Optical Character Recognition system on smart phones for Indic Scripts

A Project Report

submitted by

AMAN GUPTA

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

BACHELOR OF TECHNOLOGY AND MASTER OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS

MAY 2018

THESIS CERTIFICATE

This is to certify that the thesis titled **Optical Character Recognition system on smart phones for Indic Scripts**, submitted by **Aman Gupta**, to the Indian Institute of Technology, Madras, for the award of the degree of **Bachelor of Technology and Master of Technology**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Devendra Jalihal
Research Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: 10th May 2018

ACKNOWLEDGEMENTS

I express my sincere gratitude to Dr. Devendra Jalihal for giving me this opportunity to work under his guidance. His guidance was instrumental in enhancing my interest on the subject. Throughout the course of my project, he has encouraged me to perform better and has shown faith in my abilities.

I take this opportunity to express gratitude to all of the Department faculty members for their help and support.

Finally, I would like to thank my parents for their support, trust and encouragement throughout my stay at IIT Madras.

ABSTRACT

KEYWORDS: Optical Character Recognition; Perspective Transformation, Preprocessing, Segmentation, Binarization;

Optical Character Recognition is a method of digitising printed texts so that they can be electronically edited, searched, stored more compactly [Wikipedia 2018]. OCR is a field of research in pattern recognition, artificial intelligence and computer vision.

The system presented here takes a character recognition technique wherein it digitizes the document character by character. Given the complications of Indic scripts, it is difficult to characterize each character of these languages. It comes with a trade-off of a relatively slow process when compared to word recognition.

The system is bundled end-to-end as an Android application which provides unique interface capabilities to the user. The preprocessing step is used to convert the input image into a normalized shape. It consists of steps such as noise cleaning, skew detection and correction. Perspective Cropping is used in the application to remove portions of a photo to create focus or strengthen the composition. It removes noise and helps in better recognition of the image. Further, Binarization is done to make the image more clear.

Tesseract is used as a tool to recognize the text in an image. User is given an option to edit the digitized text if the user has prior information about the wrongly recognized characters.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

IITM	Indian Institute of Technology, Madras
RTFM	Read the Fine Manual

CHAPTER 1

INTRODUCTION

1.1 Tesserect

1.1.1 What is tesserect?

Tesseract is an OCR engine with support for unicode and the ability to recognize more than 100 languages out of the box. It can be trained to recognize other languages. Tesseract is used for text detection on mobile devices, in video. Tesseract supports various output formats: plain-text, hocr(html), pdf, tsv, invisible-text-only pdf.

In order to get better OCR results,it is required to improve the quality of the image you are giving Tesseract. The quality of image is hence improved by removing various forms of noises. It is always better to perform cropping, binarization before feeding the image to the tesseract. It is compulsory to train tesseract for specific language. Source training data for supported language is available online.

1.1.2 Brief History

Tesseract was originally developed at Hewlett-Packard Laboratories Bristol and at Hewlett-Packard Co, Greeley Colorado between 1985 and 1994, with some more changes made in 1996 to port to Windows in 1998. In 2005 Tesseract was open sourced by HP. Since 2006 it is developed by Google.

CHAPTER 2

Character Recognition vs. Word Recognition

2.1 Character Recognition

Tesseract is considered one of the most accurate open source Optical Character Recognition engines currently available. It was initially developed for English, but has now been extended to recognize French, Italian, Catalan, Czech, Danish, Polish, Bulgarian, Russian, Greek,Korean,Spanish,Japanese,Dutch,Chinese,Indonesian, Swedish, German, Thai, Arabic, and Hindi etc.

Tesseract is the go-to engine for most Indic OCR research and development. Training the Tesseract OCR Engine for Hindi language requires in-depth knowledge of Devanagari script in order to prepare the character dataset.

Apart from training of this character data set, training the engine also needs to tackle character segmentation challenges, which are very specific to the script and font being used for training. Different fonts tend to render conjunct consonants in a way, that they do not even seem related when examined by an outsider.

2.2 Word Recognition

Spelling error in the following passage is deliberate, but is believed to be readable by most people.

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it does not mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm. Tihs is bcuseae the huamn mnid deos not raed ervey lteter by istlef, but the wrod as a wlohe.

Most people read through the above passage without any difficulty. Although no such research was ever carried out at Cambridge University, this example does reveal

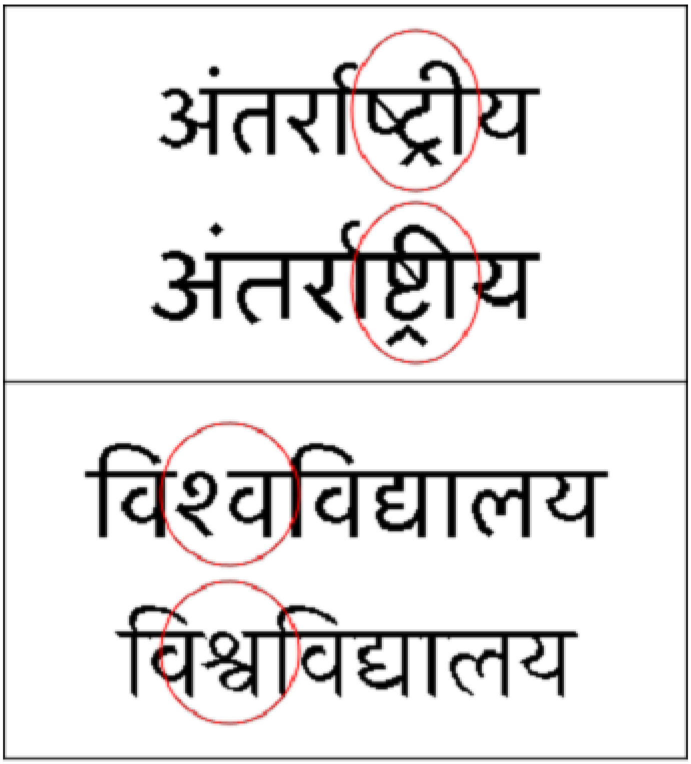


Figure 2.1: Conjunct consonants of Devanagari script complicate character segmentation algorithms.

an important aspect of the way human brain processes words. The brain predicts the entire word based on some feature in the structure of the word, rather than individual characters.

The word recognition technique has the following advantages:

- Construction of the word recognition framework is independent of the script being used. Many scripts can therefore be accommodated without extra engineering efforts
- Eliminates the need for further research in character segmentation techniques
- Discarding the character segmentation step makes pre-processing significantly faster.
- Inherent dictionary matching significantly improves word level accuracy

2.3 Problems faced in word recognition

The biggest disadvantage is the size limit. We cannot arbitrarily store all the words of the language since it will increase the bulkiness of the android application. Therefore, it comes with a huge trade-off of limited vocabulary

CHAPTER 3

Optical Character Recognition for Indian Languages

3.1 Introduction

Optical Character Recognition (OCR) is a process of converting printed or handwritten scanned documents into ASCII characters that a computer can recognize. In other words, automatic text recognition using OCR is the process of converting an image of textual documents into its digital textual equivalent. The advantage is that the textual material can be edited, which otherwise is not possible in scanned documents in which these are image files. The document image itself can be either machine-printed or handwritten, or a combination of the two.

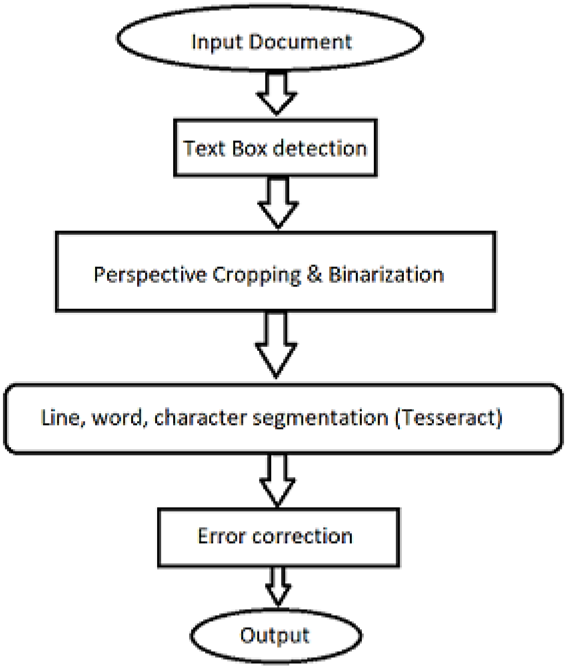

Input Document
Text Box detection
Perspective Cropping & Binarization
Line, word, character segmentation (Tesseract)
Error correction
Output

Figure 3.1: Flow Control for Indian Language OCR system

3.1.1 Motivation

For certain language scripts (e.g. Roman script), it is not difficult nowadays to develop an OCR system that recognizes well-shaped and well-spaced characters with an accuracy of 99 percent or above. However, it is still challenging to design a system that can maintain such high recognition accuracy, regardless of the quality of the input document and character font style variation.

In this work, our concern is Devanagari script, which is the script for Hindi, the national language of India and also for other languages like Sanskrit, Marathi and Nepali. The script is used by more than 300 million people across the globe. The algorithms which perform well for other scripts can be applied for Devanagari only after extensive preprocessing which makes simple adaptation ineffective. Therefore, it was necessary to do the research independently for Devanagari script. Only a few of them have considered real-life printed Hindi text consisting of character fusions and combined with a noisy environment.

Segmentation and classification are the two primary phases of a text recognition system. The segmentation process extracts recognition units from the text, which is usually a character. The classification process computes certain features for each segmented character and then they are assigned to a class that may be the true class (correct recognition), the wrong class (substitution error), or an unknown class (rejection error).

Indian scripts present great challenges to an OCR designer due to the large number of letters in the alphabet, the sophisticated ways in which they combine, and the complicated graphemes they result in. The problem is compounded by the unstructured manner in which popular fonts are designed. There is a lot of common structure in the different Indian scripts.

3.1.2 Framework Description

The Hindi Language consists of 12 vowels and 34 consonants. The presence of pre and post symbols added to demarcate between consonants and vowels introduces another level of complexity as compared to Latin script recognition. As a result, the complexity of deciphering letters from text written in Devanagari script increases dramatically because of presence of various derived letters from the basic vowels and consonants.

3.1.3 Dataset Generation

Unlike English, there are multiple ways to write the same word in Indian Languages which create difficulty in standardizing the data. Because of the limited scope of work being done in this realm, not many standard hand written dataset for Indian Languages exist. For this project we use a dataset which is provided by Google.

3.1.4 Contributions

In this project, we successfully ported Tesseract engine to the phone and made the application which works without the use of internet. The approach followed during the project was to successfully port Tesseract engine on the phone and to make the application completely off-line. Before giving an image to the tesseract, some pre processing is done in order to make it more clear so that tesseract gets the noise free image.

CHAPTER 4

Preprocessing Stage

4.1 Introduction

Preprocessing is an important step of applying a number of procedures for smoothing, enhancing, filtering etc, for making a digital image usable by subsequent algorithm in order to improve their readability for Optical Character Recognition software. The preprocessing tasks considered in the paper is conversion of gray scaled images to binary images, image rectification, and segmentation of the document and textual contents into paragraphs, lines, words, and then at the level of basic symbols. The various stages involved in the preprocessing are:

- Binarization
- Noise Elimination

4.1.1 Binarization

Linearization (thresholding) refers to the conversion of a gray-scale image into a binary image. This is also generally referred to as thresholding. There are two approaches for conversion of gray level image to binary form. First one is global threshold which picks one threshold value for the entire image, based on estimation of the background level from the intensity histogram of the image. The other one is local or adaptive threshold which uses different values for each pixel according to the local area information. The purpose of binarization is to identify the extent of objects and also to concentrate on the shape analysis, in which case the intensities of pixels are less significant than the shape of a region.

4.1.2 Noise Elimination

Noise that exists in images is one of the major obstacles in pattern recognition tasks. The quality of image degrades with noise. Noise can occur at different stages like image capturing, transmission and compression. Various standard algorithms, filters and morphological operations are available for removing noise that exists in images.

Noise elimination is also called as smoothing. It can be used to reduce fine textured noise and to improve the quality of the image.

Cropping the image greatly helps in reducing the noise. It basically removes the unwanted outer area from a photographic or illustrated image. The process often consists the removal of the outer parts or background of an image to improve framing or change aspect ratio to accentuate or isolate the subject matter. [Wikipedia]

CHAPTER 5

Android Application Development

5.1 Camera and Gallery

Upon opening the application, the user is asked to select a photo. Here, the user is given option to add the image either from the Gallery or directly from Camera. In majority of the use cases, the region of interest(region containing text) in the camera feed is a smaller portion compared to the full image. So the user is taken to the the Cropping step where the user makes use of the cropping tool to select the correct area. Cropped image is binarized in order to make it clear and noise free. The Binarized image is then saved and is given to OCR application for further processes.

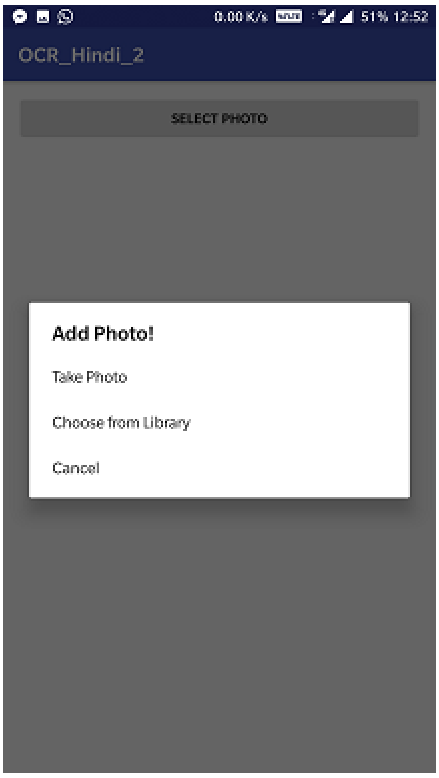


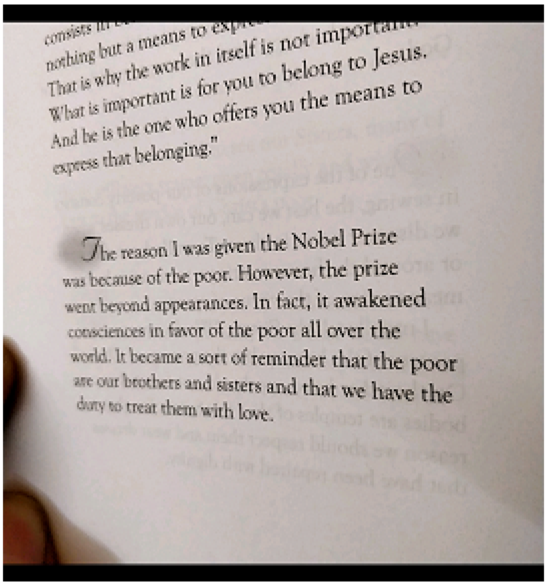
Figure 5.1: User is asked to select from gallery or take picture from Camera.

5.2 Perspective Cropping

For proper functioning of the recognition engine, the input image is required to have correct orientation with minimal distortion. In real world situations, it is not usually possible to capture a roadside sign-board or a poster from the perfect angle. Therefore, most captures are warped by a projective transformation. The correction of these captures are not possible by simple affine transformation on the image. But, correction of such images can greatly increase usability of the application.

A novel perspective cropping interface was implemented for this purpose.The Perspective Crop tool lets you transform the perspective in an image while cropping. Use the Perspective Crop tool when working with images that contain keystone distortion. Keystone distortion occurs when an object is photographed from an angle rather than from a straight on view. The user is provided with a floating quadrilateral over the base image. The corners of this quadrilateral can be moved around by the user. There are no constraints on the position of these corners. The user can now choose a warped region of interest. A transformation is applied on the source bitmap such that these 4 points are stretched out to form a rectangle. The result of this transformation is shown on the screen in real-time. Once the user is s with the perspective correction, he can now choose to proceed with recognition.

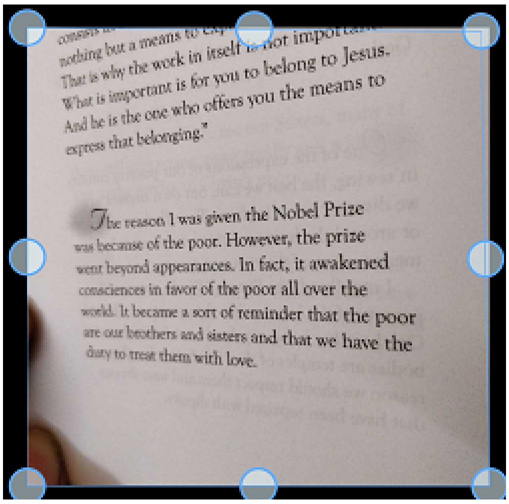
Although the aspect ratio information of the region of interest is not recovered, it will later be shown that this information has no bearing on the recognition accuracy. Therefore, aspect ratio correction can be ignored.



consists in ex
nothing but a means to expre
That is why the work in itself is not importan
What is important is for you to belong to Jesus.
And he is the one who offers you the means to
express that belonging."

The reason I was given the Nobel Prize was because of the poor. However, the prize went beyond appearances. In fact, it awakened consciences in favor of the poor all over the world. It became a sort of reminder that the poor are our brothers and sisters and that we have the duty to treat them with love.

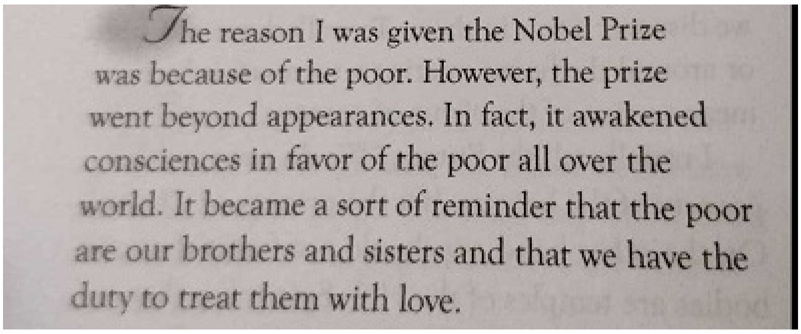
Figure 5.2: Normal Image taken from phone camera.



nothing but a means to exp
That is why the work in itself is not importa
What is important is for you to belong to Jesus.
And he is the one who offers you the means to
express that belonging."

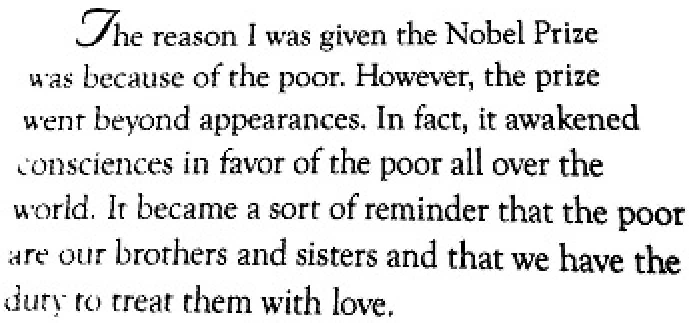
The reason I was given the Nobel Prize was because of the poor. However, the prize went beyond appearances. In fact, it awakened consciences in favor of the poor all over the world. It became a sort of reminder that the poor are our brothers and sisters and that we have the duty to treat them with love.

Figure 5.3: Floating quadrilateral is provided to the user to crop the required area



The reason I was given the Nobel Prize was because of the poor. However, the prize went beyond appearances. In fact, it awakened consciences in favor of the poor all over the world. It became a sort of reminder that the poor are our brothers and sisters and that we have the duty to treat them with love.

Figure 5.4: Croppped Image



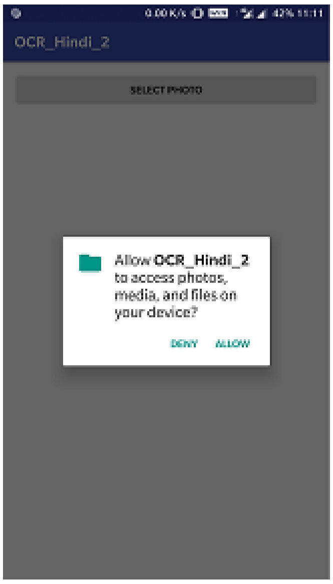
The reason I was given the Nobel Prize was because of the poor. However, the prize went beyond appearances. In fact, it awakened consciences in favor of the poor all over the world. It became a sort of reminder that the poor are our brothers and sisters and that we have the duty to treat them with love.

Figure 5.5: Binarization of the Cropped Image

5.3 Permission Overview

The purpose of a permission is to protect the privacy of an Android user. Android apps must request permission to access sensitive user data (such as contacts and SMS), as well as certain system features (such as camera and internet).

In this android application, the user who installs the application for the first time is asked to give permission to access Images from Gallery and Camera.



Allow **OCR_Hindi_2** to access photos, media, and files on your device?

Figure 5.6: Android Application asking the user to give rights for accessing its camera.

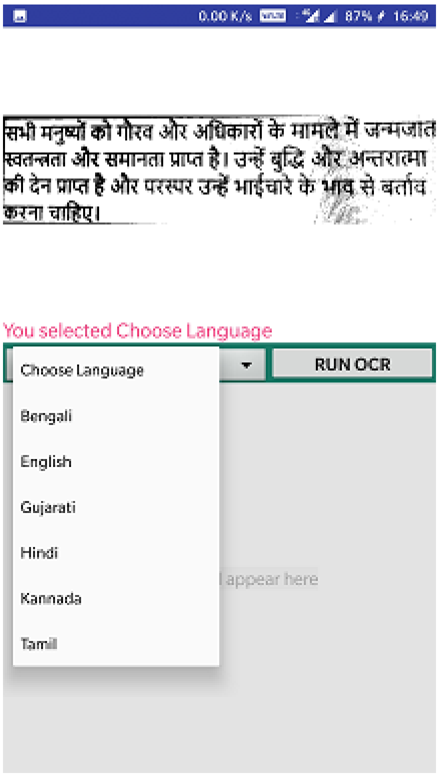
Once the text is detected by the tesseract, the user is given permission to edit the recognized text in order to correct the wrongly recognized characters/words. The text can be edited in any language since this part uses the Google Keyboard.

This step of allowing human intervention greatly helps in translating and transliterating the recognized text properly into any language since the source language is being corrected by the user.

5.4 Other Indian Languages

Since the project is successful in porting the Tesseract engine to the phone, it is easy to recognize all the indic scripts for which trained data is available. This Android Application recognizes other languages such as

- Bengali
- English
- Gujarati
- Hindi
- Kannada
- Tamil



सभी मनुष्यों को गौरव और अधिकारों के मामले में जन्मजात स्वतन्त्रता और समानता प्राप्त है। उन्हें बुद्धि और अन्तरात्मा की देन प्राप्त है और परस्पर उन्हें भाईचारे के भाव से बर्ताव करना चाहिए।

You selected Choose Language

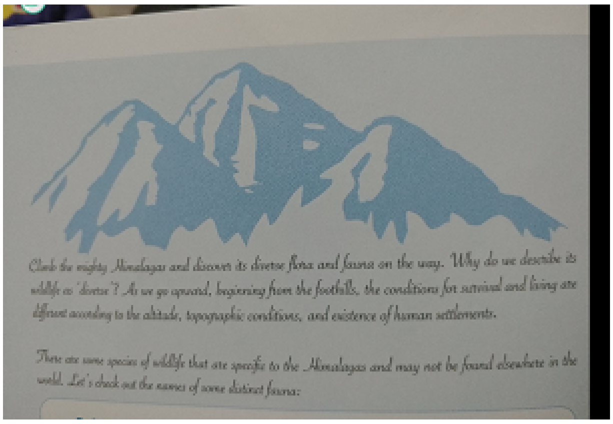
Figure 5.7: Screen shot taken from the Android Application asking the user to select the language

CHAPTER 6

Performance Evaluation

The end to end application was developed encapsulating all the modules discussed so far. For benchmarking performance, comparison was done with Tesseract OCR. The Tesseract engine also claims to be an end to end solution, but the engine has not been successfully ported to Android till now.

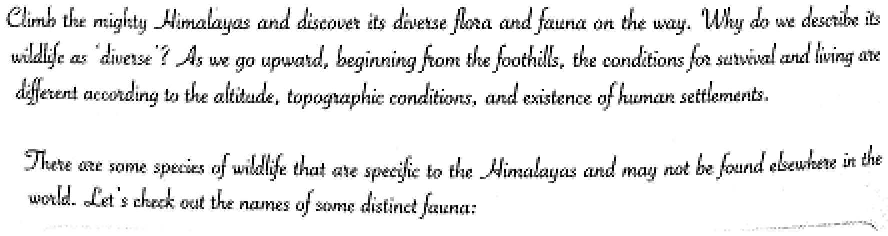
It has been found that the the android application shows much better result that the Tesseract engine on PC because in this application, processed image is given. The processed image has high quality in terms of less noise and more textual content which is an aided advantage in this android application. The difference between normal image and processed image is shown below:



Climb the mighty Himalayas and discover its diverse flora and fauna on the way. Why do we describe its wildlife as 'diverse'? As we go upward, beginning from the foothills, the conditions for survival and living are different according to the altitude, topographic conditions, and existence of human settlements.

There are some species of wildlife that are specific to the Himalayas and may not be found elsewhere in the world. Let's check out the names of some distinct fauna:

Figure 6.1: Normal Image



Climb the mighty Himalayas and discover its diverse flora and fauna on the way. Why do we describe its wildlife as 'diverse'? As we go upward, beginning from the foothills, the conditions for survival and living are different according to the altitude, topographic conditions, and existence of human settlements.

There are some species of wildlife that are specific to the Himalayas and may not be found elsewhere in the world. Let's check out the names of some distinct fauna:

Figure 6.2: Processed Image

CHAPTER 7

Conclusion

The aim of this project was to develop an OCR system with two requirements. Firstly, it should work for Indian Languages. This is done by successfully porting the Tesseract engine to the Android which wasn't done till now. The success of this project has a lot of advantage in terms of training. This gives a major boost to scaling the system to all Indian Languages, which is reported to represent about 66 different scripts. Now we can recognize all the languages at the cost of application size since we have to put the trained dataset for each language in the phone.

Secondly, the application was to be developed for a smart phone. Although Internet penetration is increasing, most users are still on slow, intermittent or in some cases no network connection. This condition is severe especially in India. Given that the system had to be designed for India and Indian languages,an offline solution carries lot of merit. This challenge was considered seriously even by Google for their Translate application, and has been heavily researched by their team.

Character recognition is one of the important applications of pattern recognition. In case of Hindi, good recognition rate is achieved for the following characters since these characters are of simplistic in nature

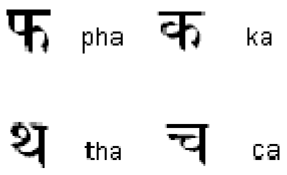


Figure 7.1: Simple nature of Hindi Characters.

Poor recognition rate of character is achieved for the following characters since these characters have close resemblance with ya and va.

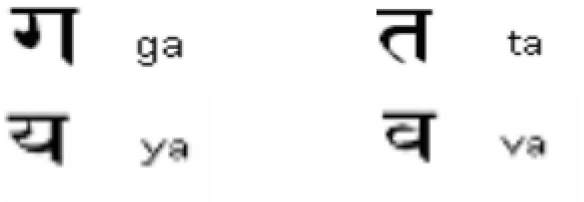


Figure 7.2: Complicated nature of Hindi Characters.

This system works for almost all Indian Languages.The user is given option to select the language it wants to detect. For the text documents which are large in size i.e they have more words to be recognized, the application becomes slow.

CHAPTER 8

Limitations and proposed suggestions

8.1 Limitations

The major limitation of this application is its size. Since everything is off-line, the application needs to store the models for each required language which leads to increase in size by almost 10 MB per language.

Table 8.1: Table of model sizes for different number of Languages

$NumberofLanguages$	$sizeoftheapplication(MB)$
1	60
2	72
3	85
4	95
5	107
6	125

Other limitations are :

- The android application doesn't detect the language of the processed image automatically. The user has to select the language.
- It becomes difficult to differentiate between characters which look alike.
- **Translation :** Translation of text will require internet service which is a problem in India. Although Internet penetration is increasing, most users are still on slow, intermittent or in some cases no network connection.

8.2 Proposed Suggestions

- There are two separate application as of now which can be combined together for better user interaction.
- **Transliteration :** It is the conversion of a text from one script to another script. It will help the user to transliterate the recognized text into the language it understands better.

CHAPTER 9

REFERENCES

1. **B.Indira** *et .al* (2012). Classification and Recognition of Printed Hindi Characters Using Artificial Neural Networks

2. **Smith,R.,** An overview of the tesseract ocr engine. *In icdar.* **IEEE, 2007.**

3. **Govindaraju,V.andS.Setlur,** Guide to OCR for Indic Scripts. Springer, **2009.**

4. **Google Open Source** URL: https://opensource.google.com/projects/tesseract

5. **Divakar Yadav** (2013). Optical Character Recognition for Hindi Language Using a Neural-network Approach

6. **Wikipedia** (2018). Optical character recognition wikipedia, the free encyclopedia.