# B. TECH. PROJECT REPORT

On

# BOOK READING SYSTEM FOR BLIND PEOPLE

BY

**AKSHAY JOSHI**

**MUHAMMAD TALHA ANSARI**

**DISCIPLINE OF ELECTRICAL ENGINEERING**

**INDIAN INSTITUTE OF TECHNOLOGY INDORE**

Dec 2017

# Book Reading System for Blind People

**A PROJECT REPORT**

Submitted in partial fulfillment of the
requirements for the award of the degrees

***of***
**BACHELOR OF TECHNOLOGY**
***in***

**ELECTRICAL ENGINEERING**

*Submitted by:*
**Akshay Joshi (140002003)**
**Muhammad Talha Ansari (140002019)**

Guided by:
**Dr. Vivek Kanhangad**
**Associate Professor, IIT Indore**

**INDIAN INSTITUTE OF TECHNOLOGY INDORE**
**Dec 2017**

# CANDIDATES DECLARATION

We hereby declare that the project entitled **Book Reading System for Blind People** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in Electrical Engineering completed under the supervision of Dr. Vivek Kanhangad, Associate Professor, Discipline of Electrical Engineering, IIT Indore is an authentic work.

Further, we declare that we have not submitted this work for the award of any other degree elsewhere.

**Akshay Joshi**

**Muhammad Talha Ansari**

---

# CERTIFICATE by BTP Guide

It is certified that the above statement made by the students is correct to the best of my knowledge.

**Dr. Vivek Kanhangad**
**Associate Proffessor**
**IIT Indore**

# Preface

This report on “ Book Reading System for Blind People ” is prepared under the guidance of Dr. Vivek Kanhangad.

*Through this report we have tried to give a detailed description of the algorithms developed to design a book reading system. Our aim was to develop a system that helps capture an image of a textbook page, recognizes the text in the image and converts it into speech. We have tried to implement the android app combining several techniques which have been theoretically proposed in research papers but haven't been used together as well as several other techniques that we have developed on our own.*

*We have tried to the best of our abilities and knowledge to explain the contents in a lucid manner. We have also added images and figures depicting the effect of the techniques used to make the report more illustrative.*

**Akshay Joshi**
**Muhammad Talha Ansari**
B.Tech. IV Year
Discipline of Electrical Engineering
IIT Indore

Acknowledgements

We wish to thank Dr. Vivek Kanhangad for his kind support and valuable guidance during the course of this B.Tech. Project. It is only because of his constant help and support, that we were able to design and develop this app and technical report.

We would like to express our special gratitude to Mr. Sandarbh Sahu, B.Tech. II Year, Discipline of Computer Science & Engineering, IIT Indore who helped us transfer all the algorithms developed and tested on MATLAB to android platform resulting in the successful implementation of this app.

We would like to thank our esteemed institute, IIT Indore that provided us this opportunity to work on a B.Tech. Project that was interesting and helped improve our technical skills.

**Akshay Joshi**
**Muhammad Talha Ansari**
B.Tech. IV Year
Discipline of Electrical Engineering
IIT Indore

# Abstract

Humans are inquisitive by nature and are constantly in search for knowledge. Books are a medium to quench their thirst for knowledge. However, there are few who cannot access the books due to several reasons like visual deformity, illiteracy, etc. Hence we propose a book reading system by means of an android app that could help such people get access to books.

Our input will be an image containing text which will be processed using several algorithms to give us an audio output in return. Firstly, the focus of the image is analysed using *fast focus assessment* algorithm. The image is passed or rejected on the basis of its focus score. Then the features of the accepted image is enhanced using several image processing techniques. Then the image is passed through an algorithm to check whether the page is completely captured or not. Then the text from the image is extracted using *optical character recognition* (OCR). Then finally, text is converted into speech output.

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Background

In the past, there have been several attempts to help the blind in academics. The most notable is the invention of the braille script. Later on a bold step was taken by Ray Kurzweil. In 1974, he started the company Kurzweil Computer Products, Inc. and continued development of omni-font OCR, which could recognise text printed in virtually any font. Kurzweil decided that the best application of this technology would be to create a reading machine for the blind, which would allow blind people to have a computer read text to them out loud. This device required the invention of two enabling technologies  the CCD flatbed scanner and the text-to-speech synthesizer. On January 13, 1976, the successful finished product was unveiled during a widely reported news conference headed by Kurzweil. In the 2000s, OCR was made available online as a service (WebOCR), in a cloud computing environment, and in mobile applications like real-time translation of foreign-language signs on a smartphone.

An intelligible text-to-speech program allows people with visual impairments or reading disabilities to listen to written words on a home computer. Many computer operating systems have included speech synthesizers since the early 1990s. In 1779 the German-Danish scientist Christian Gottlieb Kratzenstein won the first prize in a competition announced by the Russian Imperial Academy of Sciences and Arts for models he built of the human vocal tract that could produce the five long vowel sounds (in International Phonetic Alphabet notation: [a], [e], [i], [o] and [u]). Later on the system has evolved and is now available in all mobile phones.

In the modern world, advanced technology should assist any deformed individual to stand next to a normal person and receive equal number of opportunities. This project is a work to implement a book reading system by means of image processing.

There are several techniques used to check the focus of an image. A few techniques are grey level local variance methods, gradient magnitude based methods, second derivative based methods, etc. In this project, fast focus assessment method has been used.

There are various image processing techniques available in libraries of different platforms. A few techniques are filtering with morphological operators, histogram equalization, noise removal using a Wiener filter, linear contrast adjustment, median filtering, unsharp mask filtering, contrast-limited adaptive histogram equalization (CLAHE), decorrelation stretch, etc. These techniques are tested experimentally to work in synchronization with each other to maximize the efficiency of character recognition.

Optical character recognition and text to speech conversion can be directly implemented using the predefined library functions available in various platforms. Early versions of OCR needed to be trained with images of each character, and worked on one font at a time. Advanced systems capable of producing a high degree of recognition accuracy for most fonts are now common, and with support for a variety of digital image file format inputs. Some systems are capable of reproducing formatted output that closely approximates the original page including images, columns, and other non-textual components.

## 1.2 Overview

This B.Tech project is a work to develop an android based application capable of performing the following tasks :

1. To capture an image of a text containing page.
2. To determine whether the image is focused or not.
3. Carry out image processing techniques to enhance the features of image and its properties like sharpness, contrast, SNR, etc.
4. To check whether the complete page has been captured or not and guide the user accordingly to move his/her hands in a particular set of directions so as to completely capture the page.
5. Detecting the text characters/words in an image.

6. Reading out all the words at a proper pace until the end of the text page.

## 1.3 Flow of Control through Algorithms

An image has to pass through various algorithms before the user receives the expected speech output. Figure 1 depicts the stepwise flow of control through the algorithms when a captured image is passed through them. A captured image may be rejected in a few steps listed below if it doesn't match the required specifications. The user would be provided with a set of directives so that he can recapture the image satisfying the desired specifications.

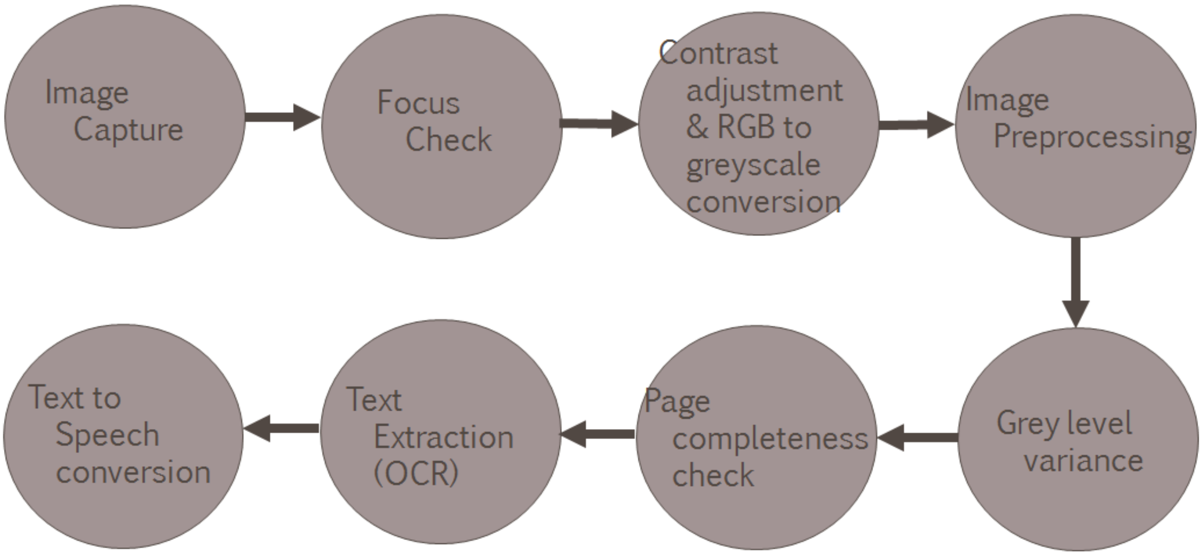



Figure 1.1: Flow chart of various algorithms through an image is passed

Firstly, the captured image is passed through the *fast focus assessment* algorithm which checks the focus of the image. This algorithm gives a focus score on the basis of which the image is either passed or rejected. For an unfocused or blurred image the focus score is low and the image is rejected. The user is then asked to recapture the image. For a properly focused image the focus score is quite high and exceeds a predefined threshold. Such an image is passed for further processing.

In the next step, the features of the image are enhanced using several image processing techniques. Image processing techniques used are contrast adjustment, colour to gray scale conversion, image sharpening and visibility enhancement. After this the image is passed through the *grey level variance* algorithm which eliminates the background and separates out the text in the image.

The pre processed image is passed through a *page completeness check* algorithm. A dark mask is generated to cover the textual part of the pre-processed image. If the mask is bounded by lighter pixels on all sides and occupies sufficient area then the page is said to be complete. If the image fails to satisfy the page completeness criteria, the user is directed to recapture the image. Otherwise, if the image is passed, the masked region is separated out from the pre processed and the resultant image is passed for further operations.

After this the image passed is through the *optical character recognition* (OCR) algorithm. This algorithm recognizes the text/character in the image, extracts it and stores it in a variable.

Finally, the image is passed through *text to speech conversion* algorithm which provides audio output for the text stored in the variable.

## 1.4 Motivation for Work

Albert Einstein once said that *knowledge is like a statue of marble standing in a desert which needs to be polished regularly otherwise it will deteriorate.* Every person is not capable to quench their thirst for knowledge by themselves. Books are a medium to quench this thirst. Few people are not able to read books because of visual disability. There are about 180 million visually impaired people in the world and if even a small percentage can be helped, that is a large number. In the modern world with advanced technology, visual deformity should not be a hindrance in the progress of any individual. Technology should be able to assist such people, so that they can stand at the same level as any normal person.

The objective of this project is to make an android app which captures image of a text containing page, extracts words from the image and provides audio output of the same. Most of the times Braille scripted books are not easily available in close proximity but normal books are, so in such situations this app would be very useful for blind people to read any book which is easily available. This android app will be useful not only for blind people and old people having poor eyesight but people who dont like to read or are illiterate, they can also listen to their books.

# Chapter 2

# Description of key features

## 2.1 Focus Assessment

Image focus assessment is performed in real time (faster than video frame rate) by measuring spectral power in middle and upper frequency bands of the 2-D Fourier spectrum of each image frame and seeking to maximize this quantity either by moving an active lens or by providing audio feedback to Subjects to adjust their range appropriately [3], [1]. The video rate execution speed of focus assessment (i.e., within 15 ms) is achieved by using a bandpass 2-D filter kernel requiring only summation and differencing of pixels, and no multiplications, within the 2-D convolution necessary to estimate power in the selected 2-D spectral bands.

The acquisition of images in good focus is essential to get satisfactory efficiency of character recognition. The un sharpened or unfocused text in the images adds to limit the efficiency further. It is therefore desirable to compute focus scores for images so that the images with good focus can be passed and the ones with poor focus can be rejected.

Pixels are combined according to the (8×8) convolution kernel shown in Fig. 2.1. The simple weights mean that the sum of the central (4×4) pixels can just be tripled, and then the outer 48 pixels subtracted from this quantity; the result is squared and accumulated; and then the kernel moves to the next position in the image, selecting every fourth row and 4th column.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | +3 | +3 | +3 | +3 | -1 | -1 |
| -1 | -1 | +3 | +3 | +3 | +3 | -1 | -1 |
| -1 | -1 | +3 | +3 | +3 | +3 | -1 | -1 |
| -1 | -1 | +3 | +3 | +3 | +3 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |

Figure 2.1: The (8x8) convolution kernel for fast focus assessment [3].

This kernel matrix when convolved with the image matrix, gives us a 2-D matrix. The absolute values of the elements of this matrix are squared and summed which generates a focus score. The higher the score, more is the focus of the image.

The experiments showed that 50 was a suitable threshold score to get satisfactory character recognition efficiency. If the score of any image is below the threshold value, the user is asked to recapture the image.

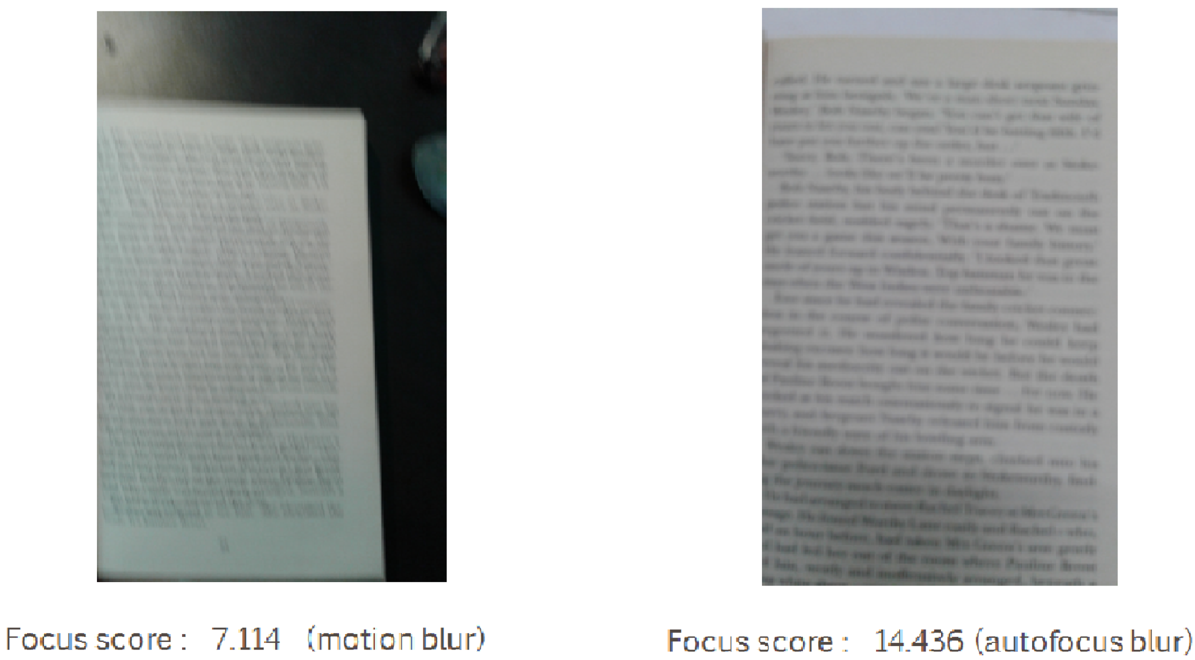
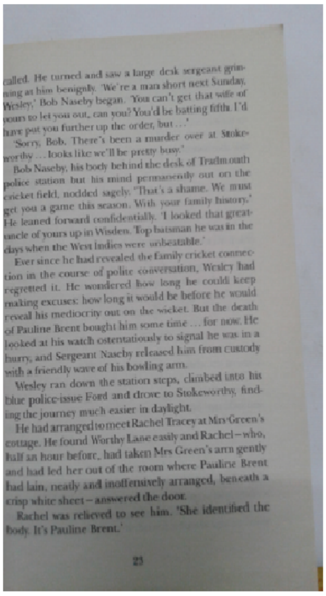



Figure 2.2: Focus scores of rejected images.



called. He turned and saw a large desk sergeant grinning at him benignly. 'We're a man short next Sunday, Wesley,' Bob Naseby began. 'You can't get that wife of yours to let you out, can you? You'd be batting fifth. I'd have put you further up the order, but ...'

'Sorry, Bob. There's been a murder over at Stokeworthy ... looks like we'll be pretty busy.'

Bob Naseby, his body behind the desk of Tradmouth police station but his mind permanently out on the cricket field, nodded sagely. 'That's a shame. We must get you a game this season. With your family history.' He leaned forward confidentially. 'I looked that great-uncle of yours up in Wisden. Top batsman he was in the days when the West Indies were unbeatable.'

Ever since he had revealed the family cricket connection in the course of polite conversation, Wesley had regretted it. He wondered how long he could keep making excuses: how long it would be before he would reveal his mediocrity out on the wicket. But the death of Pauline Brent bought him some time ... for now. He looked at his watch ostentatiously to signal he was in a hurry, and Sergeant Naseby released him from custody with a friendly wave of his bowling arm.

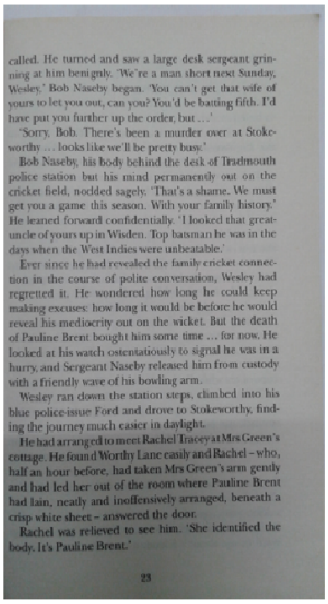
Wesley ran down the station steps, climbed into his blue police-issue Ford and drove to Stokeworthy, finding the journey much easier in daylight.

He had arranged to meet Rachel Tracey at Mrs Green's cottage. He found Worthy Lane easily and Rachel – who, half an hour before, had taken Mrs Green's arm gently and had led her out of the room where Pauline Brent had lain, neatly and inoffensively arranged, beneath a crisp white sheet – answered the door.

Rachel was relieved to see him. 'She identified the body. It's Pauline Brent.'

23

Focus score : 65.1916 (med focus)




called. He turned and saw a large desk sergeant grinning at him benignly. 'We're a man short next Sunday, Wesley,' Bob Naseby began. 'You can't get that wife of yours to let you out, can you? You'd be batting fifth. I'd have put you further up the order, but ...'

'Sorry, Bob. There's been a murder over at Stokeworthy ... looks like we'll be pretty busy.'

Bob Naseby, his body behind the desk of Tradmouth police station but his mind permanently out on the cricket field, nodded sagely. 'That's a shame. We must get you a game this season. With your family history.' He leaned forward confidentially. 'I looked that great-uncle of yours up in Wisden. Top batsman he was in the days when the West Indies were unbeatable.'

Ever since he had revealed the family cricket connection in the course of polite conversation, Wesley had regretted it. He wondered how long he could keep making excuses: how long it would be before he would reveal his mediocrity out on the wicket. But the death of Pauline Brent bought him some time ... for now. He looked at his watch ostentatiously to signal he was in a hurry, and Sergeant Naseby released him from custody with a friendly wave of his bowling arm.

Wesley ran down the station steps, climbed into his blue police-issue Ford and drove to Stokeworthy, finding the journey much easier in daylight.

He had arranged to meet Rachel Tracey at Mrs Green's cottage. He found Worthy Lane easily and Rachel – who, half an hour before, had taken Mrs Green's arm gently and had led her out of the room where Pauline Brent had lain, neatly and inoffensively arranged, beneath a crisp white sheet – answered the door.

Rachel was relieved to see him. 'She identified the body. It's Pauline Brent.'

23


Focus score : 115.0601 (clear)

Figure 2.3: Focus scores of passed images.

One can clearly observe that the score margin between the rejected and the passed images is sufficiently high.

## 2.2 Image Pre-processing

Image processing techniques enhance the features of an image which results in a significant improvement in the efficiency of OCR. Out of all the image processing techniques, the following were found most suitable to work in synchronization to improve the efficiency.

### 2.2.1 Contrast adjustment

Contrast adjustment maps the values of image matrix to new values in another matrix such that it spreads out the number of pixels confined in a small range of intensity values to a wider range. As a result the image has sharp differences between darker (high intensity value) and lighter (low intensity value) pixels.

To illustrate, the image on the left has poor contrast, with intensity values limited to the middle portion of the range. The image on the right has higher contrast, with intensity values that fill the entire intensity range. In the high contrast image, highlights look brighter and shadows look darker.

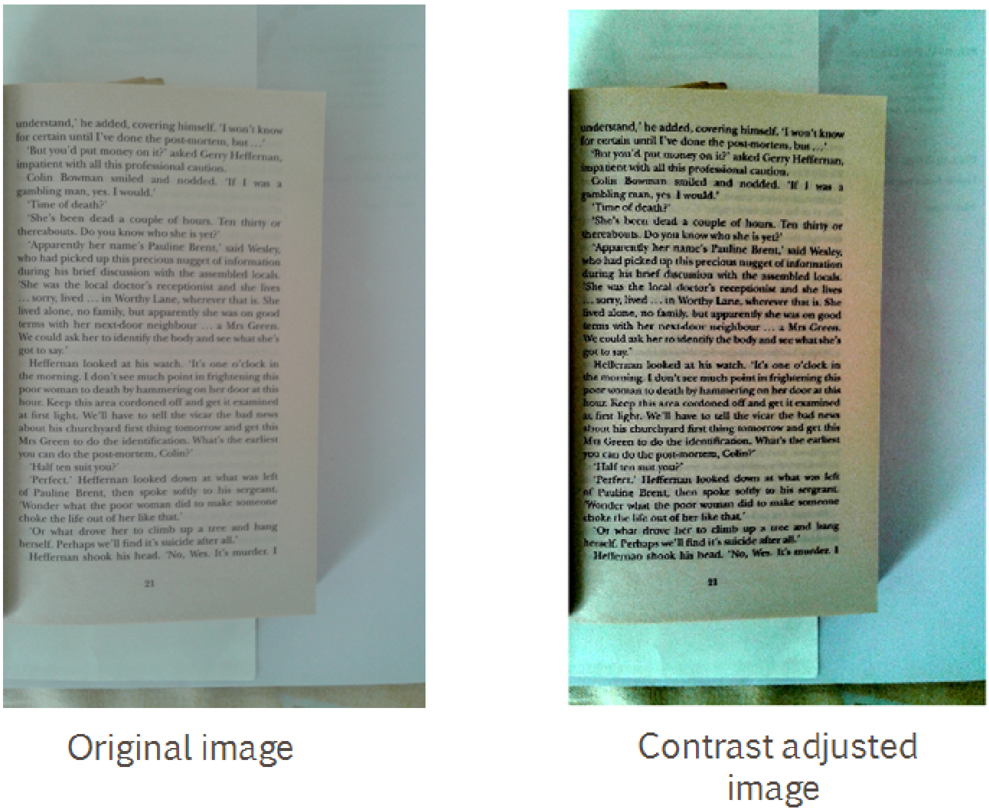



Figure 2.4: Effect of contrast adjustment on sample image.

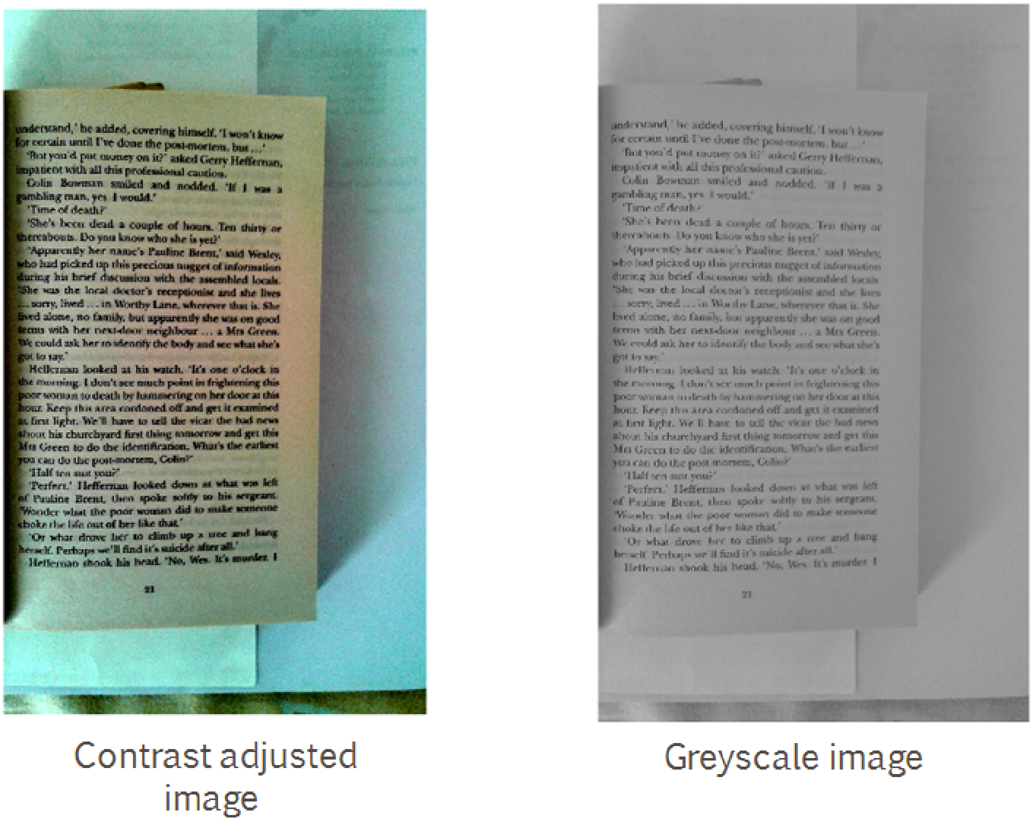
For the purpose of contrast adjustment in MATLAB, a predefined function of MATLAB library called *imadjust( )* was used. The input to this function is original captured image as well as the range of intensity values (Low to High) and the output is the contrast adjusted image. The range of intensity values can be calculated by using the following equation.

$$\text{Low} = \text{mean} - 2\sigma;$$
$$\text{High} = \text{mean} + 2\sigma;$$

where mean is the average and $\sigma$ is the standard deviation of the elements of image matrix.

### 2.2.2 RGB to greyscale conversion

This technique converts the truecolor image of channels Red, Green & Blue (RGB) to the grayscale intensity image.This feature was implemented in MATLAB via *rgb2gray( )* function. The *rgb2gray( )* function converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance. As a result the 3-D truecolor image matrix gets converted to 2-D grey scale image matrix.


understand,' he added, covering himself. 'I won't know for certain until I've done the post-mortem, but ...'
'But you'd put money on it?' asked Gerry Heffernan, impatient with all this professional caution.
Colin Bowman smiled and nodded. 'If I was a gambling man, yes. I would.'
'Time of death?'
'She's been dead a couple of hours. Ten thirty or thereabouts. Do you know who she is yet?'
'Apparently her name's Pauline Brent,' said Wesley, who had picked up this precious nugget of information during his brief discussion with the assembled locals. 'She was the local doctor's receptionist and she lives ... sorry, lived ... in Worthy Lane, wherever that is. She lived alone, no family, but apparently she was on good terms with her next-door neighbour ... a Mrs Green. We could ask her to identify the body and see what she's got to say.'
Heffernan looked at his watch. 'It's one o'clock in the morning. I don't see much point in frightening this poor woman to death by hammering on her door at this hour. Keep this area cordoned off and get it examined at first light. We'll have to tell the vicar the bad news about his churchyard first thing tomorrow and get this Mrs Green to do the identification. What's the earliest you can do the post-mortem, Colin?'
'Half ten suit you?'
'Perfect.' Heffernan looked down at what was left of Pauline Brent, then spoke softly to his sergeant. 'Wonder what the poor woman did to make someone choke the life out of her like that.'
'Or what drove her to climb up a tree and hang herself. Perhaps we'll find it's suicide after all.'
Heffernan shook his head. 'No, Wes. It's murder. I

21


Figure 2.5: Conversion of image from RGB to greyscale.

*rgb2gray( )* converts RGB values to grayscale values by forming a weighted sum of the R, G, and B components:

$$0.2989 * R + 0.5870 * G + 0.1140 * B$$

### 2.2.3 Image sharpening

Sharpness is actually the contrast between different colors. A quick transition from black to white looks sharp. A gradual transition from black to gray to white looks blurry. Sharpening images increases the contrast along the edges where different colors meet. This feature was used in the system to sharpen the boundaries of the text in order to increase their readability and therefore improves the efficiency of OCR.

This feature has been imployed in MATLAB through a function called *imsharpen( )*. *imsharpen( )* returns an enhanced version of the grayscale or truecolor (RGB) input image, where the image features, such as edges, have been sharpened using the *unsharp masking* method. The unsharp masking technique comes from a publishing industry process in which an image is sharpened by subtracting a blurred (unsharp) version of the image from itself.

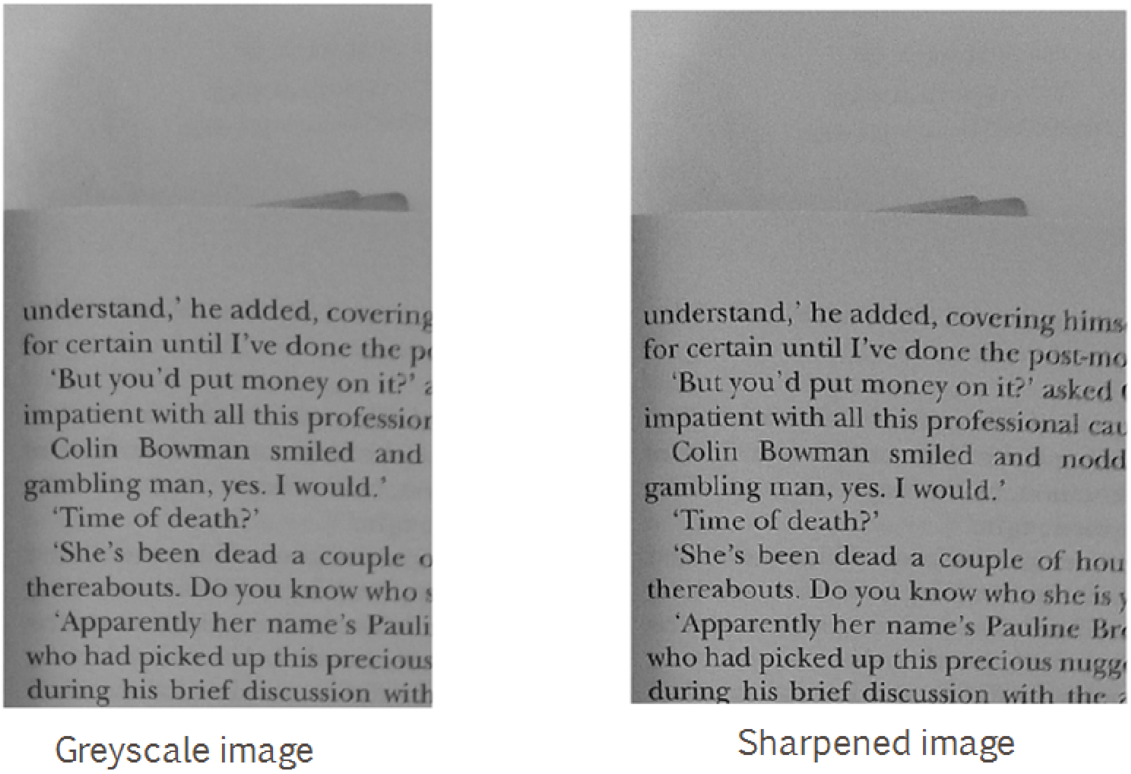



Figure 2.6: Effect of image sharpening on greyscale image.

To control the different aspects of sharpening, a few more parameters can be passed as input arguments.

**Radius**

This value controls the area (circular) of the region around the edge pixels that is affected by sharpening. A large value sharpens wider regions around the edges, whereas a small value sharpens narrower regions around edges. In the implementation of the image sharpening feature, the value of *radius* that was used was '2'.

**Amount**

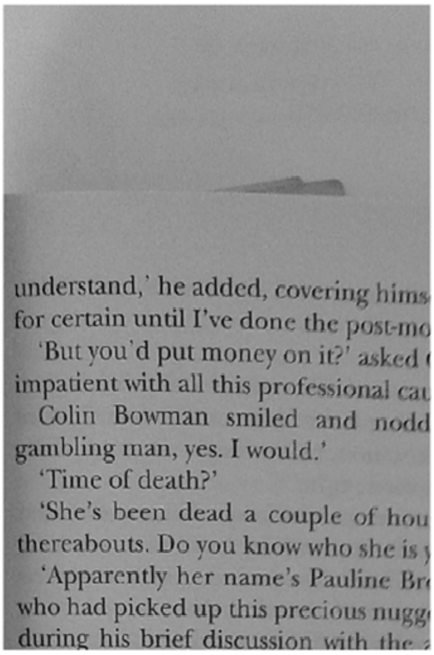
Strength or amount of the sharpening effect is quantized and specified as a numeric value. A higher value leads to larger increase in the contrast of the sharpened pixels. It is desirable that the values for this parameter are within the range [0 2], although values greater than 2 are allowed. Very large values for this parameter may create undesirable effects in the output image. In the implementation of the image sharpening feature, the value of *amount* that was used was '1'.

### 2.2.4 Visibility enhancement

The primary purpose of visibility enhancement is to enhance the color differences in an image. This feature was implemented in MATLAB uasing a predefined library function called *decorrstretch( )*. *decorrstretch( )* applies a decorrelation stretch to an image of m-by-n-by-n bands and returns the result as enhanced image. Resultant image has the same size and class as the input image, and the mean and variance in each band are the same as in input image. Input image can be an RGB image (nBands = 3) or can have any number of spectral bands.

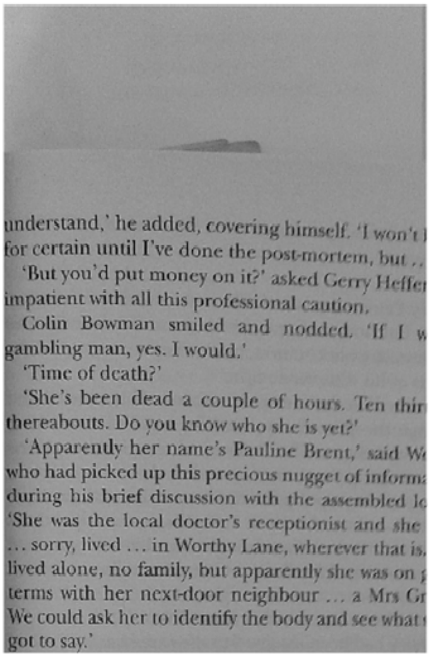
A decorrelation stretch is a linear, pixel-wise operation in which the specific parameters depend on the values of actual and desired (target) image statistics. The vector containing the value of a given pixel in each band of the input image is transformed into the corresponding pixel in output image as follows:

1. Removes a mean from each band.
2. Normalizes each band by its standard deviation (correlation-based method only).
3. Rotates the bands into the eigenspace of Corr or Cov.
4. Applies a stretch S in the eigenspace, leaving the image decorrelated and normalized in the eigenspace.
5. Rotates back to the original band-space, where the bands remain decorrelated and normalized.
6. Rescales each band according to desired output standard deviation.
7. Restores a mean in each band.



understand,' he added, covering hims
for certain until I've done the post-mo
'But you'd put money on it?' asked
impatient with all this professional ca
Colin Bowman smiled and nodd
gambling man, yes. I would.'
'Time of death?'
'She's been dead a couple of hou
thereabouts. Do you know who she is
'Apparently her name's Pauline Br
who had picked up this precious nugg
during his brief discussion with the

Sharpened image



understand,' he added, covering himself. 'I won't
for certain until I've done the post-mortem, but ..
'But you'd put money on it?' asked Gerry Heffer
impatient with all this professional caution.
Colin Bowman smiled and nodded. 'If I w
gambling man, yes. I would.'
'Time of death?'
'She's been dead a couple of hours. Ten thir
thereabouts. Do you know who she is yet?'
'Apparently her name's Pauline Brent,' said W
who had picked up this precious nugget of inform
during his brief discussion with the assembled lo
'She was the local doctor's receptionist and she
... sorry, lived ... in Worthy Lane, wherever that is.
lived alone, no family, but apparently she was on
terms with her next-door neighbour ... a Mrs Gr
We could ask her to identify the body and see what
got to say.'

Enhanced image

Figure 2.7: Effect of visual enhancement on sharpened image.

Although the difference between the images in Fig 2.7 might not be clear to naked eyes, the implementation of this feature resulted in the significant improvement in the efficiency of OCR.

## 2.3 Gray Level Variance

The *grey level variance* algorithm [3], [4] is basically used to extract the features of fingerprints like valleys and ridges. In this project, the algorithm is used to separate out text from background.

Captured image is usually composed of valid foreground region and invalid background region. Foreground region consists of textual characters, which are regular texture image area. The pixel grey value variance in this area is in accordance with the general rules of the texture image, which means that the local grey value varies within a reasonable range. Conversely, the background area can generally be given for the smoothing zone, in which no regular texture information exists. The local gray value difference is smaller, so is the local gray variance. Based on these characteristics, the local gray variance can be used for the image segmentation processing. This method can also be referred to as variance method. The algorithm implementation process can be summarized as follows:

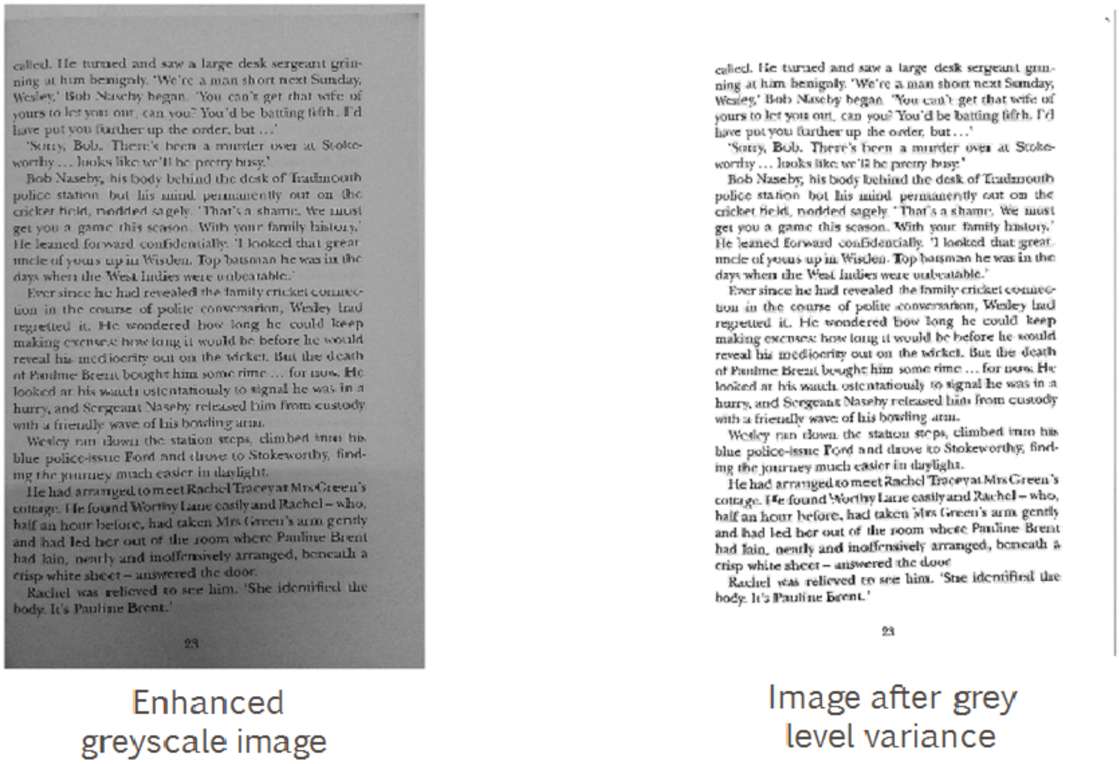
1. Input image is divided into non-overlapped image sub-blocks (window) with uniform size. Each pixel gray value denote as $H(i, j)$.
2. Mean gray values of each image sub-block, denoted as $M(u, v)$ are calculated according to formula:

$$M(u, v) = \sum_{i=(u-1)\times(w+1)}^{u\times w} \sum_{j=(v-1)\times(w+1)}^{v\times w} H(i, j)$$

3. Gray value variances of each sub-block, denoted as $S(u, v)$ are calculated according to formula:

$$S(u, v) = \sum_{i=(u-1)\times(w+1)}^{u\times w} \sum_{j=(v-1)\times(w+1)}^{v\times w} [H(i, j) - M(u, v)]^2$$

4. Each gray value variance of sub-block is compared with a predefined threshold value T. If $S(u, v)$ is greater than T, the sub-block is assigned to foreground area, whose gray value of image pixel will be elements retained for subsequent work. Otherwise, the sub-block is treated as background area and will be removed so as not to affect the following work.


called. He turned and saw a large desk sergeant grinning at him benignly. 'We're a man short next Sunday, Wesley,' Bob Naseby began. 'You can't get that wife of yours to let you out, can you? You'd be batting fifth. I'd have put you further up the order, but …'

'Sorry, Bob. There's been a murder over at Stokeworthy … looks like we'll be pretty busy.'

Bob Naseby, his body behind the desk of Tradmouth police station but his mind permanently out on the cricket field, nodded sagely. 'That's a shame. We must get you a game this season. With your family history.' He leaned forward confidentially. 'I looked that great uncle of yours up in Wisden. Top batsman he was in the days when the West Indies were unbeatable.'

Ever since he had revealed the family cricket connection in the course of polite conversation, Wesley had regretted it. He wondered how long he could keep making excuses: how long it would be before he would reveal his mediocrity out on the wicket. But the death of Pauline Brent bought him some time … for now. He looked at his watch ostentatiously to signal he was in a hurry, and Sergeant Naseby released him from custody with a friendly wave of his bowling arm.

Wesley ran down the station steps, climbed into his blue police-issue Ford and drove to Stokeworthy, finding the journey much easier in daylight.

He had arranged to meet Rachel Tracey at Mrs Green's cottage. He found Worthy Lane easily and Rachel – who, half an hour before, had taken Mrs Green's arm gently and had led her out of the room where Pauline Brent had lain, neatly and inoffensively arranged, beneath a crisp white sheet – answered the door.

Rachel was relieved to see him. 'She identified the body. It's Pauline Brent.'

23


Figure 2.8: Effect of gray level variance algorithm on enhanced image.

The figure depicts the effect of gray level variance algorithm on an enhanced image.

While implementing this algorithm, the size of the sub-block (window) is chosen as $10 \times 10$ ($w = 10$) and the threshold variance value (T) as 80. The sub-blocks exceeding the given threshold are left as it is whereas the pixels of the sub-blocks below the threshold are changed to white.

The gray level variance algorithm is important because it acts as a tool for image processing and also acts as a pre-requisite for page completeness algorithm.

## 2.4 Page Completeness Assessment

The primary purpose of this algorithm is to check whether the page is completely captured in an image or not and provide audio guidance to the user to capture the image properly. The captured image might be inappropriate in terms of page completeness assessment in several ways such as the page in the image might be tilted, the image might be captured from too far, the page in the image might be incomplete, or the image might also cover some text from the adjacent page. The image would be rejected by the algorithm in either of the above mentioned cases and the user would be provided suitable audio guidance to recapture the image properly.

A dark mask is generated to cover the textual part of the preprocessed image. If the mask is bounded by lighter pixels on all sides and covers sufficient area then the page said to be complete. The algorithm implementation process can be summarized as follows:

1. In the variance matrix S found in gray level variance algorithm, analyse each row one by one from the beginning.
2. Count the number of elements exceeding the threshold (T=80) in a given row.
3. If the ratio of count to the total number of elements in the row exceeds 0.25 then the u coordinates of the variance matrix is noted in an array for the first and the last such case encountered.
4. Repeat the steps 1, 2 and 3 to analyse columns one by one and obtain the v coordinates accordingly.
5. A mask is generated by keeping the noted u coordinates as the vertical bound and v coordinated as the horizontal bound.
6. The intensity values of the pixels covering the masked region are assigned to be 0 thereby generating a dark mask.
7. Suppose w×w is the window size and M & N are the number of rows and columns of image matrix respectively. If the following conditions are satisfied then the page in captured image is said to be complete.

$$50 \leq u_{first} \leq 300;$$
$$M - 300 \leq u_{last} \leq M - 50;$$
$$30 \leq v_{first} \leq 120;$$
$$N - 120 \leq v_{last} \leq N - 30;$$

8. If the below conditions are also satisfied then the image was captured from too far and the user needs to move a bit closer.

$$u_{last} - u_{first} \leq 30;$$
$$v_{last} - v_{first} \leq 12;$$

9. If the page in the image is complete and it covers sufficient area then the masked region is cropped out else the image is rejected.

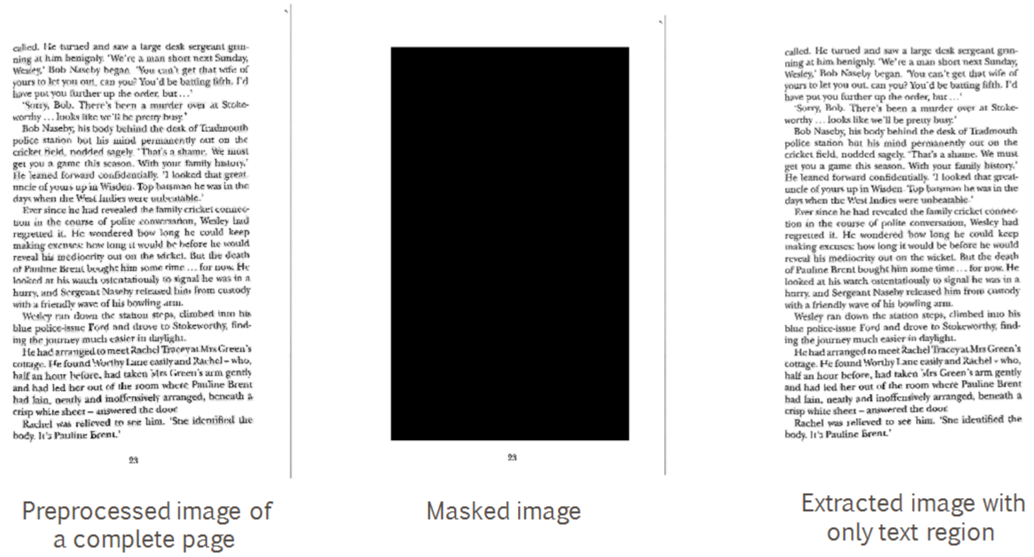



Figure 2.9: Page completeness assessment.

This feature is necessary because sometimes when the image is captured, certain dark spots or dots appear on the sides, due to unevenness of the page or non uniform lighting.They reduce the efficiency of OCR. This algorithm in conjugation with the gray level variance algorithm, eliminates such spots. One can observe that even the page number is wiped out because it is not a part of the textual mask.

## 2.5 Optical Character Recognition (OCR)

The is the core feature of the system because it determines the overall efficiency of the system. Optical character recognition (also optical character reader, OCR) is the mechanical or electronic conversion of images of typed, handwritten or printed text in to machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image (for example from a television broadcast). It is a common method of digitising printed texts so that they can be electronically edited, searched, stored more compactly, displayed on-line, and used in machine processes such as cognitive computing, machine translation, (extracted) text-to-speech, key data and text mining.

This feature recognizes, extracts text / characters from the image and stores them in a variable. This feature was implemented in MATLAB using a predefined function called *ocr( )* returns an ocrText object containing optical character recognition information from the input image. The object contains recognized text, text location, and a metric indicating the confidence of the recognition result. Libraries linked to open source OCR software called Tesseract is used as a basis for the implementation of text reading system for visually disabled in Android platform.

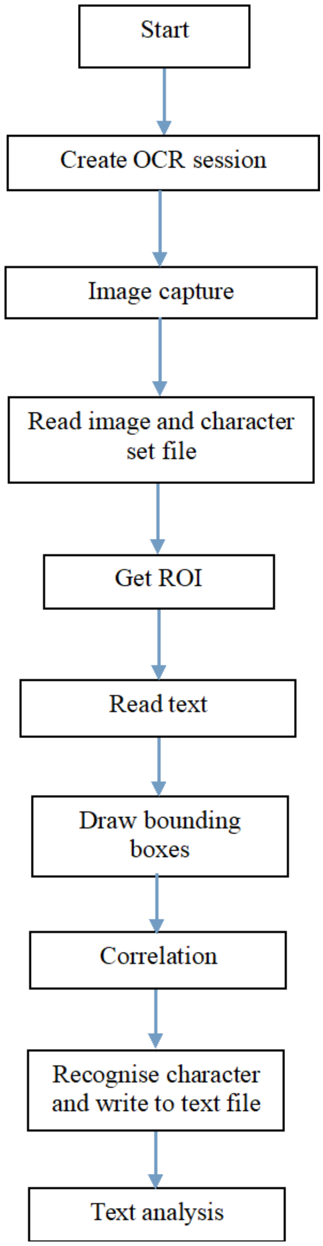



Figure 2.10: OCR flow chart.

The step wise operations involved in the execution of OCR algorithm are:

1. Filter the preprocessed image.
2. Based on the inter line spacing, crop out the lines from the image one by one.
3. Based on the inter word and inter character spacing, crop out the characters from each line one by one .

4. Resize the cropped out letters.
5. Perform correlation operation with the set of character images stored in the database.
6. Assign the character with the highest correlation score as the given character.
7. Store the extracted character one by one in a variable keeping the spacing between consecutive characters same as analysed before.

### 2.5.1 Text To Speech conversion

Speech synthesis is the artificial production of human speech. A text-to-speech (TTS) system converts normal language text into speech. Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output. A text-to-speech system (or "engine") is composed of two parts: a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. This process is often called text normalization, pre-processing, or tokenization. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences. Phonetic transcriptions and prosody information together make up the symbolic linguistic representation that is output by the front-end. The back-endoften referred to as the synthesizer then converts the symbolic linguistic representation into sound [5]. The speech synthesis from text was first implemented on MATLAB using Microsoft Win 32 speech application program interface library using the *.NETaddAssembly('System.Speech ')* command that utilizes the *speech synthesizer class* to produce speech information available for computer in MATLAB. The speech synthesizer class triggers the Windows own text to speech feature to generate the required audio output at a controllable pace. In android app, speech conversion was implemented using *TTS API* that enables our system (Android device) to speak text of different languages at a controllable pace and pitch.

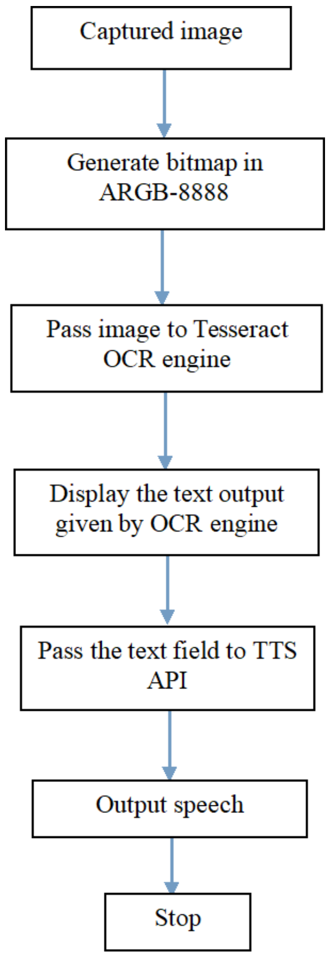



Figure 2.11: Process flow chart of text to speech conversion.

# Chapter 3

# Results

## 3.1 Experimental results

The above system was analysed on a set of 30 images and following results were obtained:

1. From the focus assessment experiments carried out on the sample images, threshold focus score was chosen as 50 which was optimally satisfying the requirements of character recognition.
2. The image processing techniques that were used and tested in this project, maximized the efficiency of OCR .
3. The gray level variance algorithm was successful in eliminating the irregularities like non-uniform lighting in the captured image.
4. The page completeness was successful in masking the textual region properly for normal images as well as images captured from too far away.
5. For the tested images, the efficiency of OCR in MATLAB was noted to be around 95% and that for the android app was noted around 80%.
6. The speech output was tested by varying characteristics such as pitch and pace.

## 3.2 UI of the Android app

The user interface of the android app is depicted in the below shown images. One can clearly observe the various stages that image has to pass through before the user receives an audio output.

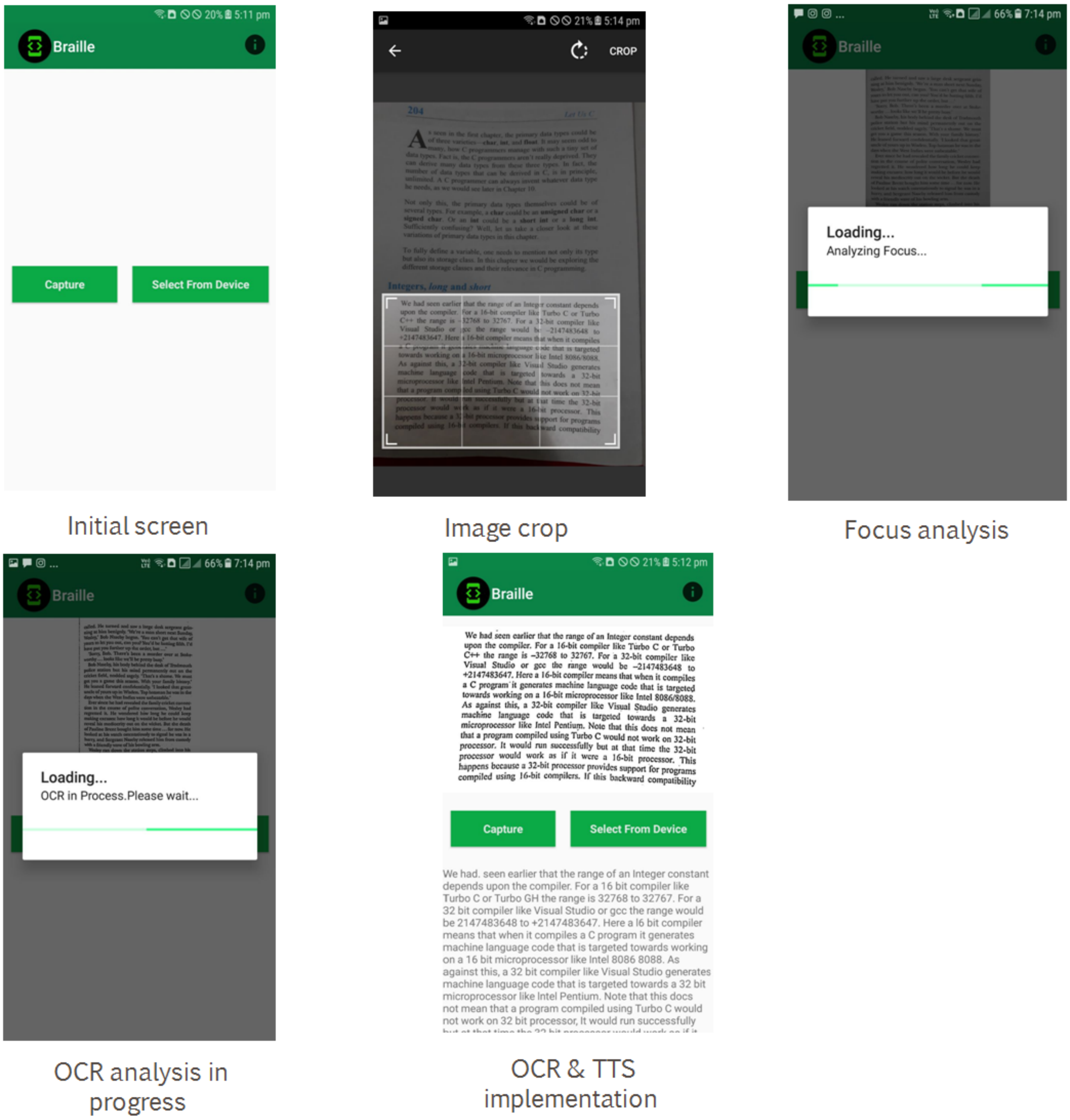



Figure 3.1: UI of Android app

# Chapter 4

# Conclusion

## 4.1 Conclusion

There are about 45 million blind people and 135 million visually impaired people worldwide. Disability of visual text reading has a huge impact on the quality of life for visually disabled people. Although there have been several devices designed for helping visually disabled to see objects using an alternating sens such as sound and touch, the development of text reading device is still at an early stage.

Existing systems for text recognition are typically limited either by explicitly relying on specific shapes, colour mask, requiring user assistance or may be of high cost. Therefore we need a low cost system that will be able to automatically locate and read text aloud to visually impaired persons.

This project is an effort to suggest an approach for image to speech conversion using optical character recognition and text to speech technology. By this approach we can read text from a document and can generate synthesized speech through a computer's speakers or phones speaker. System can be used to make information browsing for people who do not have the ability to read or write. This approach can be used in part as well. If we want only to text conversion then it is possible and if we want only text to speech conversion then it is also possible easily. People with poor vision or visual dyslexia or totally blindness can use this approach for reading the documents and books. People with speech loss or totally dumb person can utilize this approach to turn typed words into vocalization. Experiments have been performed to test the text reading system and good results have been achieved.

## 4.2 Scope for future work

In future, the following features need to be added to the system to make it perfect

1. Improving the efficiency of page completeness algorithm.
2. Taking care of special and exceptional cases of image capture as mentioned previously.
3. Improving the efficiency of OCR by implementing better pre-processing techniques.
4. Implementing the page completeness algorithm on the android app.
5. Implementing the audio guidance system in the android app.

# References

[1] W. Bocker, W. Rolf, W. Muller, C. Streffer, "Investigations about autofocus-algorithms for fluorescent-microscopy", *SPIE Applications of Digital Image Processing XIX*, vol. 2847, pp. 445-456, 1996.

[2] J. Daugman, "How iris recognition works", *IEEE Transactions on circuits and systems for video technology*, vol. 14, no.1, pp. 21–30, 2004.

[3] P. Pacheco, J. L. Cristbal, G. C. J. Martinez, and J. F. Valdivia, "Diatom autofocusing in brightfield microscopy: a comparative study". *In Pattern Recognition: Proceedings of the 15th International Conference on Pattern Recognition. ICPR-2000, Barcelona, Spain, September 3-7, 2000*, Eds. New York: IEEE, 2000, pp. 314-317.

[4] J. H. Chang, F. K. Chin, "Fingerprint ridge allocation in direct gray-scale domain", *Pattern Recognition*, vol. 34, no. 10, pp. 1907–1925, 2001.

[5] R. W. Sproat, J. P. Olive "Text-to-speech synthesis", *AT & T Technical Journal* , vol. 74, no. 2, pp. 35–44, 1995.