DETECT AND RECOGNIZE TEXT IN NATURAL IMAGES

A report submitted for the course named Project IV (CS-410)

By

RAJESH KUMAR

ROLL NO. 15010105

Supervised by

Dr. Nongmeikapam Kishorjit Singh

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY MANIPUR

Bachelor of Technology, Computer Science and Engineering

April,2019

Department of Computer Science & Engineering
Indian Institute of Information Technology Manipur

Dr. Nongmeikapam Kishorjit Singh Email: kishorjit@iiitmanipur.ac.in
Assistant Professor

Certificate

This is to certify that the project report entitled **"DETECT AND RECOGNIZE TEXT IN NATURAL IMAGES"**, submitted to the Department of Computer Science and Engineering, Indian Institute of Information Technology, Manipur, in partial fulfillment for the award of the degree of **Bachelor of Technology** in Computer Science and Engineering, is a record of bona fide work carried out by Mr.**RAJESH KUMAR**, Roll No.**15010105**, under my supervision and guidance.

No part of this report has been submitted elsewhere for award of any other degree.

Signature of Supervisor

(Dr. Nongmeikapam Kishorjit Singh)

Date :

Department of Computer Science & Engineering
Indian Institute of Information Technology Manipur

Certificate

This is to certify that the project report entitled **"DETECT AND RECOGNIZE TEXT IN NATURAL IMAGES"**, submitted to the Department of Computer Science and Engineering, Indian Institute of Information Technology Senapati, Manipur, in partial fulfillment for the award of the degree of **Bachelor of Technology** in Computer Science and Engineering, is a record of bona fide work carried out by Mr.**RAJESH KUMAR**, Roll No.**15010105**.

(Dr. Nongmeikapam Kishorjit Singh)
Assistant Professor Head,
Department of CSE
IIIT, Manipur

Signature of Examiner 1:

Signature of Examiner 2:

Signature of Examiner 3:

Signature of Examiner 4:

Declaration

I declare that this submission represents my idea in my own words and where others' idea or words have been included, I have adequately cited and referenced the original source. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/sources in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and can also evoke penal action from the sources which have thus not been properly cited or from proper permission has not been taken when needed.

(Signature)

(Rajesh Kumar)

Date: (15010105)

Acknowledgement

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.
I am highly indebted to **”Dr Nongmeikapam Kishorjit Singh”** *for their guidance and constant supervision as well as for providing necessary information regarding the project also for their support in completing the project.*
I would like to express my gratitude towards my parents and member of "Department of Computer Science and Engineering, IIIT Manipur" for their kind co-operation and encouragement which help me in completion of this project. I would like to express my special gratitude and thanks to **”Dr Kabita Thaoroijam”**, **”Dr Navanath Saharia ”** , **”Dr Prerna Mohit ”** *and* **”Mr. Himangshu Sarma”***many other industry persons for giving me such attention and time.*
My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

- Rajesh Kumar

Contents

List of Tables

List of Figures

List of abbreviations

F

FAST	Features From Accelerated Segment Test

M

MSER	Maximally Stable Extremal Regions

O

OCR	Optical Character Recognition

S

SWV	Stroke Width Variation

Abstract

In images, some documents embedded. Recognition of text space and extracting that text from that that image may be a novel drawback. several others offer there own approach for it, however in most of them basic digital image process techniques are accustomed find text from the photographs. This contain preprocessing of image , extraction of the image or text localization of image , classification and character detection of image. the various classification strategies used are SVM, AdaBoost, CNN , Text-CNN etc. This project provides an in depth study of evolution of text detection in natural pictures. It compares, analyzes and additionally discusses the various strategies to beat existing challenges in text detection.

Reading text from images is basically a difficult task that has received a significant quantity of attention. 2 key parts of most systems is (i) text detection from pictures and (ii) character recognition, and lots of recent methods are projected to style higher feature representations and models for both . This project applies strategies recently developed in machine learning specifically, large-scale algorithms for learning the options mechanically from unlabeled information and show that they permit to construct extremely effective classifiers for each detection and recognition to be utilized in a high accuracy end-to-end system.

This project shows a way to find regions in a picture that contain text. This is a common task performed on unstructured scenes. Unstructured scenes images that contain undetermined or random eventualities. as an example, detect and recognize text mechanically from captured video to alert a driver a couple of road sign. this is often totally different than structured scenes, that contain acknowledged eventualities where the position of text is understood beforehand.

Keywords - MSER, OCR , Image processing , Unstructured scene , Structured scene , Aspect ratio , Eccentricity , Euler number, Extent , Solidity , Stroke width Variation

Chapter 1

Introduction

Detection of text and identification of characters in scene pictures could be a difficult visual recognition downside. As in field of laptop or computer vision, the challenges display by the complexity of those pictures are combated with hand designed options and models that incorporate numerous items of high-level previous information.

In distinction to additional classical OCR issues, wherever the characters square measure generally monotone on fastened backgrounds, character recognition in scene pictures is probably way more complicated thanks to the numerous potential variations in background, lighting, texture and font. As a result, building complete systems for these eventualities needs to create representations that account for all of those kinds of variations.

Indeed, vital effort has gone into making such systems, with high performers integrating dozens of smartly combined options and process stages. Recent add machine learning, however, has wanted to form algorithms which will learn higher level representations of knowledge mechanically for several tasks. Such systems may be significantly valuable wherever specialised options square measure required however not simply created by hand. Another potential strength of those approaches is that it will simply generate massive numbers of features that alter higher performance to be achieved by classification algorithms.

Segmenting text from the unstructured image or scene greatly helps with extra tasks such as optical character recognition (OCR). The automatic text detection formula during this, detects an oversized range of text region candidates and increasingly removes those less likely to contain text.

Flow of project :

1.Detect Candidate Text Regions exploitation MSER

2.Remove Non-Text Regions supported Basic Geometric Properties

3.Remove Non-Text Regions supported Stroke breadth(Width) Variation

4.Merge Text Regions For Final Detection Result

5.Recognize Detected Text exploitation OCR

Chapter 2

Existing System Study

2.1 Text Detection in Document Images

2.1.1 Optical Character Recognition

The development of character recognition in last decade is outstanding and also the technique for character detection is immense. The advancements of Character Recognition is evident in Optical Character Recognition (OCR), Document Classification, This advanced computer or pc Vision, Data Mining, form Recognition, and identification . Character recognition is the method to categorise the input character per the predefined character class . Character recognition from some years has become a big curious area for research and it has its own application in identification of text in pictures. The text perhaps a scanned document or a written text.

A. Text from Natural Images:
In previous some years, the graphical trend to digitalize documents [1] has increasing upward and it has been emerged positively. With digitalization of the globe the paper primarily based documents ought to be born again into digital for additional handy,searchable and protective of the documents. Optical Character recognition is employed for this method. OCR will be represented as Mechanical or electronic conversion of scanned images wherever pictures will be written, written or written text. For over

a 0.5 century analysis during this space is in progress and character recognition rate in trendy OCR is above ninety nine percent on a top quality document and ninety percent on written documents. For degraded documents and books the potency of OCR comes all the way down to eighty percent. In recent times, several organizations rely on OCR for higher performance and additional efficiency.OCR will be performed offline and/or on-line. on-line recognition the OCR processor acknowledges the character as they're given. In offline methodology, the processor may acknowledge each document furthermore as written characters however recognition in offline mode extremely depends on the standard of the scanned pictures.

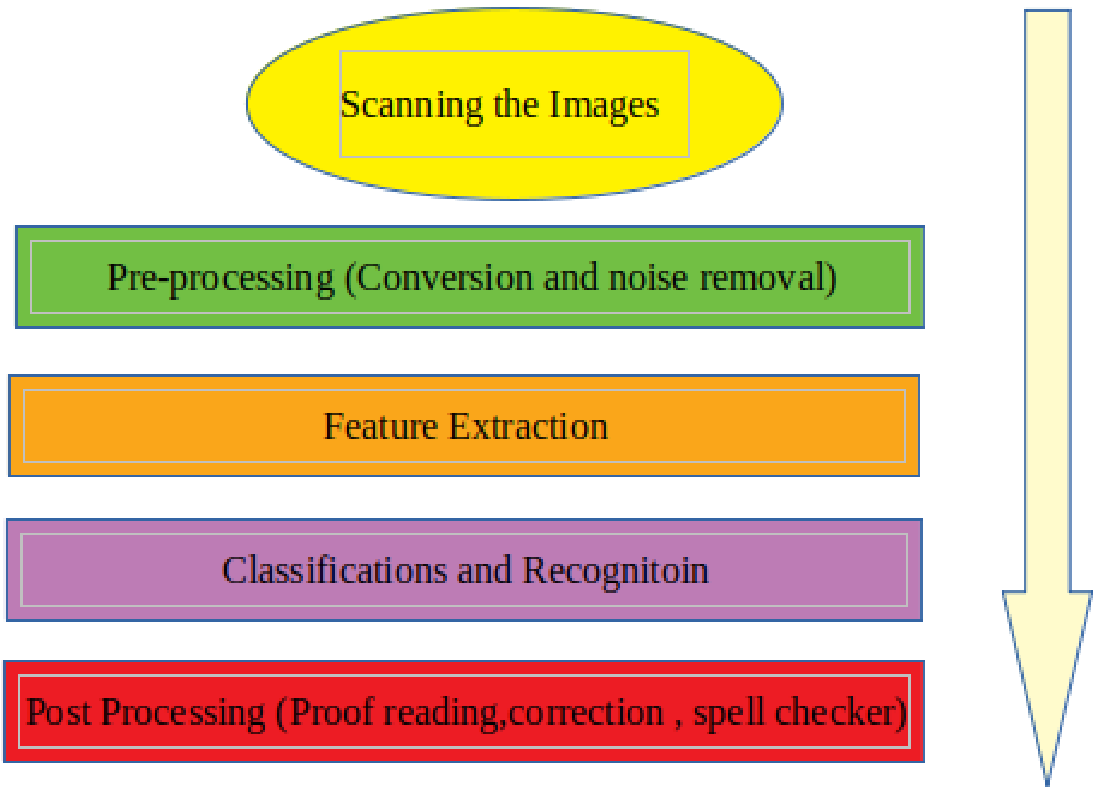

Figure 2.1: Stages for OCR

OCR consists of the many phases like Scanning [2] of image, Pre-processing, Segmentation, Feature Extraction, Classifications and Recognition [3], Post process. The task of preprocessing relates to the removal of noise and variation within the image. In scanning step the image is nonheritable. the standard of image depends extremely on the scanner being used. In sensible applications, the scanned pictures don't seem to be excellent there is also some noise thanks to some supererogatory details within the image which might cause a stoppage in the detection of the characters within the image. Preprocessing involves removal of noise (applying filters like mathematician filter, Gabor filter etc.) and correct conversion of image like a colored image will be born-again into grey scale or binary image for more process of image. Feature extraction involves recognizing the feature needed. Classifications and Recognition part is that the extraction part of the method. once finishing the OCR process many postprocessing steps area unit necessary reckoning on the applying, e.g. tagging the documents with meta-data (author, year, etc.) or proof-reading the documents for correcting OCR errors and writing system mistakes. OCR remains in analysis and far advancement ought to be created during this technology. the longer term scope of this can be OCR in mobile devices, handwriting recognition, recognition of assorted languages except English (like Arabic, Devanagari, Telugu text), extraction and process of pictures from video, processing and restoration of recent documents and plenty of additional.

2.1.2 Document Images

A document image contains numerous info like texts, photos and graphics. These pictures obtained by scanning written documents, recent documents, printed documents, journals etc. several challenges faces for recognizing scanned documents like low distinction, low resolution, color hurt, advanced background and unknown text color, size, position, orientation, layout etc. notwithstanding the OCR system is of supreme quality the system will still not provide correct output because of the issues mentioned higher than. Generally, the method of OCR works best if the background of the image is clean and the image is freed from any noise.

2.1.3 Extraction from Document Images

Many techniques are used for text extraction in document pictures. this text, will use a really straightforward approach supported quick point's rule. Firstly, this divide the document image into smaller non-overlapping blocks of a set size,then check the density in every block mistreatment quick corner [4] detection technique. The denser blocks were labeled as text blocks and therefore the less dense were the image region or noise region. Then checks the property of the blocks to cluster the blocks in order that the text half will be isolated from the image, then builds the text region and reserve it. This technique is incredibly quick and versatile, it will be wont to discover varied languages, handwriting and even pictures with loads of noise and blur. even if it's a really straightforward program the preciseness of this technique is nearer or beyond 90percent. lastly, this technique helps in additional accurate and fewer advanced detection of text from document pictures.

2.2 COMPONENTS OF AN OCR SYSTEM

A distinctive OCR system consists of assorted parts for OCR systems. OCR consists of the many phases like Scanning of image, Pre-processing, Segmentation, Feature Extraction, Classifications and Recognition, Post process. The task of preprocessing relates to the removal of noise and variation within the image. In scanning step the image is attained and also the image is digitalized. the standard of image depends extremely on the scanner being used. In sensible applications, the scanned pictures aren't excellent there could also be some noise because of some uncalled for details within the image which may cause a stoppage in the detection of the characters within the image. Preprocessing involves removal of noise (applying filters like Gaussian filter, gabor filter etc.) and correct conversion of image like a colored image may be born-again into grey scale or binary image for any process of image. Feature extraction [5]involves recognizing the feature needed. Classifications and Recognition part is that the extraction part of the method. once finishing the OCR method many postprocessing steps area unit necessary betting on the appliance, e.g. tagging the documents with secondary information like author, year, etc.or proof-reading the documents for correcting OCR errors and writing system mistakes.

1. Optical Scanning:In the scanning method the digital image of the document is captured. A scanner is employed to scan the documents. the standard of the document depends extremely on the scanner getting used. So, a scanner with high speed and smart color quality is important for correct process of the image.
2. Location and segmentation:
3. This method locates the places wherever contents area unit gift. the method that determines the constituents [6] of a picture is segmentation. it's essential to find the regions of the document that have knowledge written and distinguish them from noise and footage. for instance, throughout automatic mail- sorting, the address is found and separated from different constituents of the envelope like stamps or logos, before recognition method. Segmentation is that the separation of characters

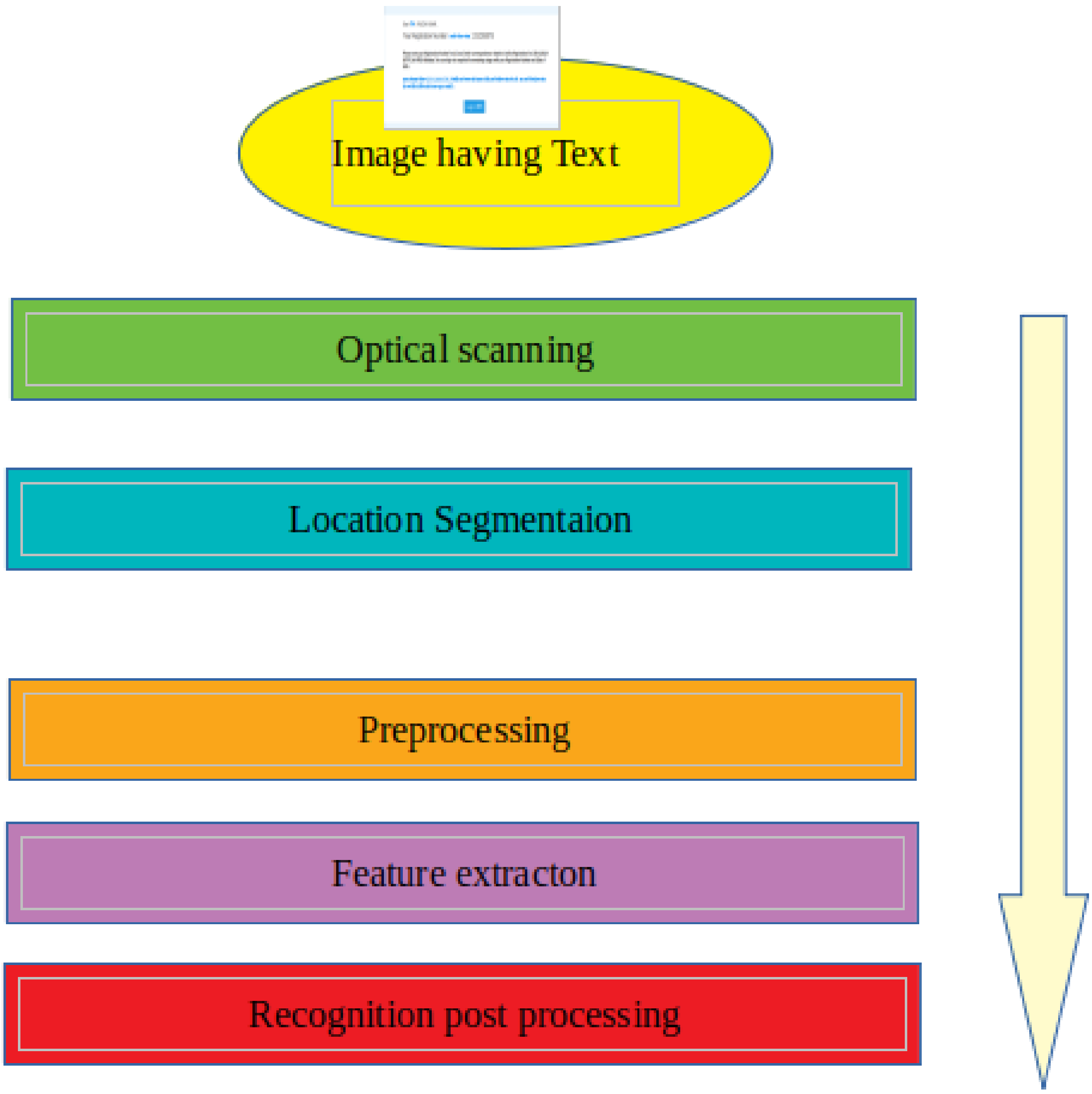

Figure 2.2: Components of an OCR System

or words from image that is performed on text. Most optical character recognition systems segment the words into isolated characters that area unit documented separately. This technique [7] is straightforward to device, however issues happens if the characters' bit or if characters area unit disjointed and contains many elements. the most issues in segmentation could also be divided into four groups:
1. Extraction of touching and disjointed characters.
2. identifying noise from text. Dots and accents could also be mistaken for noise, and the other way around. 3. interpretation graphics or pure mathematics for text.

This results in nontext being sent to recognition.

4. interpretation text for graphics or pure mathematics. during this case the text won't be passed

to the popularity stage. This usually happens if characters area unit connected to graphics.

4. Pre-Processing:The image is scanned and is regenerate into grey scale. The gray scale image perhaps regenerate to binary image. This method is named digitisation [8] of image (Binarization). In sensible applications, a scanner isn't perfect ,the image that's scanned might have some noise. this might flow from to some redundant details gift within the image. The denoised image is created by applying some appropriate strategies. This denoised image is saved for more process. Depending on the resolution on the scanner and also the success of the applied technique for thresholding, the characters might not be dead scanned.
5. Feature extraction: The pre-processed image is the input to the present and every single character within the image is discerned. The image from the extraction stage is matched with all the preloaded characters within the system. Once the matching is completed, the example with the utmost correlative worth is asserted because the character gift within the image. the target of feature extraction is to observe the essential characteristics of the characters, and it's usually accepted that this is often one of the foremost tough issues of pattern recognition. the most effective means of describing a character is by the particular image. The techniques for extraction of such options are usually divided into 3 main teams, wherever the options area unit found from: • The distribution of the points. • Transformations and series expansions. • Structural analysis.
6. Post processings : when feature extraction stage, there may be some unrecognized characters, those characters might get outlined within the post-processing step. Character grouping to create a pregnant text and error detection and correction is finished in this step.

2.3 Scene Text Detection via Stroke Width

2.3.1 Text Detection Algorithm

An overview of our algorithm is depicted in Figure 2.3 . On every input color image, first resize it into 640 × 480 (or 480 × 640) resolution, then MSERs are detected and considered as text region candidates. As a next step,design some simple heuristic rules to remove MSERs which are not text regions. Different from stroke width transform in the work,then propose stroke width generated by distance transform on the skeleton of each CC to eliminate nontext areas. In the final step, group characters into words based on Euclidean distance, orientation and similarities between characters.

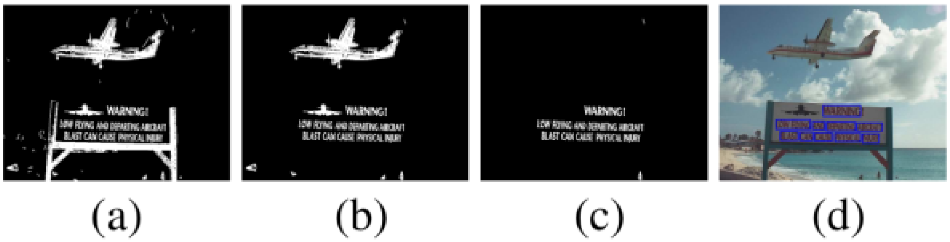

Figure 2.3: Overview of text detection process.

Description of Figure 2.3 (a) Detected MSERs. (b) CCs after geometric filtering. (c)CCs after stroke width extraction. (d) Detected text.

2.3.2 Contrast-enhanced MSER Detection

The concept of MSER introduced by Matas etal. Since a single letter usually shares matching color and its intensity has often quite distinct from background, MSER can locate these text area correctly.MSER has many good properties, such as invariance to affine transformation of image intensities, stability etc., however, it is sensitive to image blur. An example demonstrating this is shown in Figure 2.5. It is obvi- ous that most of characters are blurred and connected, so it is really difficult for us to get true stroke width of every character. In order to overcome this problem, Here propose a novel contranst-enhanced MSER algorithm as follows.

For input scene image I , depend on the experiments that there are large changes in intensity at the boundary be- tween text pixels and background, an intensity image In is obtained as

======> In = (R + B + G)/3 in HSI col- or space. After that, check intensity gradient using

======> In(i+1, j)−In(i−1, j) > T 1 , where T 1 is a threshold, if this condition is met, then update:

======> I C (i ± 1, j) = I C (i ± 1, j) ± T 2 ,

where C R, G, B, parameter T 2 is a predefined threshold. The aim of this procedure is to enhance the contrast between characters and background (Figure 2.5b). Finally, conduct MSER detection on this contrast- enhanced image. Figure 3 (2.5c) illustrates the result of contrast-enhanced MSER detection where every letters in the same word are separated.

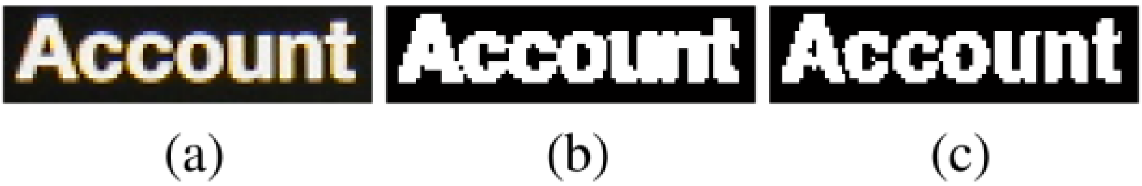

Figure 2.4: Contrast-enhanced MSER Detection

Description of Figure 2.4(a) Original characters. (b) Bare MSER detection. (c) Contrast-enhanced MSER detection.

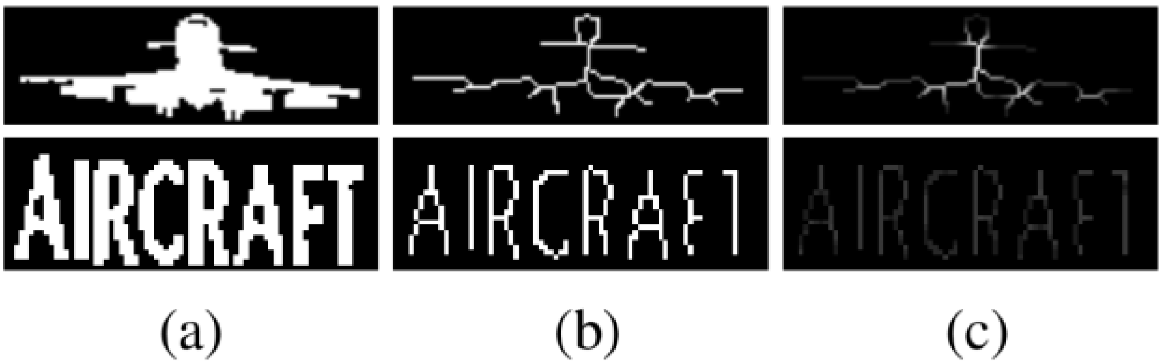

Figure 2.5: Stages for OCR

Description of Figure 2.5(a) Detected MSER of false positive and text. (b) Skeleton map. (c) Skeleton-distance map.

2.3.3 Geometric Filtering

After locating bounding boxes of MSER, design some simple geometric rules to filter out obvious nontext regions. Firstly, by assuming all characters have been separated, limit the aspect ratio of each bounding box between 0.3 and 3. Secondly, text region candidates with low saturation (less than 0.3) or small area(less than 30 pixels) are unlikely to be text regions, thus they should be removed. Thirdly, since text may be surrounded by non-text CCs (e.g., the signboard containing characters is detected in Figure 2.3),reject this kind of false positive by limiting the number of bounding boxes within a particular bounding box to three. For definitions of aspect ratio, saturation and area.

2.3.4 Stroke Width Extraction

Stroke width defined as the length of the straight line or lime segment from a text edge pixel to another along its gradient direction. Basic motivation of stroke width extraction algorithm is a stroke width almost remains the same in a single character,whereas, there is significant change in stroke width in non-text regions as the result of their irregularity or dissimilarity. There are many researches exploited this property which calculate , stroke width from a stroke area to another along gradient area. Since skeleton is the effective tool to represent, structure of a region, inspired by which uses skeleton to analyze text string straightness,This take advantage of skeleton to extract stroke width. The first step of stroke width extraction is to get skeletons of MSERs remained. On each foreground pixel on skeleton, distance transform is applied to compute Euclidean distance from this pixel to the nearest boundary of the corresponding MSER. Then obtain a skeleton-distance map. This process is depicted in illustrates a non-text MSER and text MSER from Figure 2.3 (a), and their correspond- ing skeleton and skeleton-distance map are shown in Figure 2.3 (b) and Figure 2.3 (c) respectively.Variance on skeleton-distance map of each CC computed to measure the difference between text regions and false positives. values of variance obtained from Figure 2.3 (c). Note that text characters have much smaller variances compared with the false positive. Based on this property remove CCs with large variances. It can be seen in Figure 2.3 (c) that some false positives are eliminated after this procedure.

Chapter 3

System Design & Implementation

3.1 System Analysis

In distinction to a lot of classical OCR issues, wherever the characters are usually monotone on mounted backgrounds, character recognition in scene pictures is probably much more complicated because of the numerous attainable variations in background, lighting, texture and font. As a result, building complete systems for these eventualities needs us to create representations that account for all of those sorts of variations. Indeed, important effort has gone into making such systems, with high performers desegregation dozens of smartly combined features and process stages.Recent add machine learning, however, has sought to create algorithms that may learn higher level representations of information mechanically for many tasks. Such systems may well be notably valuable wherever specialised options are needed however not simply created by hand.Another potential strength of those approaches is that it will simply generate massive numbers of options that change higher performance to be achieved by classification algorithms. during this project, here appling MSER , Basic Geometric Properties and Stroke dimension Variation to see to what extent these algorithms could be helpful in scene text detection and character recognition. MSER , Basic Geometric Properties and Stroke dimension Variation algorithms have enjoyed a string of successes in different fields (for instance, achieving high performance in visual recognition and audio recognition). sadly, one caveat is that these systems have usually been too computationally costly, particularly for application to massive images. to use these algorithms to scene text applications, this can therefore use a a lot of scalable feature learning system. Specifically, this use a variant of K-means agglomeration to train a bank of options, equally to the system in. Armed with this tool, this will produce results showing the impact on recognition performance as increase the amount of learned options. Our results can show that it's attainable to try and do quite well just by learning several options from the info. Our approach contrasts with a lot of previous add scene text applications, as none of the options used here are expressly designed for the application at hand. Indeed, the system follows closely the one planned in.

3.2 Design

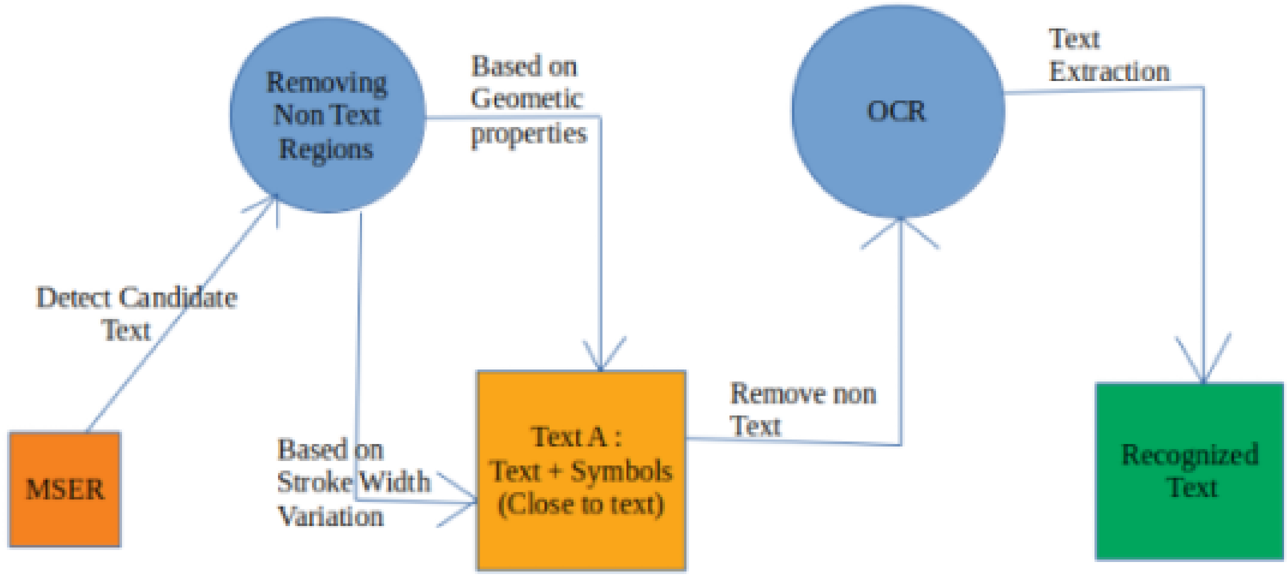

Figure 3.1: Data Flow Diagram (Level 0)

Maximally Stable Extremal Regions (MSER) as our basic letter candidates. Despite their favorable properties, MSER are reportable to be sensitive to image blur.To allow for detection tiny letters in pictures of restricted resolution, the complimentary properties for canny edges MSER combined in edge enhanced MSER. Further propose to come up with the stroke dimension remodel image of those regions victimisation the gap transform to expeditiously acquire additional reliable results. The geometric likewise as stroke width info are then applied to perform filtering and pairing . Finally, letters are clustered into lines and extra checks performed to eliminate false positives. The overall method of the text detection offers higher result . Compared to different text detection approaches, rule offers the subsequent major blessings. First, the edge enhanced MSER detected within the question image are often wont to extract feature descriptors for visual search. Therefore our text detection are often combined with visual search systems without more process load to observe interest regions. Further, our system provides a reliable binarization for the detected text, which might be passed to OCR for text recognition. Finally, the planned rule is straightforward and economical.

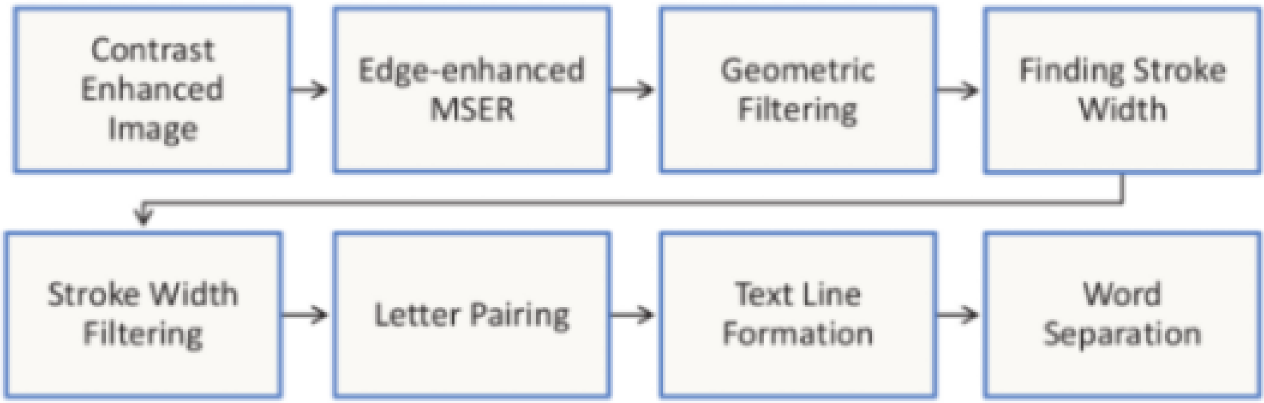

Figure 3.2: Data Flow Diagram (Level 1)

Description of Figure 3.2 System Flow Chart of MSER

Description of Figure 3.3 System Flow Chart

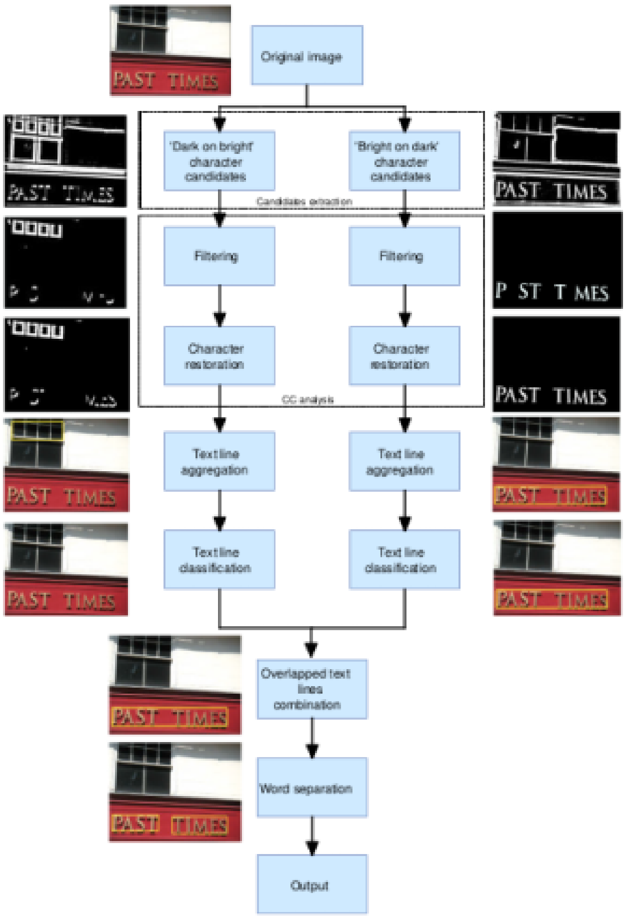

Figure 3.3: Flow Chart

3.3 Implementation

This project justify you ways to discover regions in a picture that contain text. This is a common task performed on unstructured scenes. Unstructured scenes are pictures that contain undetermined or random eventualities. as an example,this discover and acknowledge text automatically from captured video to alert a driver a couple of road sign. this can be totally different than structured scenes, that contain notable eventualities wherever the position of text is thought beforehand.
Segmenting text from unstructured scene greatly helps with extra tasks such as optical character recognition (OCR). The automatic text detection formula during this project detects an oversized range of text region candidates and increasingly removes those less probably to contain text.

3.4 Detect Candidate Text Regions Using MSER

The MSER feature detector works well for locating text regions [9].It works well for text because the consistent color and high distinction of text ends up in stable intensity profiles.

Using the detectMSERFeatures perform to search out all the regions among the image and ploting these results. Notice that there square measure several non-text regions detected aboard the text.

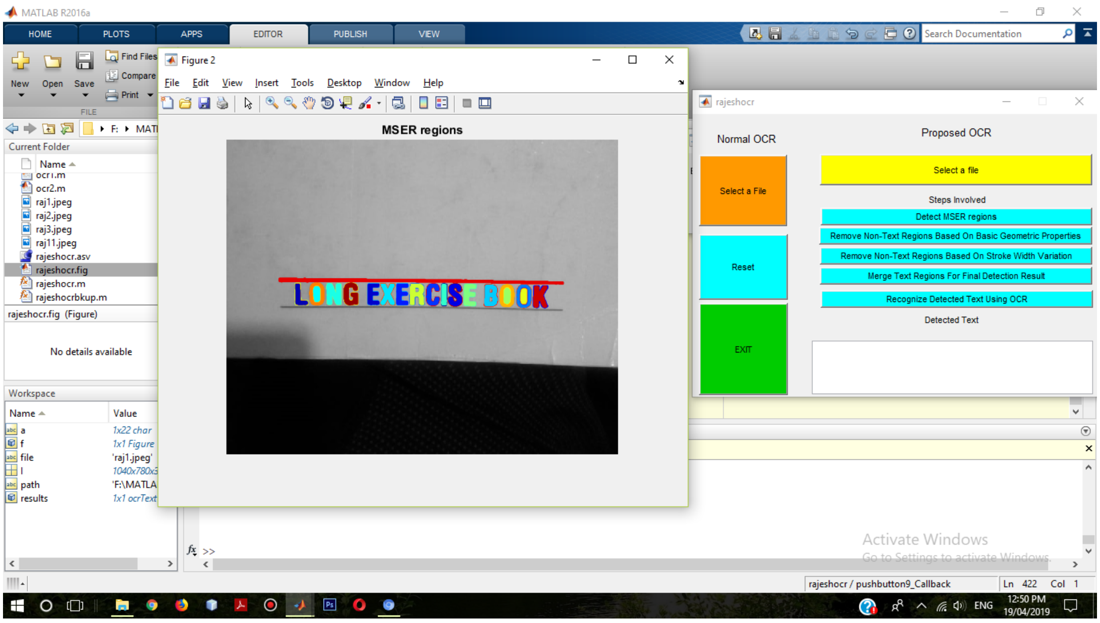

Figure 3.4: MSER Regions

3.5 Remove Non-Text Regions Based On Basic Geometric Properties

Although the MSER algorithmic rule picks out most of the text, it conjointly detects several alternative stable regions within the image that don't seem to be text. this may use a rule-based approach to remove non-text regions. as an example, geometric properties of text may be accustomed filter out non-text regions exploitation easy thresholds. instead, employing a machine learning approach to coach a text vs. non-text classifier. Typically, a mix of the 2 approaches produces higher results [10]. This uses a straightforward rule-based approach to filter non-text regions supported geometric properties.

There are many geometric properties that are smart for discriminating between text and non-text regions [11] [12], including:

1 . Aspect ratio
2. Eccentricity
3. Euler number
4. Extent
5. Solidity

Using the regionprops to live some of those properties so take away regions based on their property values.

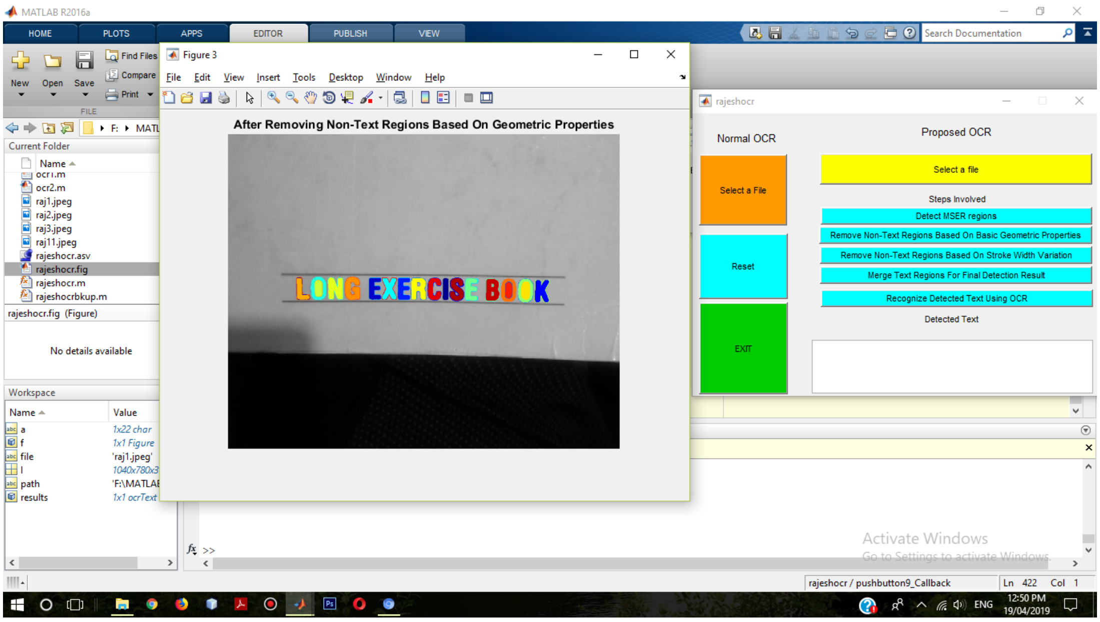

Figure 3.5: After Removing Non-Text Regions Based On Basic Geometric Properties

3.6 Remove Non-Text Regions Based On Stroke Width Variation

Another common metric accustomed discriminate between text and non-text is stroke dimension. Stroke dimension could be a live of the dimension of the curves and features that structure a personality.Text regions tend to own very little stroke dimension variation, furthermore non-text area tend to have big variations.

To help perceive however the stroke dimension may be accustomed take away non-text regions, estimate the stroke dimension of 1 of the detected MSER regions. you'll be able to try this by employing a distance rework and binary cutting operation [12].

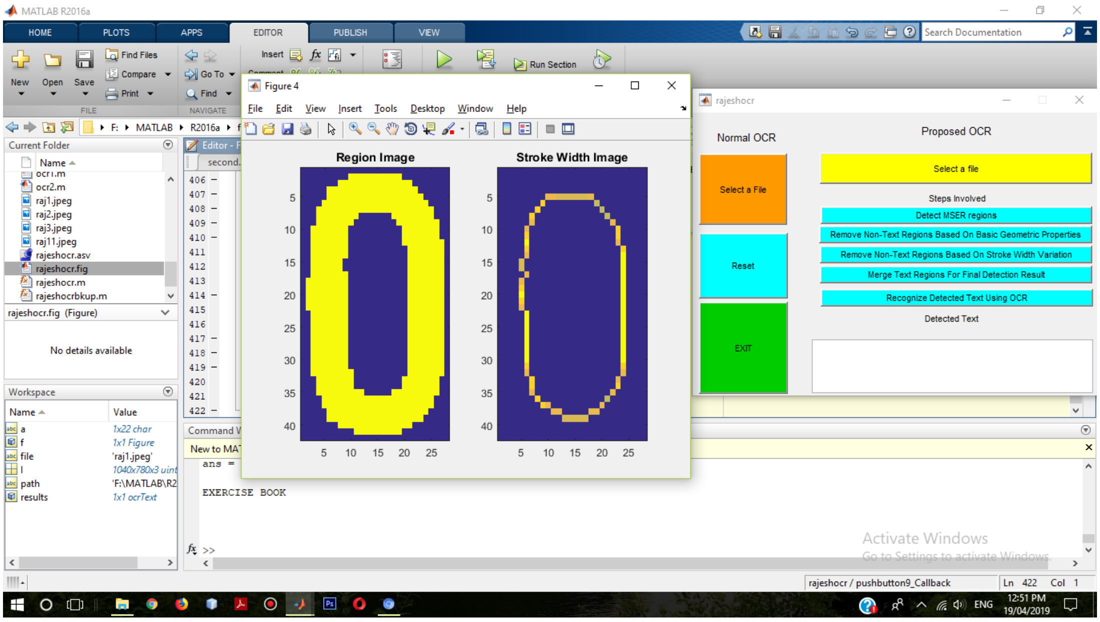

Figure 3.6: Region Image vs Stroke Width image

In the pictures shown here, notice however the stroke dimension(Width) image has little or no variation over most of the region. this means that the region is a lot of seemingly to be a text region as a result of the lines and curves that conjure the region all have similar widths, which could be a common characteristic of human decipherable text.

In order to use stroke dimension variation to get rid of non-text regions employing a threshold price, the variation over the complete region should be quantified into one metric .

Then, a threshold may be applied to get rid of the non-text regions. Note that this threshold value might need standardization for pictures with totally different font designs. The procedure shown higher than should be applied severally to every detected MSER region. The following for-loop processes all the regions, and so shows the results of removing the nontext regions exploitation stroke breadth variation.

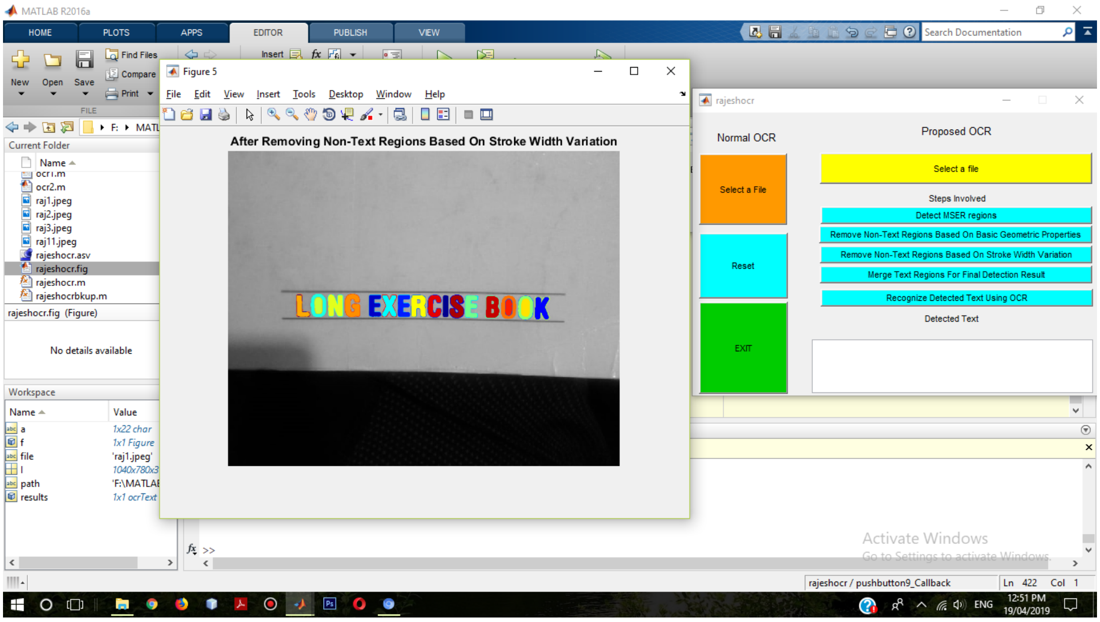

Figure 3.7: After Removing Non-Text Regions Based On Stroke Width Variation

3.7 Merge Text Regions For Final Detection Result

At this time, all the detection results square measure composed of individual text characters. To use these results for recognition tasks, like OCR, the individual text characters should be integrated into words or text lines. this permits recognition of the particular words in image, that carry a lot of important data than simply the individual characters. For example, recognizing the string ’EXIT’ vs. the set of individual characters ’X’,’E’,’T’,’I’, where the which means of the word is lost while not the proper ordering.

One approach for merging individual text regions into words or text lines is to 1st find neighboring text regions so type a bounding box around these regions. To find neighboring area, explore the bounding boxes computed earlier with regionprops. This makes the bounding boxes of neighboring text regions overlap specified text regions that square measure a part of an equivalent word or text line type a sequence of overlapping bounding boxes.

Now, the overlapping bounding boxes is integrated along to create one bounding box around individual words or text lines. To do this, reckon the overlap ratio between all bounding box pairs. This quantifies the space between all pairs of text regions in order that it’s potential to seek out teams of neighboring text regions by wanting for non-zero overlap ratios. Once the pair-wise overlap ratios square measure computed, use a graph to find all the text regions ”connected” by a non-zero overlap quantitative relation.

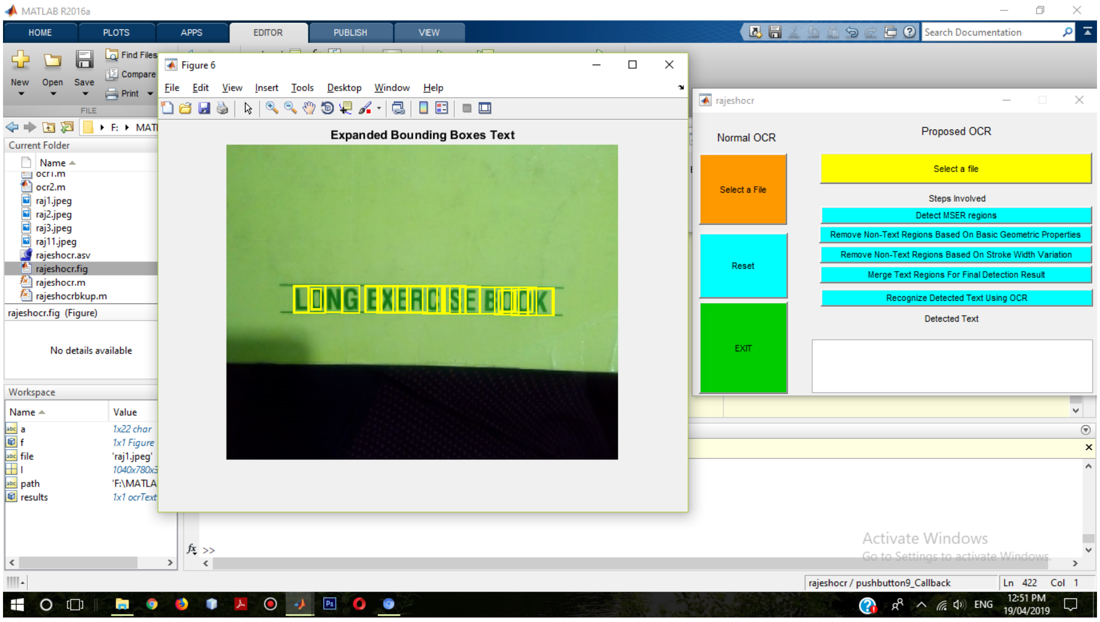

Figure 3.8: Expanded Bounding Boxes Text

Using the bboxOverlapRatio operate to calculate the pair-wise overlap ratios for all the enlarged bounding boxes, then use graph to seek out all the connected regions.

The output of conncomp are indices to the connected text regions to that every bounding box belongs. Utilize these indices to match or merge multiple neighboring bounding boxes into a single bounding box by computing the minimum and most of the individual bounding boxes that form up every connected part.

Finally, before showing the ultimate detection results, suppress false text detections by removing bounding boxes created from only 1 text region. This removes isolated regions that are unlikely to be actual text providing text is sometimes found in teams (words and sentences).

3.8 Recognize Detected Text Using OCR

After sleuthing the text regions, use the ocr perform to acknowledge the text at intervals every bounding box. Note that while not initial finding the text regions, the output of

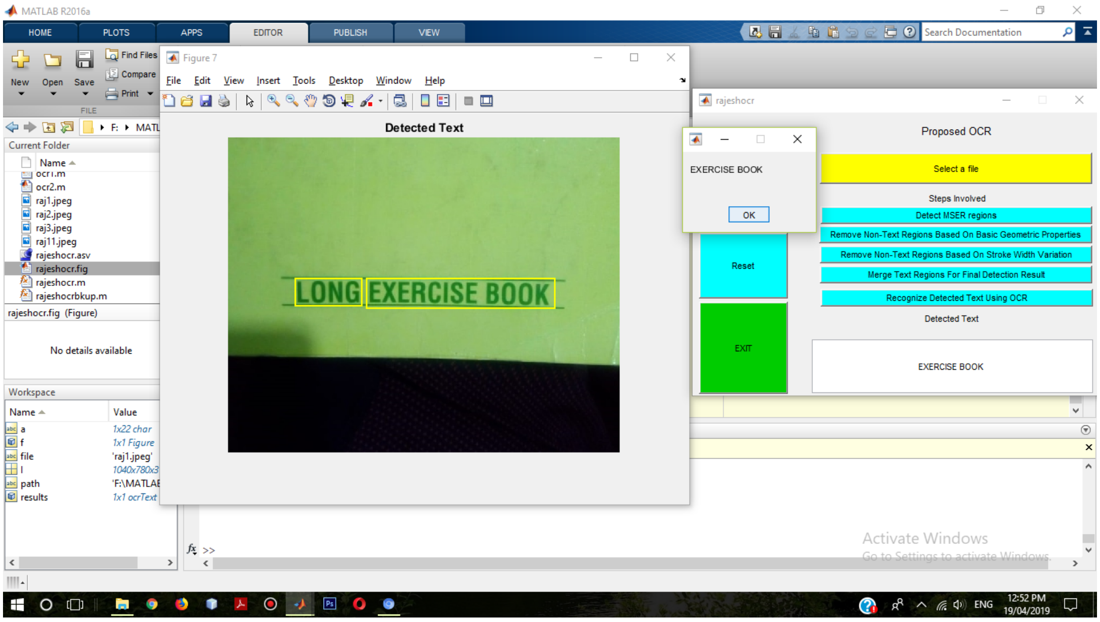

Figure 3.9: Detected Text

the ocr perform would be significantly additional screaming.
Algorithm showed the way to discover text in a picture victimization the MSER feature detector to initial notice candidate text regions, so it delineate the way to use geometric measurements to get rid of all the non text regions. This procedure may be a smart start line for developing additional sturdy text detection algorithms.

3.9 Result

Table 3.1: Accuracy Comparision

Comparision(Accuracy of Text extraction on Natural Images)		
No. of Images	Existing System Model(in percentage)	Proposed Model(in percentage)
Image 1	85.0	88.9
Image 2	77.4	80.2
Image 3	88.0	90.2
Image 4	66.2	76.4
Image 5	76.8	82.3
Image 6	90.2	96.2
Image 7	83.1	92.1
Image 8	65.4	73.6
Image 9	76.4	79.6
Image 10	66.8	72.9
Image 11	55.7	58.9
Image 12	89.4	91.9
Image 13	93.2	95.9
Image 14	87.8	98.9
Image 15	76.0	82.9
Image 16	55.3	76.9
Image 17	78.4	85.9
Image 18	89.4	93.2
Image 19	90.4	92.4
Image 20	75.3	91.8
Total	77.81	85.055

Here percentage is calculated based on number of text extracted to the number of text present in the image.for an example if there are 25 text in an natural image and text extracted by first is 19 and 22 by other one than the percentage for first and second is 76 and 88 percent respectivally .

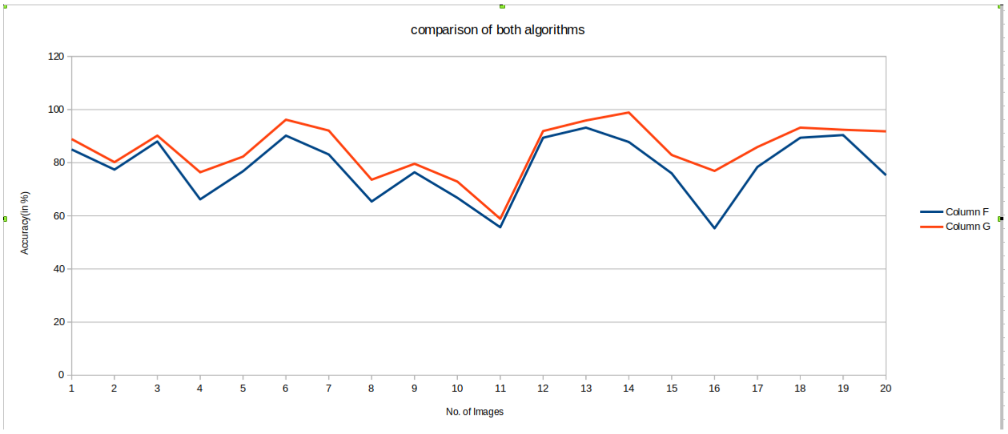

Figure 3.10: Comparision of Both Algorithm(Blue-Existing one , Red - Proposed one)

1 . Based on Experiment done on 20 different images by both algorithm ,Text extraction from document images by OCR(existing system) algorithm and proposed algorithm , It has been concluded that proposed algorithm produces better result.

2 . As compared to existing system OCR for extracting Text from Document Images which gives upto 77.81 precent accuracy ,Proposed system gives upto 85.055 percent.

3 . As compared with Normal OCR technique used in existing system used for enhancement in this project , it has following steps :

(a)Scanning the Images ,(b)Pre-processing(Conversion and noise removal,(c)Feature Extraction,(d)Classification and recognition,(e)Post Processing(Proof reading,correction,spell checker)
which produces accuracy upto 77.81 percent but using proposed algorithm can enhanced it upto 85.055 percent because of MSER and Removing non text by Basic geometric properties and stroke width variation.

4 . Finally it has been concluded that proposed algorithms produces better result with better accuracy as corresponding to existing one(used in this project).

Chapter 4

Conclusion

To extract text from images this project presents a novel approach based on MSER that produced a text detection and recognition system in natural images through detecting candidate text regions using MSER first then removing non text regions based on basic geomatric properties as Aspect Ratio , Eccentricity , Euler Number , Extent , Solidity then removing Non Text Regions Based on Stroke Width Variation . Finally merge regions for final detection result , Then as normal detect text using ocr . Novelly this project make a good approach to detect a text region recognition using MSER ,this could be very simple to obtain an accurate text extraction at low cost and without using a lot about parameters. Finally, this method seems also to be very efficient in extracting more complex layouts such as paragraphs, and lines.

4.1 Future direction

More precise Font Independent OCR:
Development of OCR considering the multiple font vogue has to be developed within the future.

OCR for all Indian Languages:
Development of OCR for languages other than English needs to be reserched on and developed in the future.

Cursive Characters OCR:
There is heavy demand for an OCR system which recognizes handwritten cursive scripts. This avoids keyboard typing and font coding for the image. This method helps in detecting handwritten characters with a precision of about 90 percent.

Language Converter through OCR:
Once detect languages this can can develop a converter to convert sentences from one language to another through a conversion and translation scheme.

Speech recognition from OCR:
Speech recognition is one amongst the foremost necessary application today. The recognized written or written OCR might be recorded and speech output might be generated. This would facilitate the blind to send and receive info.

Speech to text converter through OCR:
Speech recognition is one of the most important application today. The recognized speech could be recorded and output of text could be generated.

Appendix A

Screenshot and Description of the Implemented System

A.1 Screenshots Part - I

A.1.1 Original Pic

Figure A.1: Original Pic of Entrance(Faculty Block II)

A.1.2 Basic GUI

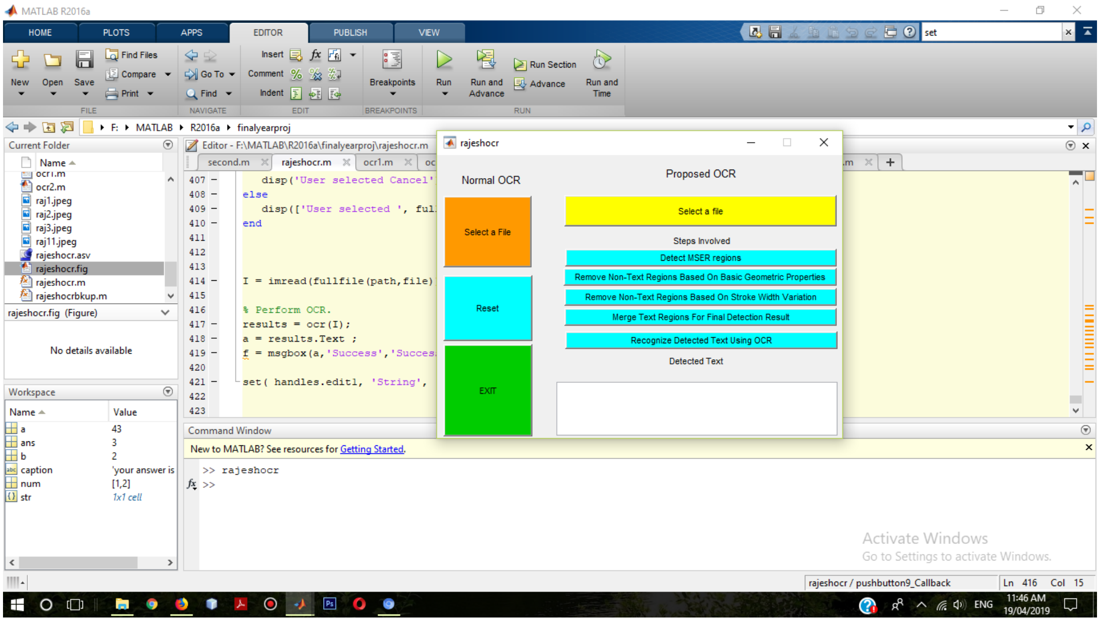

Figure A.2: Basic GUI design of RajeshOCR

A.1.3 Detecting Candidate Text Regions

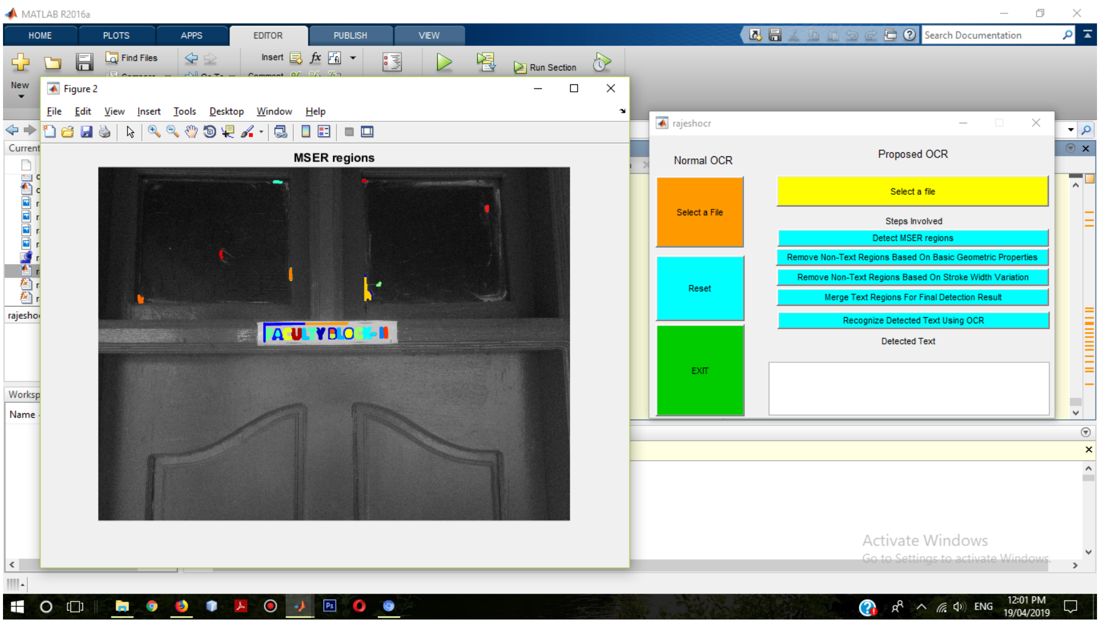

Figure A.3: Detecting Candidate Text Regions Using MSER

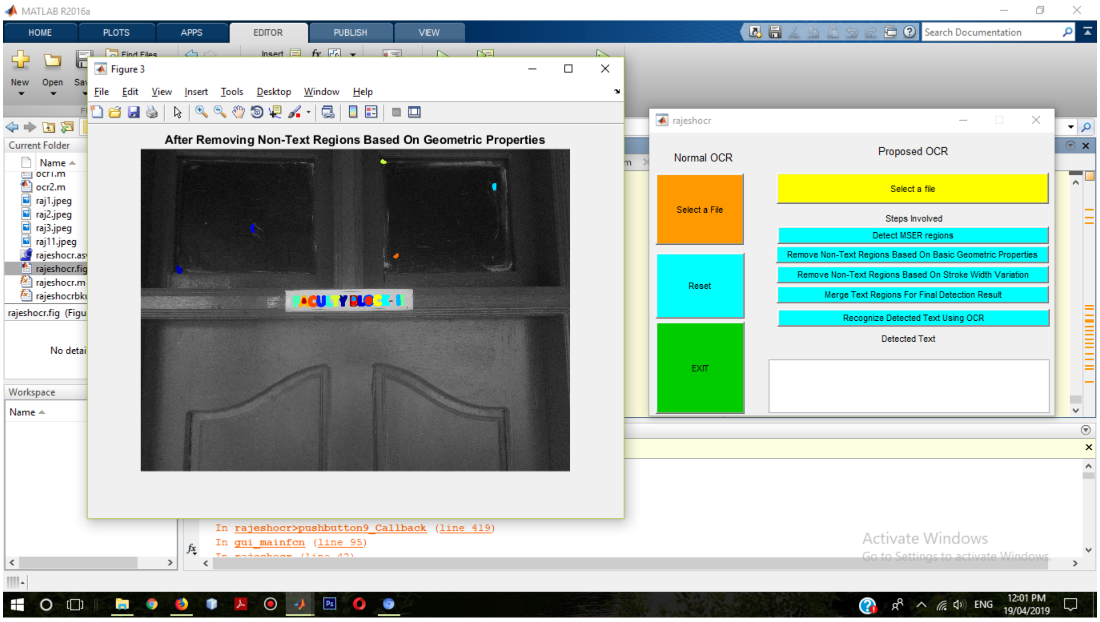

Figure A.4: Removing Non-Text Regions Based On Basic Geometric Properties

A.1.4 Removing Non-Text Regions

A.1.5 Removing Non-Text Regions

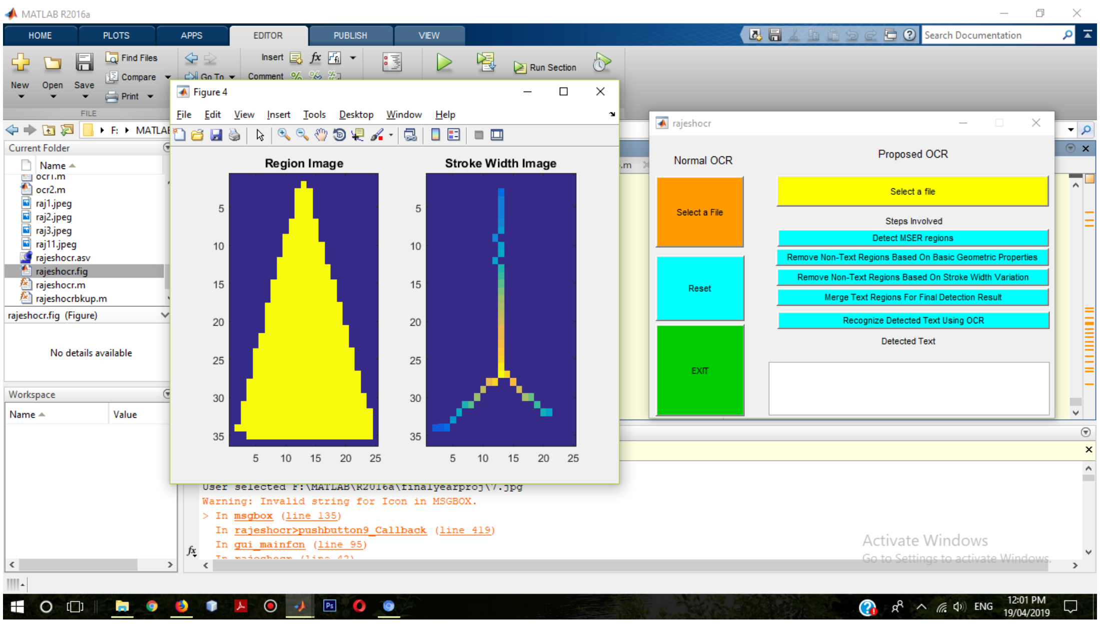

Figure A.5: Removing Non-Text Regions Based On Stroke Width Variation

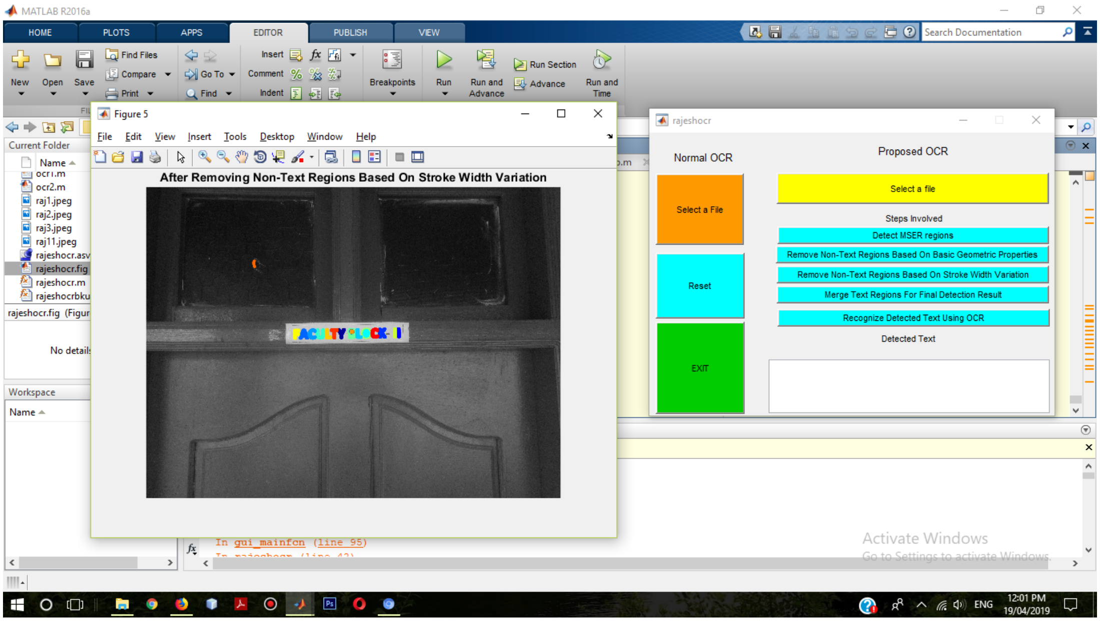

Figure A.6: Merging Text Regions For Final Detection Result

A.1.6 Merging Text Regions

A.1.7 Recognize Detected Text

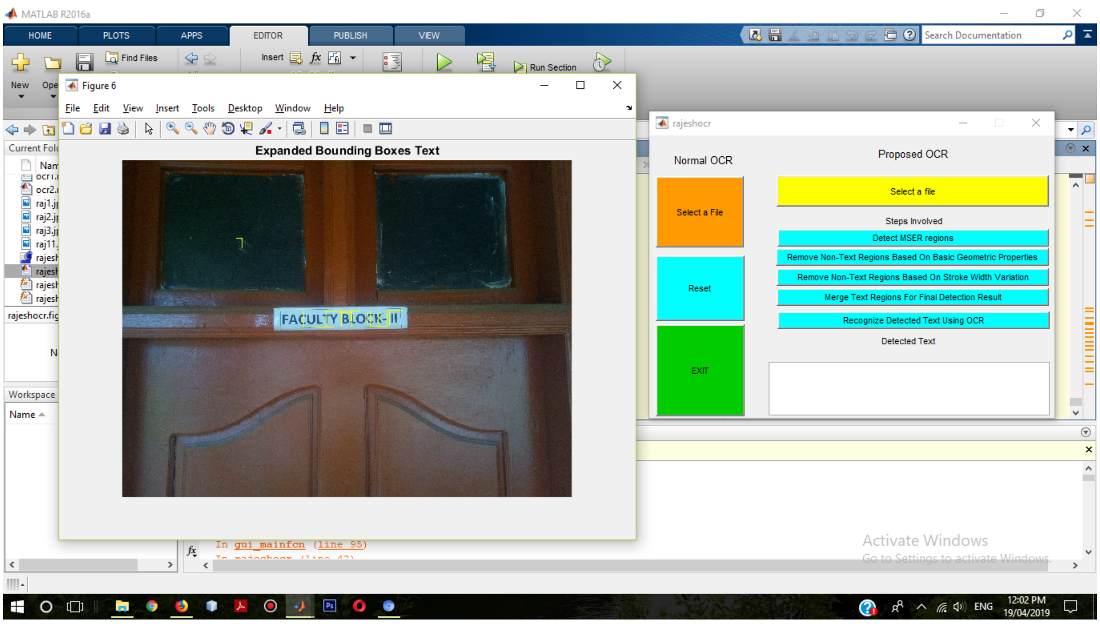

Figure A.7: Recognize Detected Text Using OCR

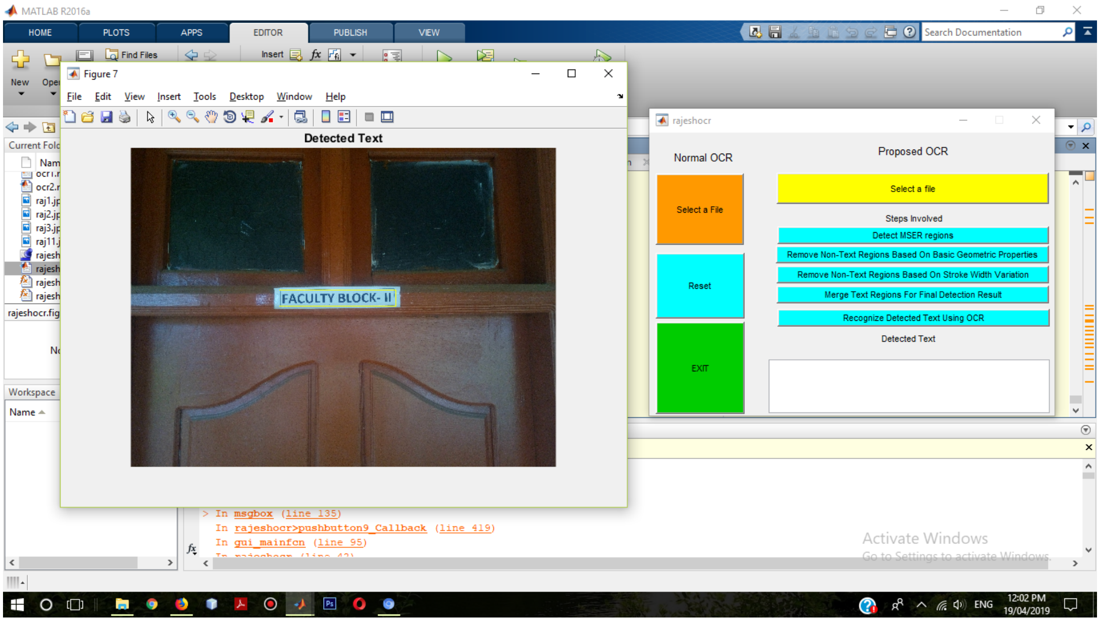

Figure A.8: Combining all Text Area for Extraction of Text

A.1.8 Recognizing the Text area

A.1.9 Extracted Text

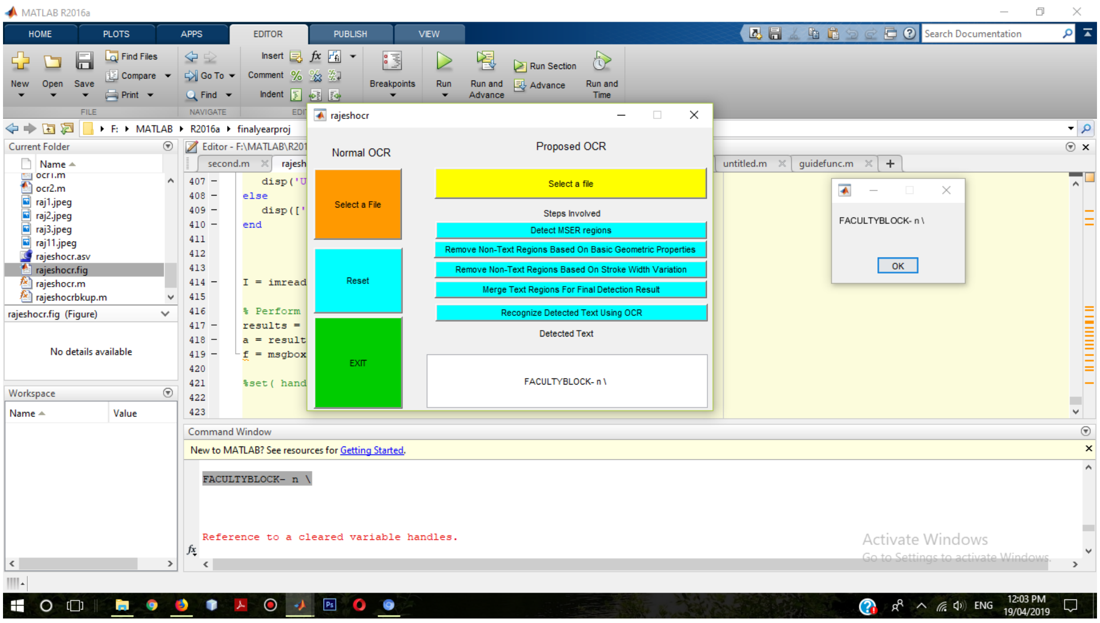

Figure A.9: Extracted Text from Natural Image and Showed in Text Area

A.2 Screenshots Part - II

A.2.1 GUI RajeshOCR

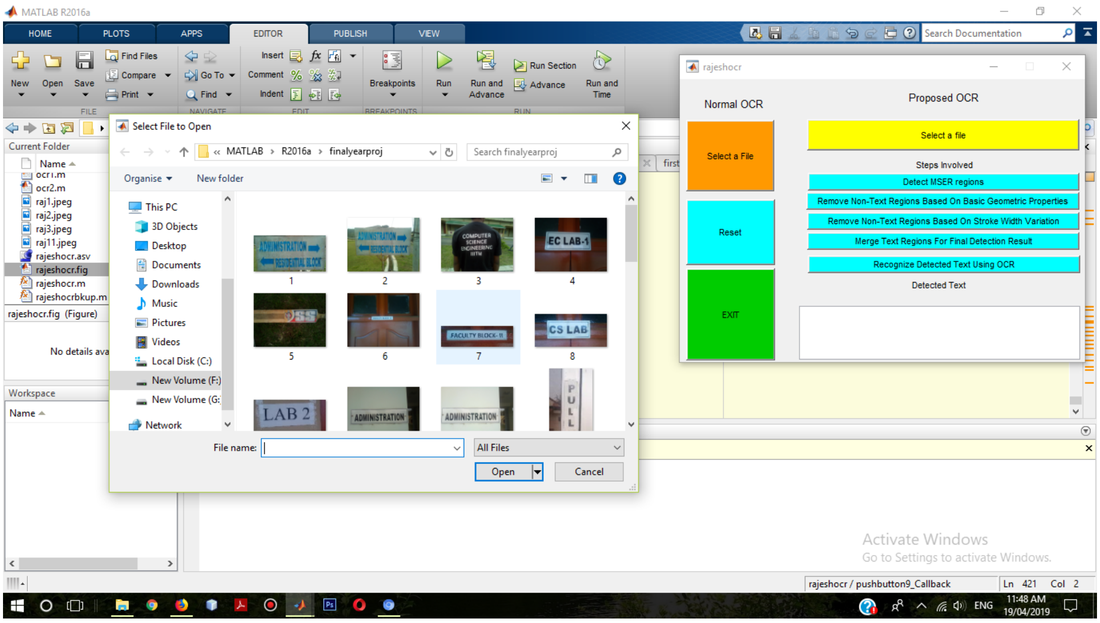

Figure A.10: GUI Design of RajeshOCR

A.2.2 Original Pic

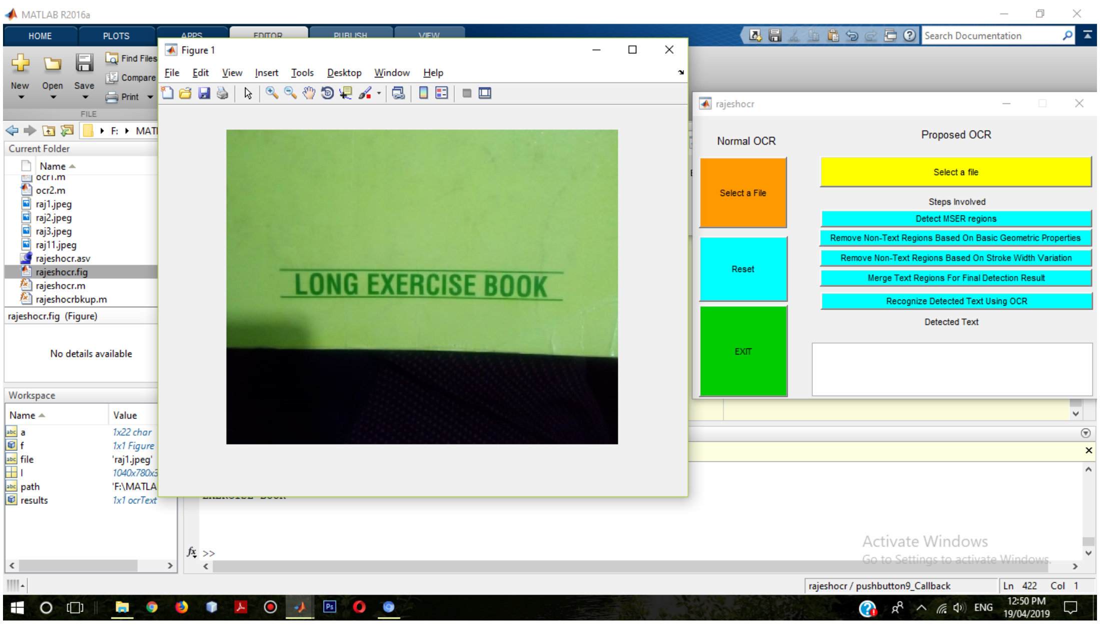

Figure A.11: Original pic of a notebook CoverPage

A.2.3 Detecting Candidate Text Regions

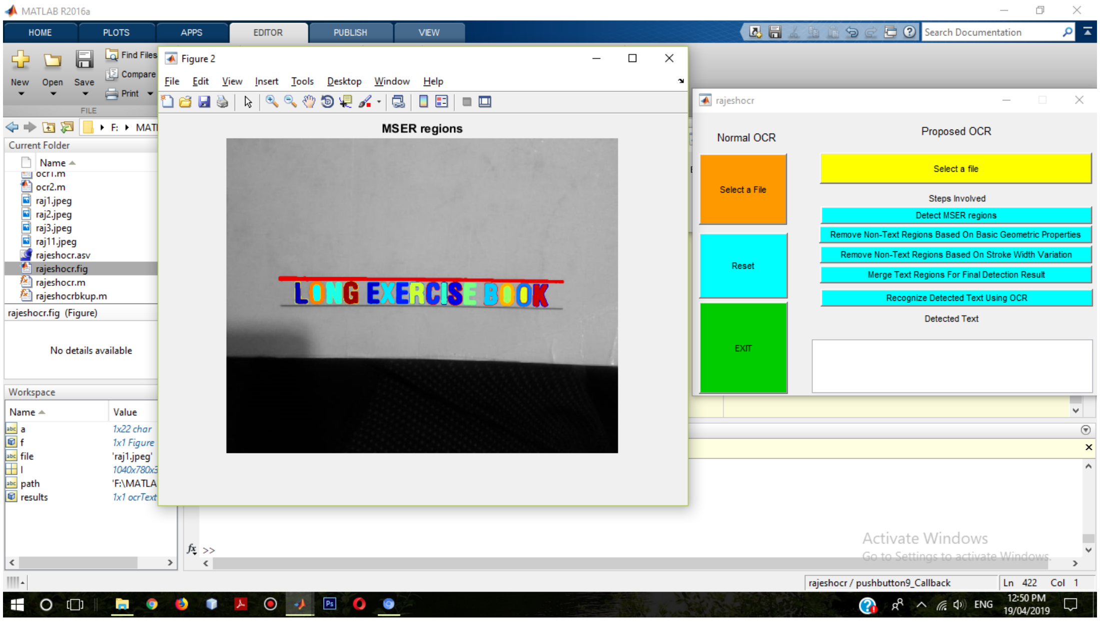

Figure A.12: Detecting Candidate Text Regions Using MSER

A.2.4 Removing Non-Text Regions

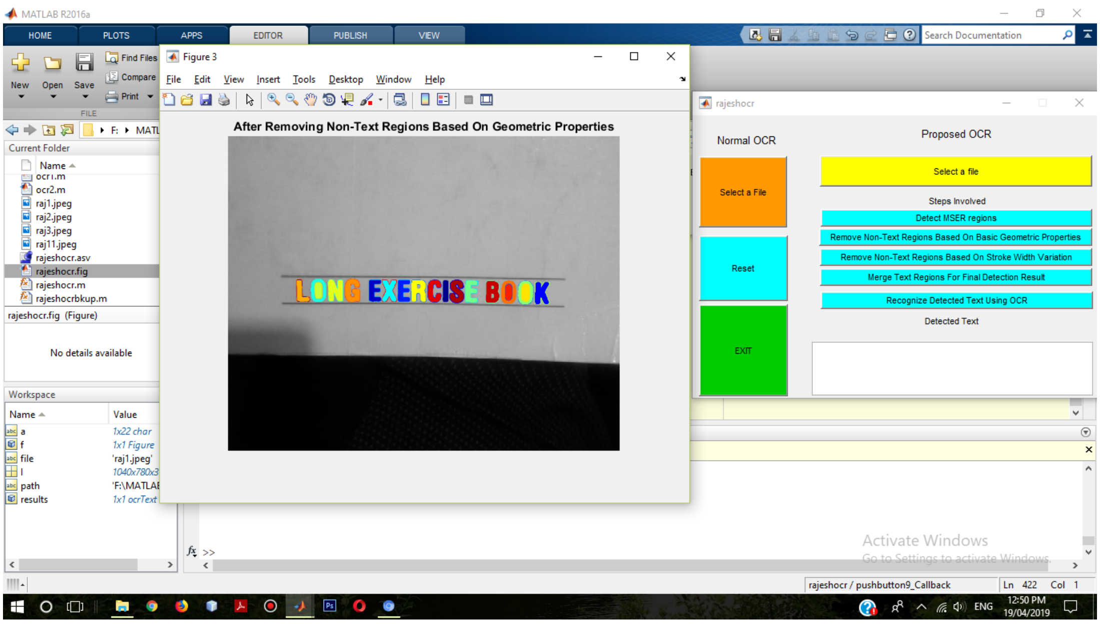

Figure A.13: Removing Non-Text Regions Based On Basic Geometric Properties

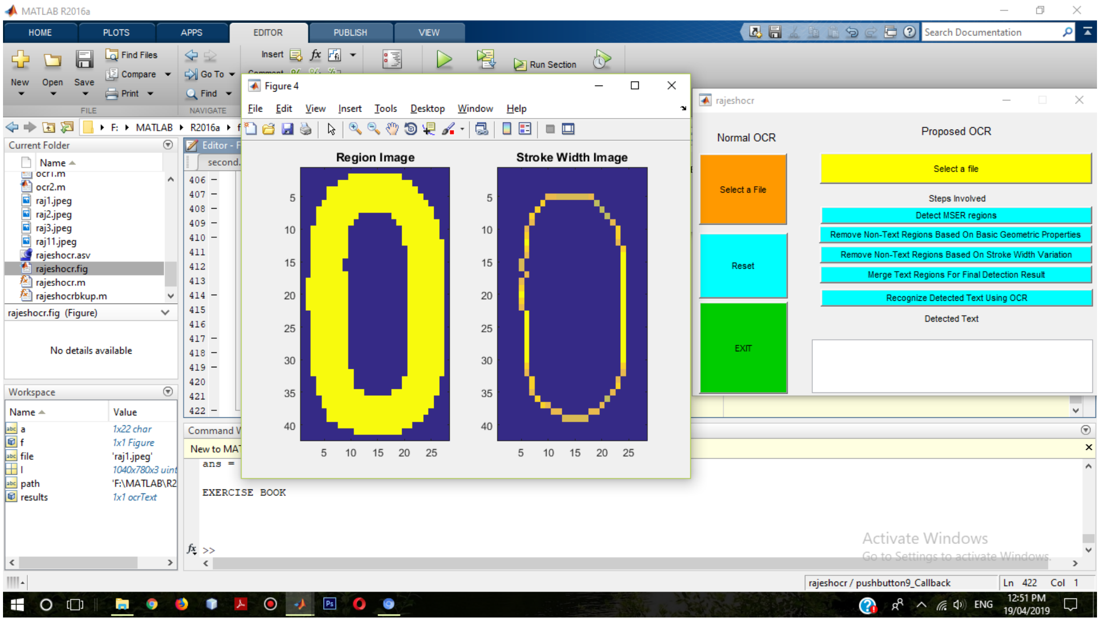

Figure A.14: Removing Non-Text Regions Based On Stroke Width Variation

A.2.5 Removing Non-Text Regions

A.2.6 Merging Text Regions

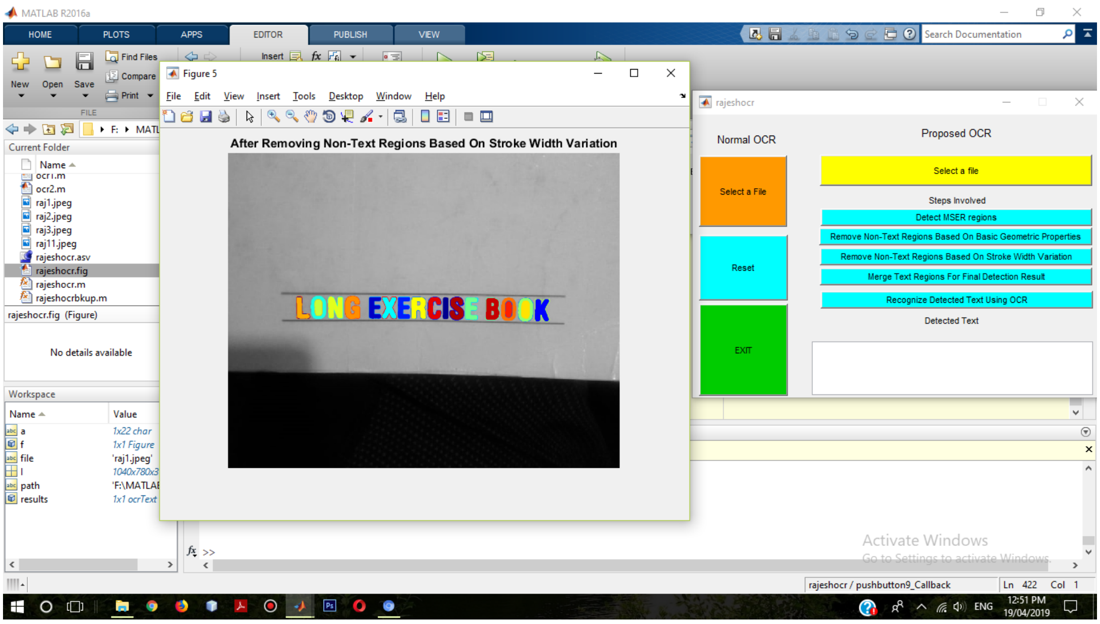

Figure A.15: Merging Text Regions For Final Detection Result

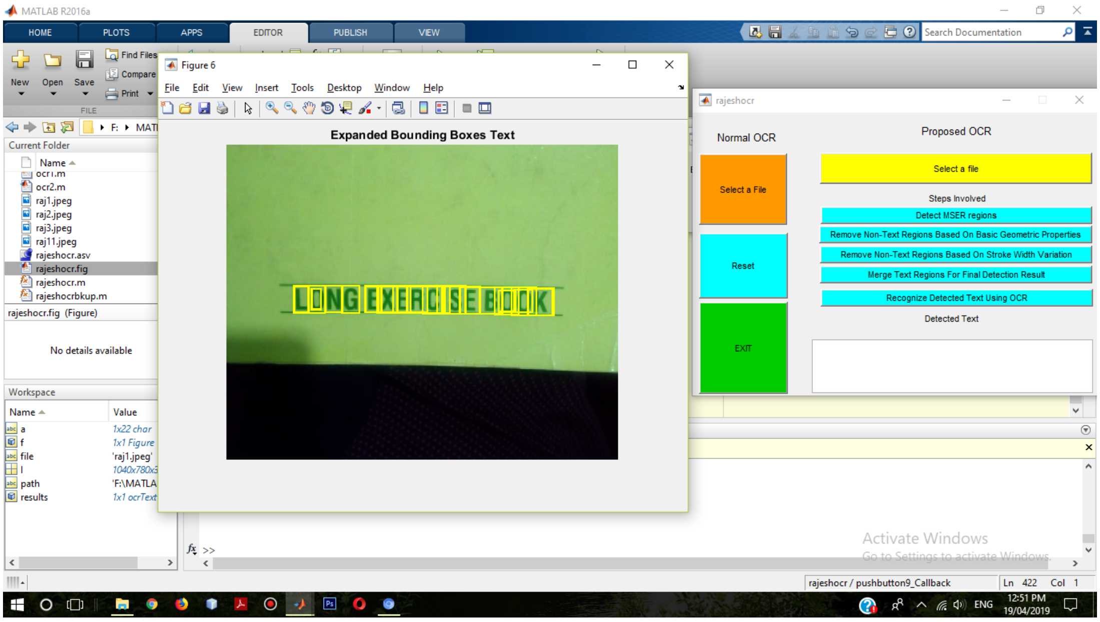

Figure A.16: Merging Text Regions For Final Detection Result and Recognize Detected Text Using OCR

A.2.7 Merging Text Regions and Recognize Detected Text

A.2.8 Recognizing the Text area

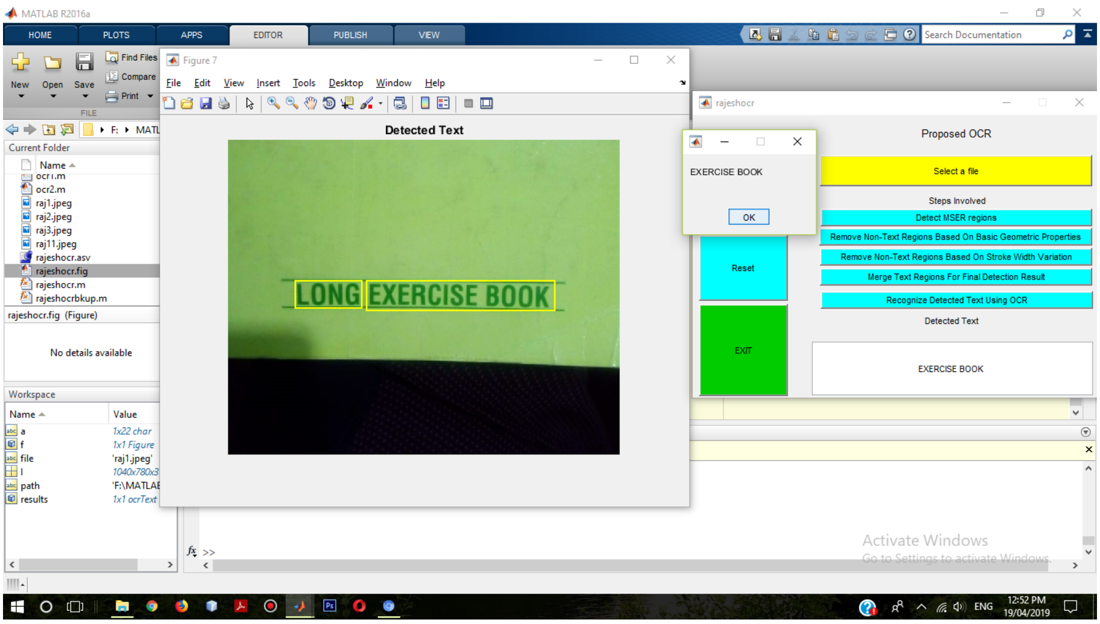

Figure A.17: Combining all Recognized Text and Showed in Text Area

Appendix B

User manual

B.1 Introduction

In my system there is five steps working together to find the better results ,this project try to implement better algorithm for Text recognition and extraction , first Detecting Candidate Text Regions Using MSER then Removing Non-Text Regions Based On Basic Geometric Properties as they are
1 . Aspect ratio
2. Eccentricity
3. Euler number
4. Extent
5. Solidity
and then remove Non-Text Regions Based On Stroke Width Variation and after this Merge Text Regions For Final Detection Result and finally Recognize Detected Text Using OCR .

For this all Procedural flow a GUI with Five Buttons Tabs has been prepared for ease of users to understand it, which looks beautiful and helpfull.

B.2 Step to install your implemented system

You can use GUI or source code by running it over Different Plateforms as Windows , Ubuntu , MacOS and others which supports Matlab environments and there Dependencies.

Bibliography

[1] A. Shahab, F. Shafait, and A. Dengel, ``Icdar 2011 robust reading competition challenge 2: Reading text in scene images,'' in *2011 international conference on document analysis and recognition.* IEEE, 2011, pp. 1491--1496.

[2] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques.* Morgan Kaufmann, 2016.

[3] X. Chen, J. Yang, J. Zhang, and A. Waibel, ``Automatic detection and recognition of signs from natural scenes,'' *IEEE Transactions on image processing*, vol. 13, no. 1, pp. 87--99, 2004.

[4] L. Neumann and J. Matas, ``A method for text localization and recognition in real-world images,'' in *Asian Conference on Computer Vision.* Springer, 2010, pp. 770--783.

[5] J. Gao and J. Yang, ``An adaptive algorithm for text detection from natural scenes,'' in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2. IEEE, 2001, pp. II--II.

[6] C. Li, X. Ding, and Y. Wu, ``Automatic text location in natural scene images,'' in *Proceedings of sixth international conference on document analysis and recognition.* IEEE, 2001, pp. 1069--1073.

[7] Q. Ye, J. Jiao, J. Huang, and H. Yu, ``Text detection and restoration in natural scene images,'' *Journal of Visual Communication and Image Representation*, vol. 18, no. 6, pp. 504--513, 2007.

[8] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, ``Reading digits in natural images with unsupervised feature learning,'' 2011.

[9] H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, and B. Girod, ``Robust text detection in natural images with edge-enhanced maximally stable extremal regions,'' in *2011 18th IEEE International Conference on Image Processing.* IEEE, 2011, pp. 2609--2612.

[10] L. Neumann and J. Matas, ``Real-time scene text localization and recognition,'' in *2012 IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, 2012, pp. 3538--3545.

[11] A. Gonzalez, L. M. Bergasa, J. J. Yebes, and S. Bronte, ``Text location in complex images,'' in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. IEEE, 2012, pp. 617--620.

[12] Y. Li and H. Lu, ``Scene text detection via stroke width,'' in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. IEEE, 2012, pp. 681--684.