# Text Recognition and Translation for Indian Regional Languages

**Muskan Momin**
Department of Computer Engineering
M.H. Saboo Siddik College of Engineering
Mumbai, India
muskanmomin89@gmail.com

**Durriya Bandukwala**
Department of Computer Engineering
M.H. Saboo Siddik College of Engineering
Mumbai, India
durriyabandukwala@gmail.com

**Akmal Khan**
Department of Computer Engineering
M.H. Saboo Siddik College of Engineering
Mumbai, India
akmalu786@gmail.com

**Aasim Khan**
Department of Computer Engineering
M.H. Saboo Siddik College of Engineering
Mumbai, India
aasimk015@gmail.com

**Lutful Islam**
Department of Computer Engineering
M.H. Saboo Siddik College of Engineering
Mumbai, India
lutful.islam@mhssce.ac.in

***Abstract*— In the workplace, government departments, classrooms, and universities, digital resources are in high demand. Many texts, such as letters and papers, are available in electronic media but must be translated into a readable regional language. Standard scanners, on the other hand, are hindered by their huge size and tedious activity. The need for document scanning is discussed in this paper, as well as how this application can help users resolve the language barrier by translating scanned documents. The web application developed is a Natural Language Processing-based framework for intelligent text recognition and translation.**

***Keywords*— *OCR, Text recognition, Text translation, Electronic form***

## I. INTRODUCTION

The Government of India has thousands of historic documents in regional languages depending on the states due to the country's immense diversity. Most of the languages that these documents are written aren't known around the world. This web application aims to extract the text from a document and then translate it into a global English language according to the user's requirements. Our goal is to integrate Tesseract OCR with various Indian regional language trained info. Tesseract, an Optical Character Recognition (OCR) programme, is used to convert images to text. This conversion aids in deciphering the language present in a specific picture. An open source online "ImTranslator" is used for text translation, which translates the understood text into English. A save as ".doc" option is also available for saving the document to a computer device. The user can also print hardcopies of these saved documents. This web application will ensure that regional language documents are made accessible to others, who will be able to easily access them with the help of our text recognition and translation feature.

## II. LITERATURE REVIEW

In [1] we review the outline of the algorithm used in the various stages of Tesseract's framework. It explains Adaptive Thresholding, which involves clustering image pixels and then page layout analysis separates the image into text and non-text sections, as well as breaking multi-column text into columns. We then find rows on the page. The detected words are then segmented into characters in the following section. Static and adaptive character classification are used. It also provides information on Tesseract OCR's capabilities.

In [2] we are provided with details on manuscripts and the available OCR framework. Many researchers have been working on handwritten character recognition in recent years, according to the paper. There seems to be no full OCR system that can comprehend Indian scripts. We studied about several attempts at handwriting recognition in Tamil, Bangla, Oriya, Devanagari and Gurmukhi. The paper contains a survey of OCR of these most familiar Indian scripts

## III. SYSTEM OVERVIEW

Language is a major impediment to globalisation. Nearly 36 states and union territories make up India, and nearly 20 regional languages are spoken. The government has thousands of documents in regional languages such as Sanskrit, Tamil, Bengali, and Marathi, among others, owing to the overwhelming diversity. There is also no OCR system available for Indian regional languages that is fully accurate. However, the OCR method is available for some regional languages. Since we are from Maharashtra, incorporating Marathi into text recognition was a critical challenge. Most government documents in Maharashtra are written in Marathi because it is easier for newcomers to understand; however, it is difficult to identify globally. In the OCR scheme, no web application offers an integrated real-time translation functionality. The ability to save a softcopy of known or translated data is also lacking, which would save a lot of space and manpower. On the other hand, the web application overcomes all of the above drawbacks. The programme uses an open source OCR called Tesseract for text recognition, and an integrated ImTranslator API for translation, all of which are open source. HTML code is used to save the text. If the user wishes to translate the text, he can do so using the additional ImTranslator programme.The problem of storing hardcopies of such documents will also be solved by this programme.The save to.doc function saves the recognised text in doc format, which saves a softcopy of the document to the user's device automatically.

## IV. PROPOSED METHODOLOGY

The primary aim of our application is to convert text documents written in regional languages like Marathi, Bengali, Tamil, Kannada, Telugu, Hindi, and Malayalam into understandable English. This web application uses “Tesseract,” a common OCR system, and “ImTranslator,” a Google Translator alternate source, to perform text recognition and translation.The application also reduces personnel required by storing a softcopy of such documents on a file server. The system contains three major features- text extraction and recognition, language translator, and text saving into a Word document.

### 1] Text Extraction And Recognition Text

Text extraction's main objective is to collect text from an image and recognise the characters within it. The OCR's preprocessing phase aids it in extracting the image's black coloured text and white background. Tesseract

is used to detect text. After that, the image is processed, and the spacing between each character is calculated. OCR uses pattern recognition and feature extraction technology to identify characters in tokens after they have been tokenized. Tokenization is the marking of a character between non-text pixels in an image as a 'token'. Each regional language's data is trained separately in the Tesseract-OCR library. Text recognition is performed depending on the language selected. This feature extracts text from an image, recognises it according to the language selected, and then displays it on the screen.

Figure 1. shows text extraction from document image.

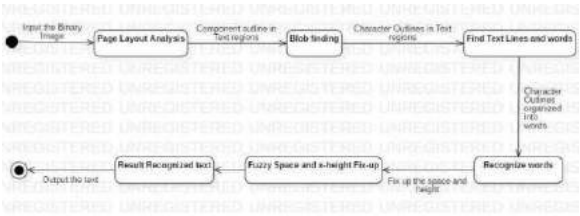


Fig. 1. Text extraction from image

## 2] Language Translator

In Text Translation, we will translate the recognized text into English language. The language translation is done with the open source "ImTranslator" API. The translator deciphers the structure of sentences in the regional language and produces a translation in the chosen language. ImTranslator is used to translate the text into English. It breaks the text into words and then converts each word into English, making the text simple to read for all. The text recognition and translation process is depicted in Figure 2.

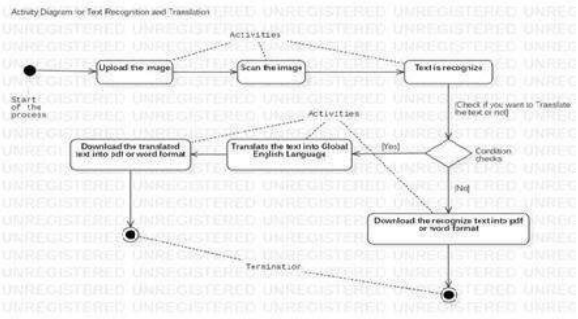


Fig. 2. Text recognition and translation

## 3] Text Saving into a Word Document

It is very traditional to store hardcopies of documents in today's modern era of technology. As a result, this web application incorporates the functionality, which saves the softcopy of these documents. With this tool, we can also save the recognised or translated text into doc format as a soft copy in the database. This reduces the number of people needed to physically manage the records.

# V. RESULTS

Even to the naked eye, each language has a few alphabets that look familiar. Apart from that, OCR has a difficult time distinguishing half-joined sentences too. The findings of the detection study for the languages used are presented in Figure 3.

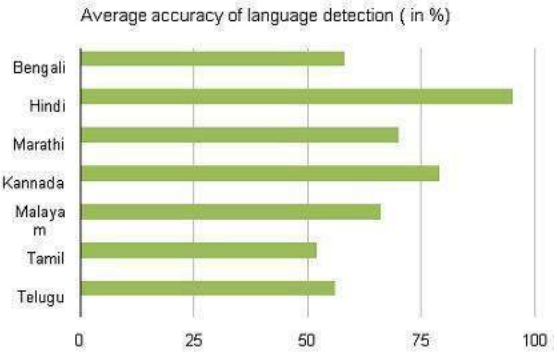


Fig. 3. Analysis of detection in all languages

### A. Test Results for Bengali Recognition

In Bengali, 14 words were given as input resulting in accuracy of 58%. The accuracy of detecting half joined words is increased. Figure 4. shows detection of sample Bengali text.

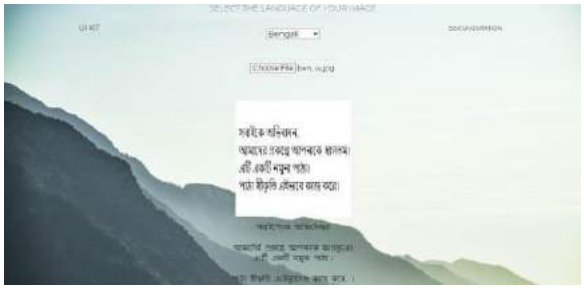

Fig.4. Detection of Bengali text

B. Test Result for Malayalam Recognition

For Malayalam, out of 15 input words, 8 were detected accurately, resulting in accuracy of 66%. For many letters, ം is detected wrongly since it is a common alphabet which is likely the result of over fitting. Figure 5. displays detection of sample Malayalam text.

Fig. 5. Detection of Malayalam text

C. Test Results for Tamil Recognition

In Tamil, out of 14 words as input, 5 were detected correctly, giving the accuracy of 57%. Figure 6. shows detection of Tamil text

Fig.6. Detection of Tamil text

D. Test Results for Kannada Recognition

For Kannada, out of 15 words as input, 10 were detected accurately, with the accuracy of 76%.The detection accuracy for individual characters is relatively higher but the accuracy for half joined words is much less, bringing it down to average. Figure 7. shows detection of sample Kannada text.

Fig.7. Detection of Kannada text

E. Test Results for Telugu Recognition

Telugu, for an input of 15 words, 7 words were detected correctly, resulting in accuracy of 43%. Figure 8. displays detection of sample Telugu text. Fig.8. Detection of Telugu text

Fig.8. Detection of Telugu text

F. Test Result for Hindi Recognition

Hindi has so far the highest accuracy with 15 words detected accurately out of 14 words giving the accuracy of 95%. Figure 9. shows detection of sample Hindi text.

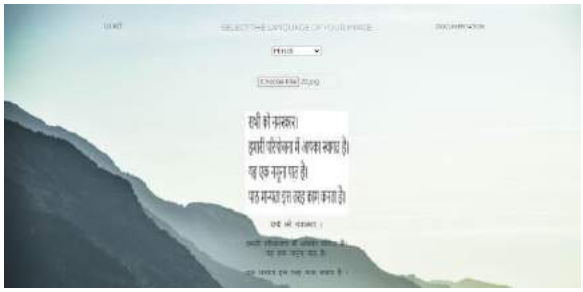

Fig.9. Detection of Hindi text

G. Test Results for Marathi Recognition

For Marathi, out of 15 words as input, 12 were detected correctly. Since Marathi language was a new addition to OCR, the aim was to achieve maximum accuracy. We were able to reach a highest accuracy of 68%. Even though the script for Marathi and Hindi is same, introduction of letter ऑcould increase Marathi's detection accuracy by much more. Figure 10. displays detection of Marathi text.

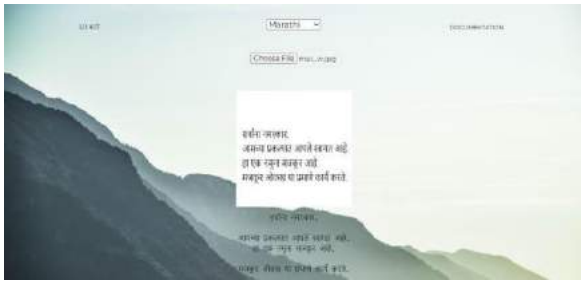

Fig.10. Detection of Marathi text

H. Translating into English from any Regional Language

The 2nd part of the project is translation into English. In Figure 11. an example of Hindi to English translation is shown. It works similarly for other languages. Fig.11. Translation of Hindi text into English I. Saving the file into .docx format The recognized text is saved into .docx format after translation. Fig 12. shows saving recognized text into a document for future reference.

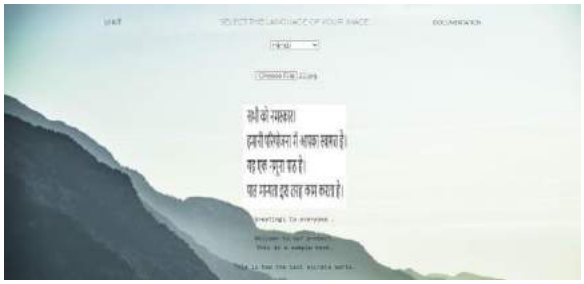

Fig.11. Translation of Hindi text into English

I. Saving the file into .docx format

The recognized text is saved into .docx format after translation. Fig 12. shows saving recognized text into a document for future reference.

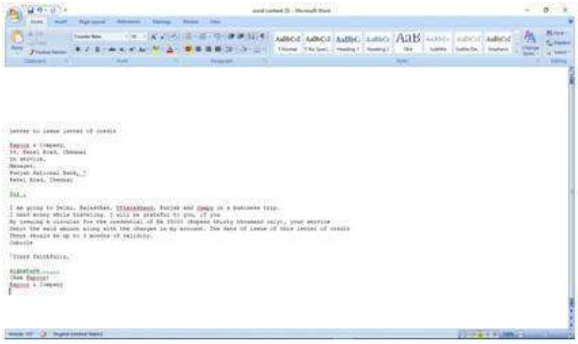

Fig. 12 Saving recognized text into .docx format

## VI. CONCLUSION AND FUTURE SCOPE

During the accuracy testing of Tesseract OCR, the following observations were repeated: A character for a blank space is recorded. Different letters are detected in various ways. Since Tesseract OCR's algorithm has been tested on millions of images, these cases may be the consequence of computational complexity. We worked towards our project in all seven languages. The best results were obtained for Hindi, which received 95% accuracy, followed by Kannada with 76%, Marathi with 68% and Malayalam with 66%, then Bengali, Tamil and Telugu with 58%, 57%, 43% respectively. With the will complexity of various cases in the future, accuracy could enhance. The softcopy could be easily understood and shared amongst users due to the inclusion of a Translation API and the ability to save the text in.doc format. Each language could be improved with different training algorithms with the assistance of language experts. Adding more functionality to a translation API, this programme could well operate on a phone browser, turning it into a mobile application for a quicker and more efficient flow of documents.

REFERENCES

[1] Akhil S, "Overview of Tesseract OCR engine", a seminar report in April 2017.

[2] Munish Kumar, M.K. Jindal, and R.K. Sharma, "Review on OCR for Handwritten Indian Scripts Character Recognition", in International Conference on Digital Image Processing and Information Technology at CCIS Springer-Verlag Berlin Heidelberg 2011.

[3] Sagar Patil, Mayuri Phonde, Siddharth Prajapati, Sarange Rane, "Multilingual speech and text recognition and translation using image", International Journal of Engineering Research & Technology (ILERT), Vol. 6 Issue 04,April-2016.

[4] Karez Abdulwahhab Hamad, Mehmet Kaya, "A Detailed Analysis of Optical Character Recognition Technology", in the International Journal of Applied Mathematics,Electronics and Computer in 2016.

[5] Sahil Thakare, Ajay Kamble, Vishal Thengne, Mrs U.R.Kamble, "Document Segmentation and Language Translation Using Tesseract-OCR", 2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS).

[6] K. Elissa, Hiral Modi,M.C.parikh, "A Review On Optical Character Recognition Techniques", International Journal of Computer Application,2017.

[7]The Tesseract open source OCR engine, https://github.com/naptha/tesseract.js#tesseractjs

[8]https://www.codexworld.com/export-html-to-word -doc-docx-using-javascript/