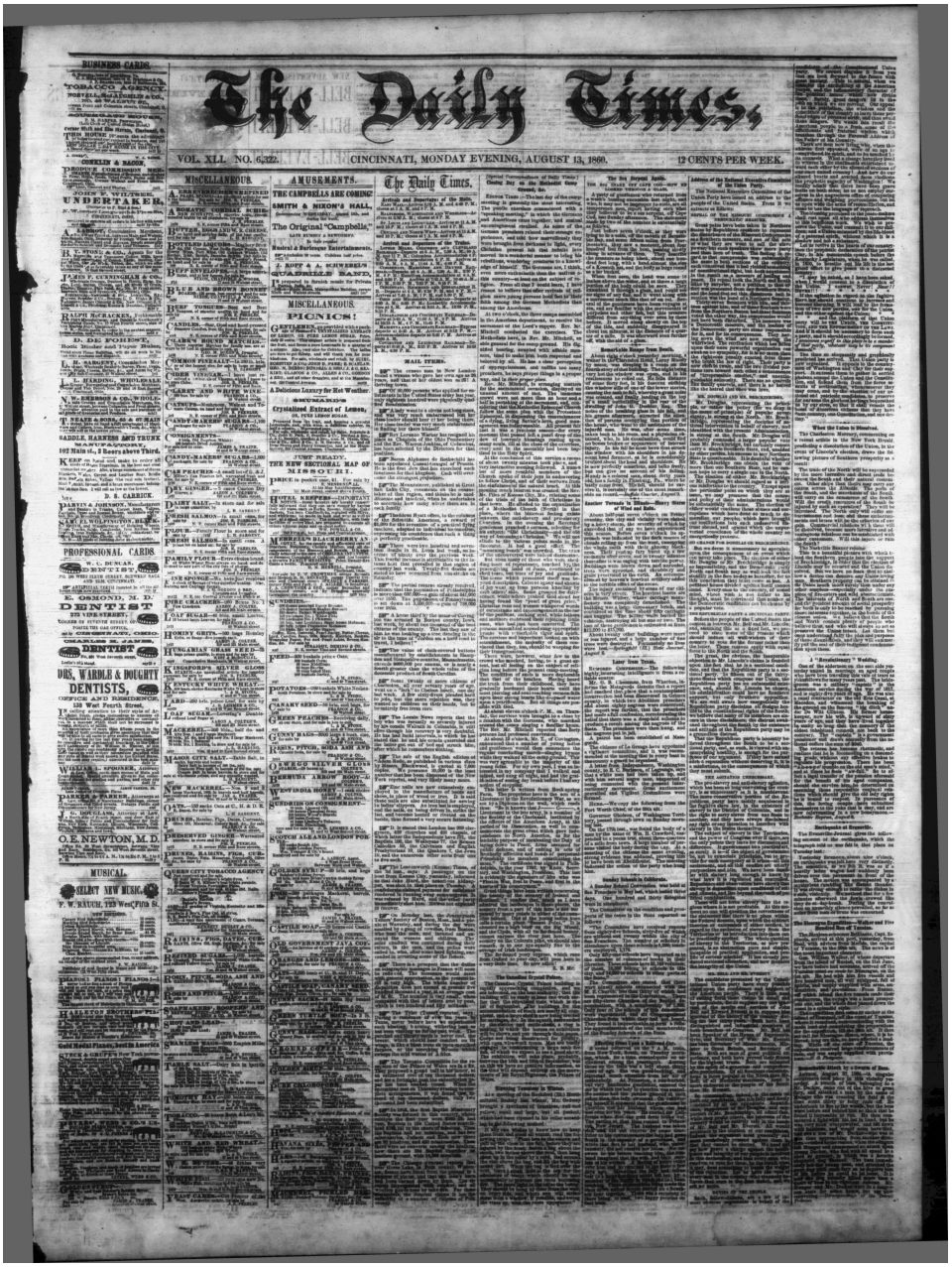


Figure 4.11: Sample of Historic newspaper

2. Column Segmentation
3. Light correction[Refer - 3.10]
4. Article Segmentation

Deskewing:

Skew is the amount of rotation necessary to return an image to horizontal and

vertical alignment. Skew is calculated and the image is rotated in the opposite direction with the same angle.

Column Segmentation:

Projection based method:

These methods are most commonly used for printed document segmentation. The horizontal projection profile can be obtained by summing up the pixel values along the vertical axis for each x value.

$$\textbf{profile(x)} = \sum_{y=0}^{y=M} f(x, y)$$

Horizontal projection of the above page:

The above profile can be smoothened by using a median filter to remove the

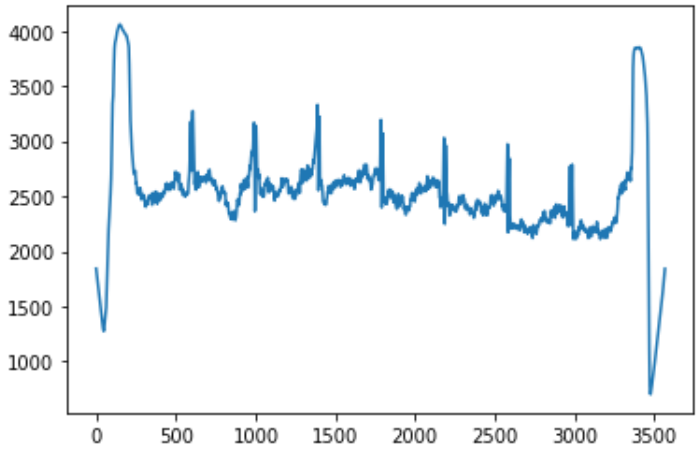


Figure 4.12: Horizontal Projection of Historic newspaper

local maxima.

The peaks in the above noisy graph correspond to the column separators. There are many local maxima, so it is difficult to exactly find the peak. Hence we used the AMPD algorithm [3.11] to find out the peaks corresponding to the column separators. Figure 4.13. Column segmentation is done according to such peaks as shown in Figure 4.14

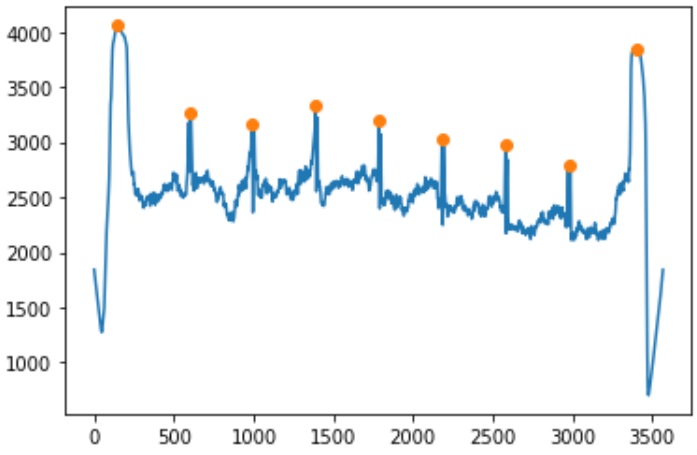


Figure 4.13: Horizontal Projection with detected peaks

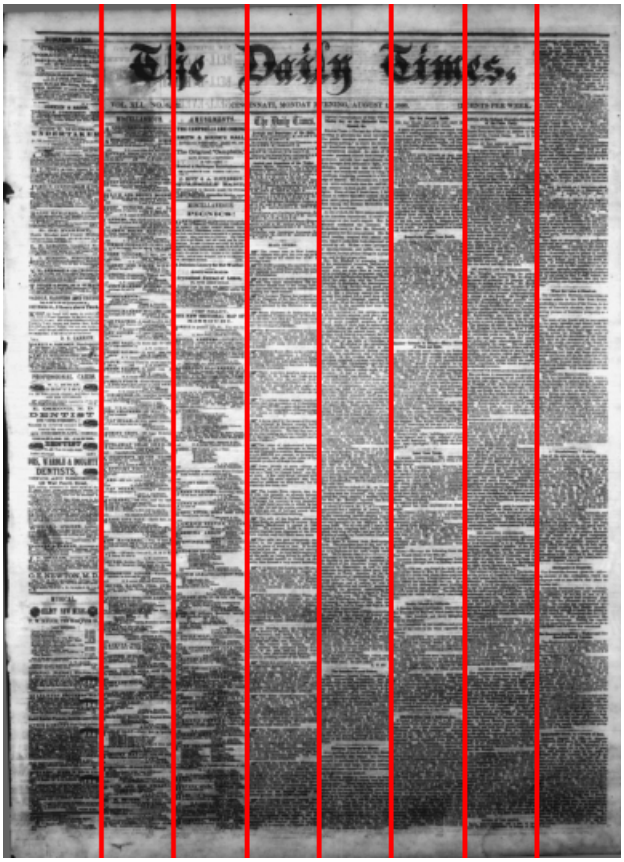


Figure 4.14: Column segmented image of Historic newspaper

To avoid missing few characters on either side, after obtaining a column, we padded a small space either side and then segmented the column again using

the same projection approach. Figure 4.15

There is internal skew between the lines i.e the lines are in itself not parallel to each other. Hence to reduce that skew, we divided a page into smaller (say 3 or 4) horizontal parts. Now applying the above projection method for column segmentation reduced this skew.

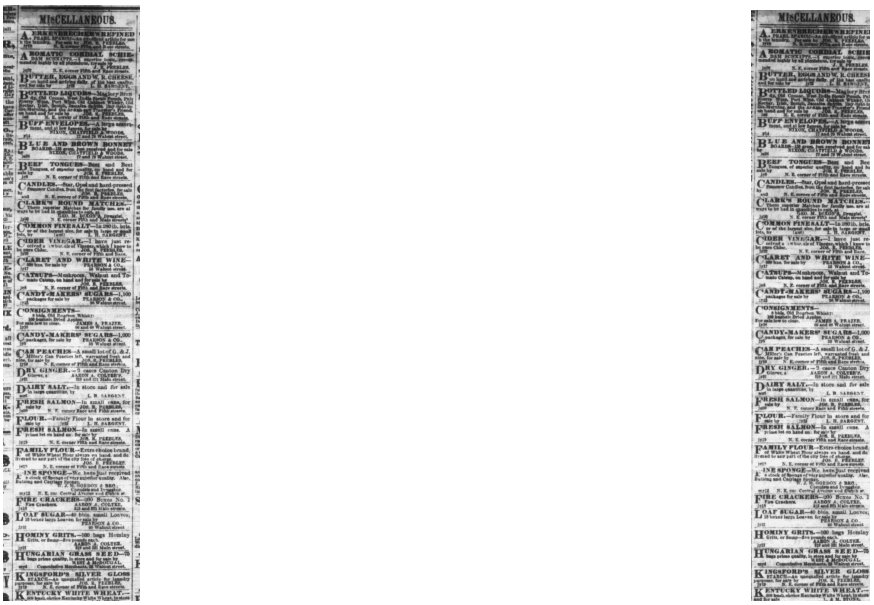


Figure 4.15: Padding followed by Segmentation

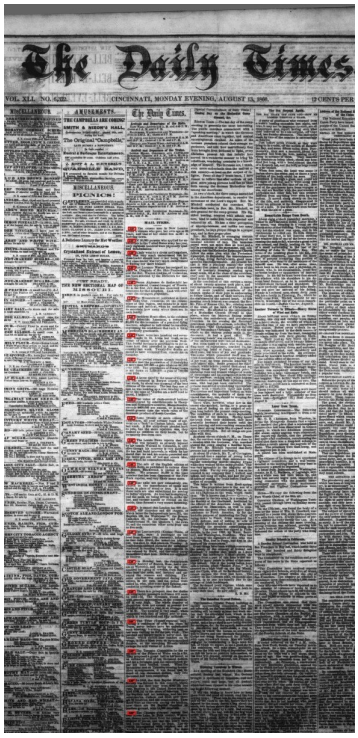
Identifying symbols:

There are few symbols such as hand, decorated line, mouth etc. in the newspapers, which are acting as article separators or identifiers. Identifying them will help us in identifying the articles. We cropped out the symbols of hand, mouth and decorated line which was used as a template.

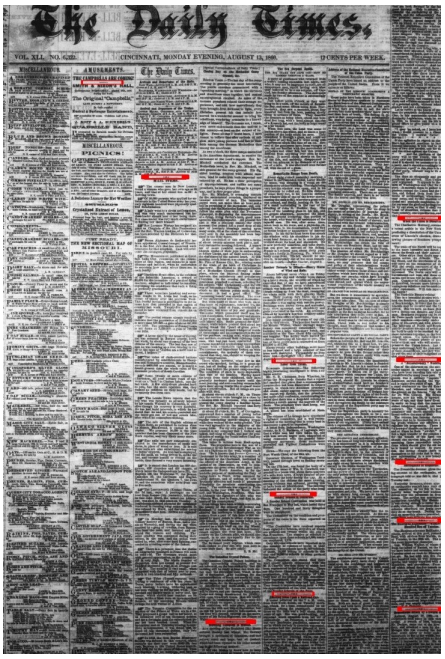
Template matching technique is used for this purpose. cv2.matchTemplate(), an Open CV method is used for this task. [Refer section 3.5.1].

Results:

Light Correction:



(a) Hand detection



(b) Decorated line detection

Figure 4.16: Detected symbols in Historic newspaper sample

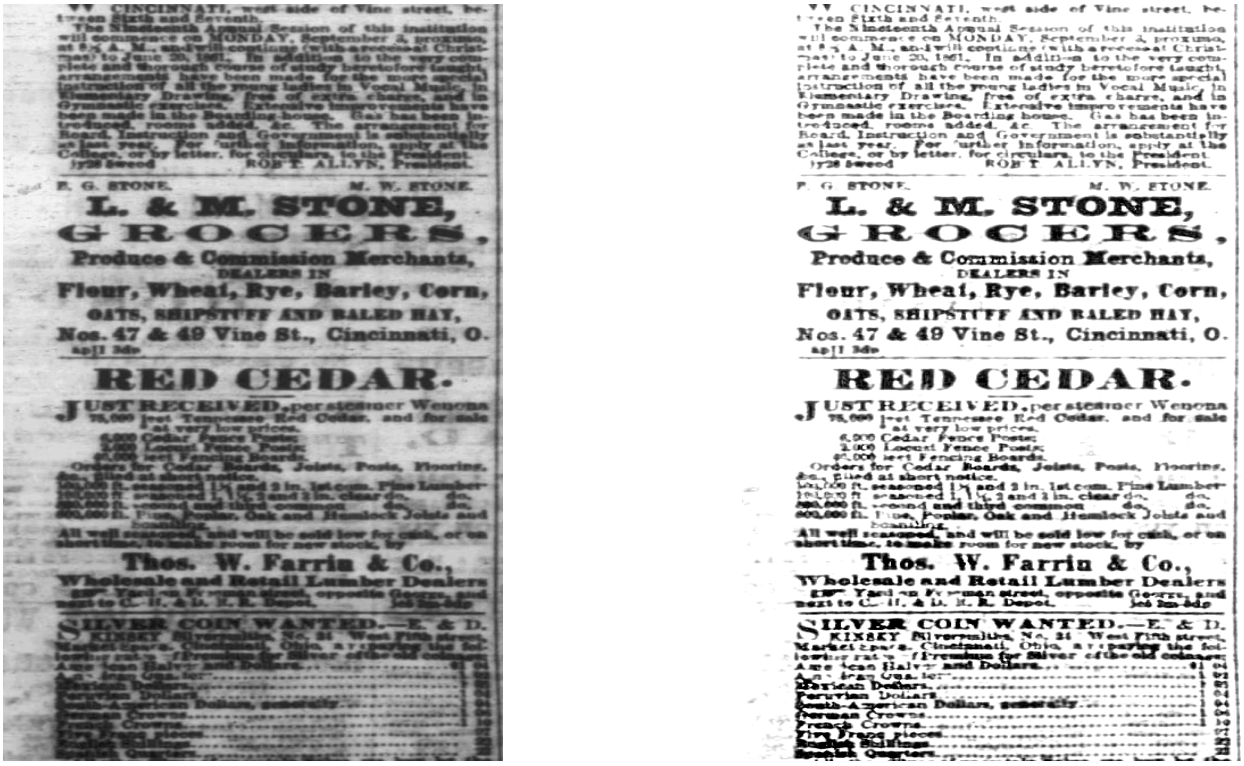
A portion of column is considered. On applying the technique as referred in 3.10, we got enhanced and noise reduced image. Figure 3.10

Article Segmentation:

The image obtained above is first converted to Grayscale and then various morphological operations are applied as discussed in [3.7].

Horizontal kernel: Taking use of the image's unique structure, instead of utilizing a general kernel with dilation/erosion, we used a horizontal kernel that connects the endpoints of horizontal lines but does not link the neighbouring lines.

$$\textbf{Horizontal kernel} = \begin{bmatrix} 1 & 1 & 1 & 1 & . & . & . & . & 1 \end{bmatrix}$$



(a) Noisy image

(b) Corrected Image

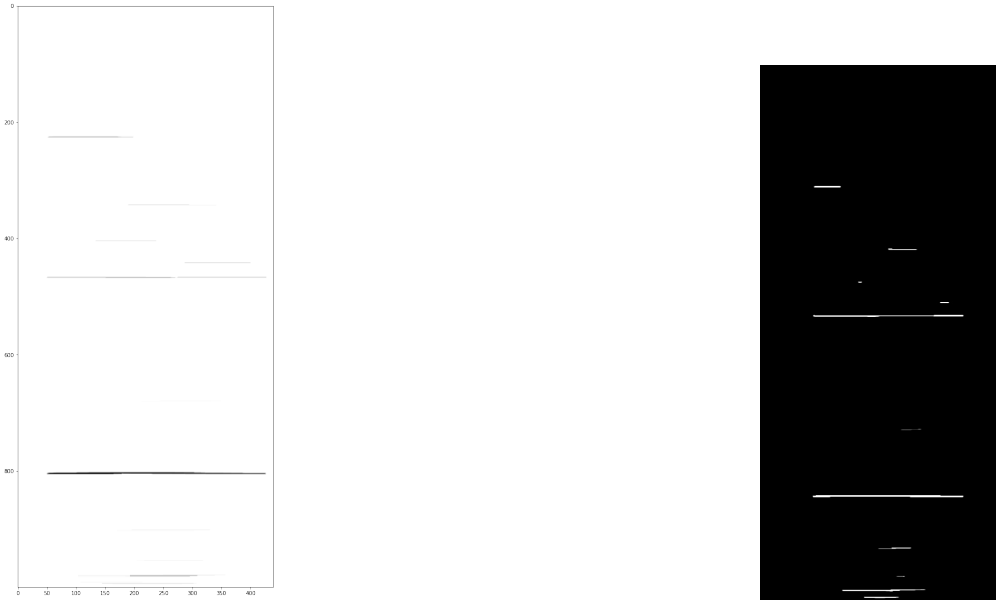
Figure 4.17: Light Correction of Historic newspaper sample

Closing operation i.e dilation followed by erosion is performed. Sobel operator is applied followed by erosion is done. Figure 4.19

Vertical projection profile is done by summing up the pixel values along the horizontal axis for y value. Figure **??**

Peaks correspond to the possible article separators and are found out using the AMPD algorithm considering some base value to be a peak and pruning away the near values. Lines corresponding to the peaks are drawn in the original image.

In the result we found that there are no False negatives. False positives are fine but false negative should not be there i.e it should be able to detect all actual separator lines. False positives can be eliminated easily.



(a) Closing (b) Sobel + Erosion

Figure 4.18: Morphological Operations on Segmented Column

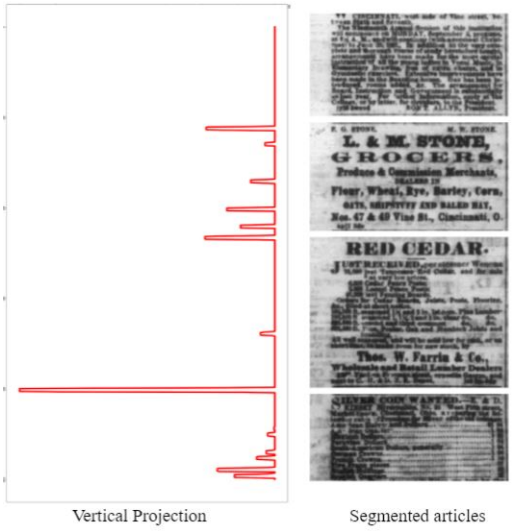


Figure 4.19: Article Segmentation of Segmented Column

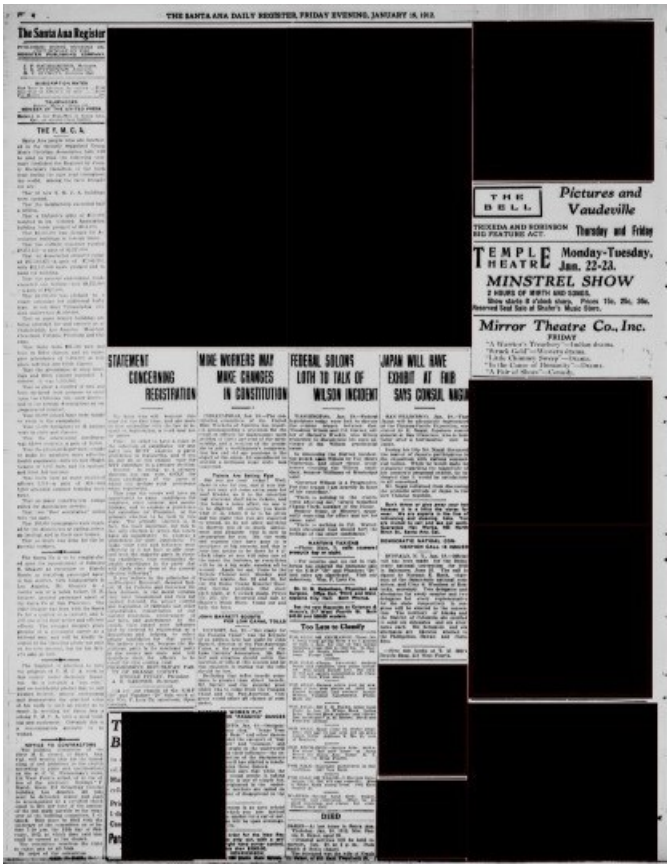
Segmenting Advertisements: For a certain layouts of newspapers which we had, Column segmentation cannot be done directly. Figure 4.20

The advertisements need to be removed before performing the column segmentation operation. The specific feature of this layout is that all the advertise-

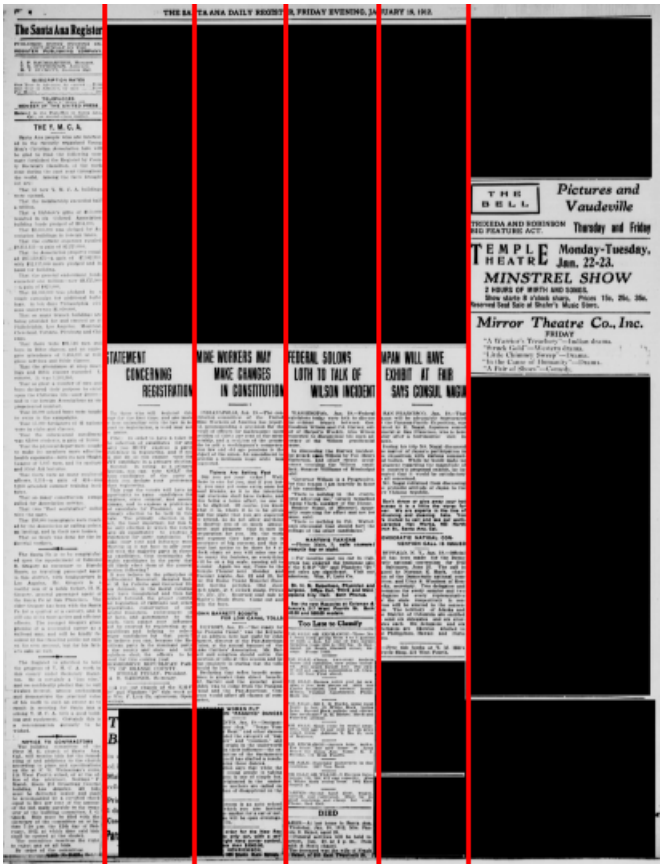
ments are enclosed by rectangles. Multiple iterations of erosion are done so that the boundary lines are thickened and then the method called findcontours of OpenCV which was discussed in [3.5.2] section was used to detect these rectangles.

After finding the advertisements, that area is blackened and the previous algorithm starting with column segmentation is followed to get the segmented articles.

Figure 4.20: Santa Ana Layout



(a) Modified image



(b) Resultant image

Figure 4.21: Column Segmentation of Layouts with advertisements

CONCLUSION AND FUTURE WORK

In this report, we presented an approach of segmenting paragraphs in Telugu newspapers using YOLOv3 and word level segmentation using CRAFT model which can in turn be fed into Telugu OCR for recognition. We also discussed a novel approach to detect the field entities in forms. We also presented article segmentation in Historic English newspapers using document analysis and various image processing techniques.

Future work includes of improving the algorithm for the form processing problem, finding ways to generate more form data, improving the accuracy of article segmentation in the English newspapers, text recognition part of Old Historical newspapers and converting the results into XML format.

References

[1] Kenning Arlitsch and John Herbert. "Microfilm, Paper, and OCR: Issues in Newspaper Digitization. The Utah Digital Newspapers Program". In: *Microform and Imaging Review* 33 (Mar. 2004). DOI: 10.1515/MFIR.2004.59.

[2] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. "Character Region Awareness for Text Detection". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 9365–9374.

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

[4] Syed Saqib Bukhari, Thomas M. Breuel, Abedelkadir Asi, and Jihad El-Sana. "Layout Analysis for Arabic Historical Document Images Using Machine Learning". In: *2012 International Conference on Frontiers in Handwriting Recognition*. 2012, pp. 639–644. DOI: 10.1109/ICFHR.2012.227.

[5] Kuo-Nan Chen, C. Chen, and C. Chang. "Efficient illumination compensation techniques for text images". In: *Digit. Signal Process.* 22 (2012), pp. 726–733.

[6] Alperen Mustafa Colak, Yuichiro Shibata, and Fujio Kurokawa. "FPGA implementation of the automatic multiscale based peak detection for real-time signal analysis on renewable energy systems". In: *2016 IEEE International Conference on Renewable Energy Research and Applications (ICRERA)*. 2016, pp. 379–384. DOI: 10.1109/ICRERA.2016.7884365.

[7] B. Gatos, S.L. Mantzaris, K.V. Chandrinos, A. Tsigris, and S.J. Perantonis. "Integrated algorithms for newspaper page decomposition and article tracking". In: *Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR '99 (Cat. No.PR00318)*. 1999, pp. 559–562. DOI: 10.1109/ICDAR.1999.791849.

[8] Fei Liu, Yupin Luo, M. Yoshikawa, and Dongcheng Hu. "A new component based algorithm for newspaper layout analysis". In: *Proceedings of Sixth International Conference on Document Analysis and Recognition*. 2001, pp. 1176–1180. DOI: 10.1109/ICDAR.2001.953970.

[9] Joseph Redmon. *Darknet: Open Source Neural Networks in C*. http://pjreddie.com/darknet/. 2013–2016.

[10] Joseph Redmon and Ali Farhadi. *YOLOv3: An Incremental Improvement*. 2018. arXiv: 1804.02767 [cs.CV].

[11] Piotr Skalski. *Make Sense*. https://github.com/SkalskiP/make-sense/. 2019.

[12] P Umesh. "Image Processing in Python". In: *CSI Communications* 23 (2012).

[13] Dacheng Wang and Sargur N Srihari. "Classification of newspaper image blocks using texture analysis". In: *Computer Vision, Graphics, and Image Processing* 47.3 (1989), pp. 327–352. ISSN: 0734-189X. DOI: https://doi.org/10.1016/0734-189X(89)90116-3. URL: https://www.sciencedirect.com/science/article/pii/0734189X89901163.