Report on Internship Project

UNDERGONE IN OCR BESTENLIST

(Optical Character Recognition in Multilingual Text)

SUBMITTED BY

M.SIVAGANESH

ABSTRACT :

Optical character recognition (OCR) method has been used in converting printed text into editable text. OCR is very useful and popular method in various applications. Accuracy of OCR can be dependent on text preprocessing and segmentation algorithms. Sometimes it is difficult to retrieve text from the image because of different size, style, orientation, complex background of image etc. We begin this paper with an introduction of Optical Character Recognition (OCR) method, History of Open Source OCR tool Tesseract, architecture of it and experiment result of OCR performed by Tesseract on different kinds images are discussed. We conclude this paper by comparative study of this tool with other commercial OCR tool Transym OCR by considering vehicle number plate as input. From vehicle number plate we tried to extract vehicle number by using Tesseract and Transym and compared these tools based on various parameters.

INTRODUCTION :

Humans can understand the contents of an image simply by looking. We perceive the text on the image as text and can read it. Computers don't work the same way. They need something more concrete, organized in a way they can understand. This is where *Optical Character Recognition* (OCR) kicks in. Whether it's recognition of car plates from a camera, or hand-written documents that should be converted into a digital copy, this technique is very useful. While it's not always perfect, it's very convenient and makes it a lot easier and faster for some people to do their jobs. In this article, we will delve into the depth of Optical Character Recognition and its application areas. We will also build a simple script in Python that will help us detect characters from images and expose this through a Flask application for a more convenient interaction medium.

PACKAGE :

Pip install tesseract

Pip install pdf2image

Pip install pillow

REQUIREMENT :

Anaconda Enviroment (python 3)

Jupyter notebook

Tesseract :

Python-tesseract is an optical character recognition (OCR) tool for python. That is, it will recognize and “read” the text embedded in images. ... Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file.

Pillow :

Pillow is a Python Imaging Library (PIL), which adds support for opening, manipulating, and saving images. The current version identifies and reads a large number of formats. Write support is intentionally restricted to the most commonly used interchange and presentation formats.

Optical Character Recognition (OCR):

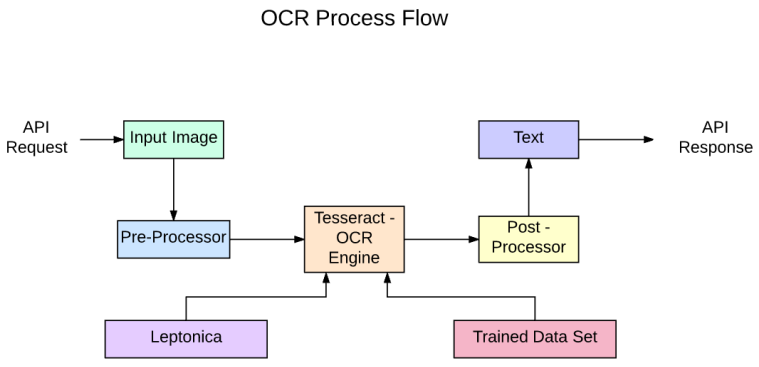
Optical Character Recognition involves the detection of text content on images and translation of the images to encoded text that the computer can easily understand. An image containing text is scanned and analyzed in order to identify the characters in it. Upon identification, the character is converted to machine-encoded text.

The image is first scanned and the text and graphics elements are converted into a bitmap, which is essentially a matrix of black and white dots. The process may not be 100% accurate and might need human intervention to correct some elements that were not scanned correctly. Error correction can also be achieved using a dictionary or even Natural Language Processing (NLP). The output can now be converted to other mediums such as word documents, PDFs, or even audio content through text-to-speech technologies.

Tesseract :

Python-tesseract is an optical character recognition (OCR) tool for python. That is, it will recognize and “read” the text embedded in images. Python-tesseract is a wrapper for Google’s Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported by the Pillow and Leptonica imaging libraries, including jpeg, png, gif, bmp, tiff, and others. Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file.

Tesseract is an open source text recognition (OCR) Engine, available under the Apache 2.0 license. It can be used directly, or (for programmers) using an API to extract printed text from images. It supports a wide variety of languages. Tesseract doesn't have a built-in GUI, but there are several available from the 3rdParty page. Tesseract is compatible with many programming languages and frameworks through wrappers that can be found here. It can be used with the existing layout analysis to recognize text within a large document, or it can be used in conjunction with an external text detector to recognize text from an image of a single text line.



Working of Tesseract :

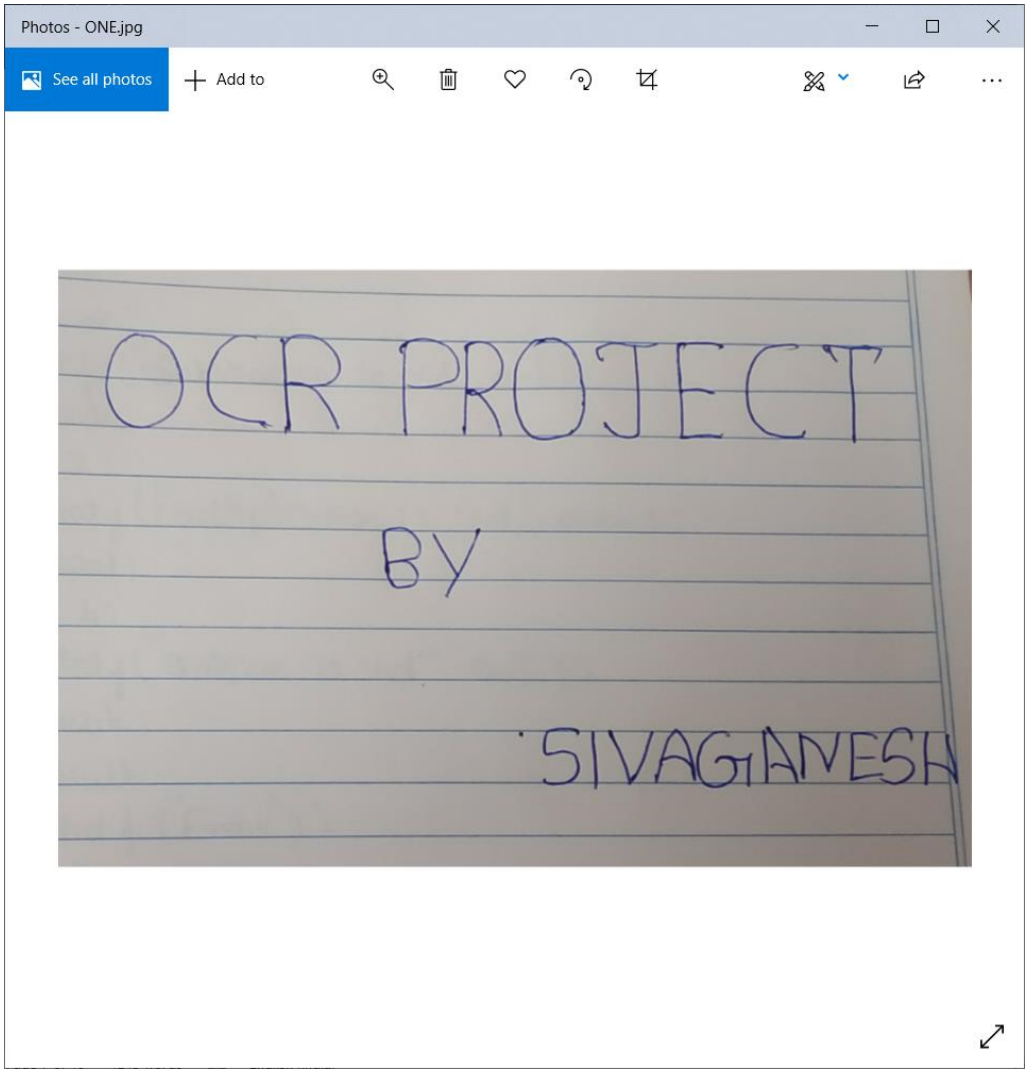
An image with the text is given as input to the Tesseract engine that is command based tool. Tesseract command takes two arguments: First argument is image file name that contains text and second argument is output text file in which, extracted text is stored. The output file extension is given as .txt by Tesseract, so no need to specify the file extension while specifying the output file name as a second argument in Tesseract command.

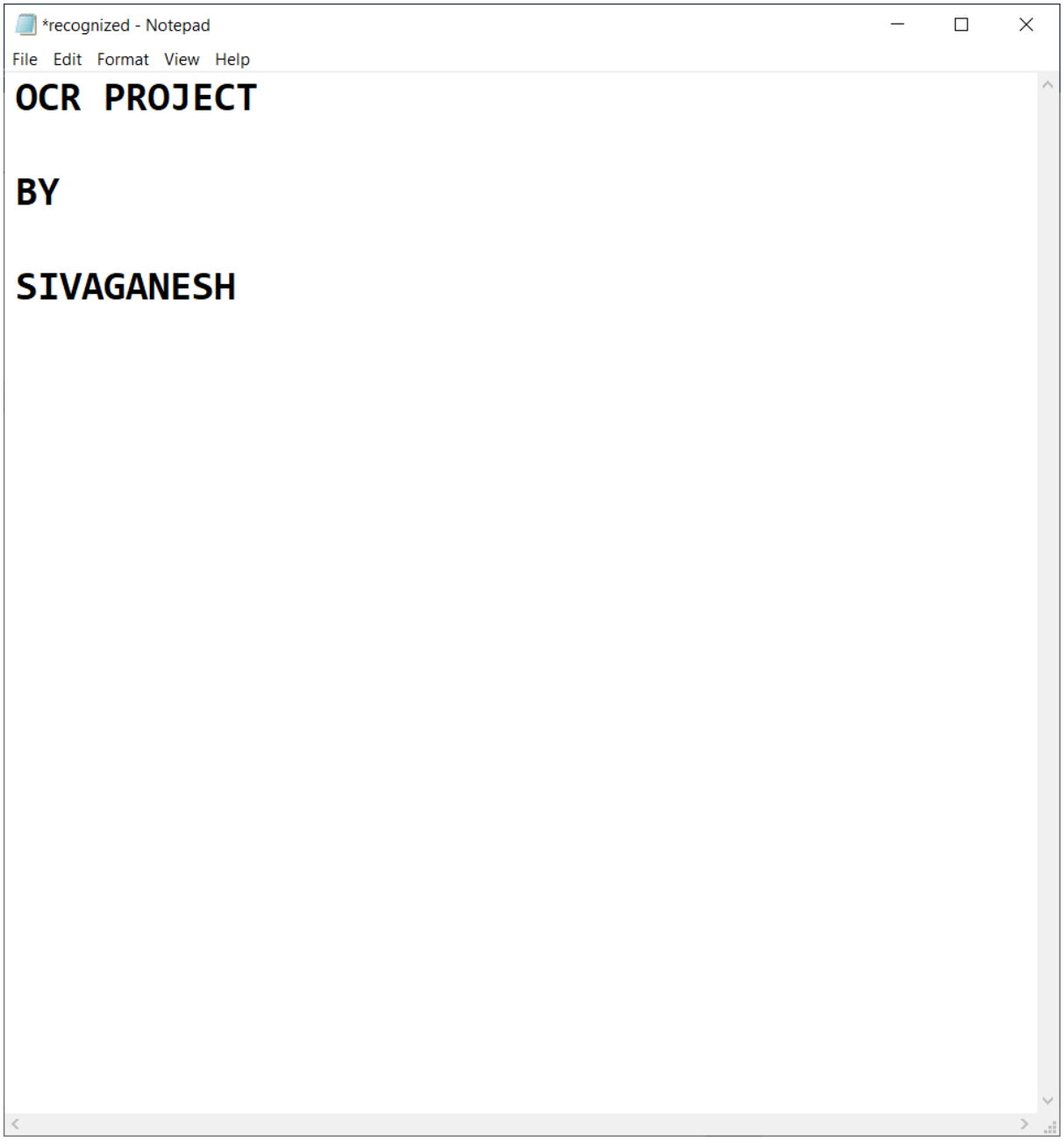
As Tesseract supports various languages, the language training data file must be kept in the tessdata folder. In this research, the purpose is to extract English text from the images so we have kept only English language file in the tessdata folder. After processing is completed, the content of the output file shown in fig 4. In simple images with or without color (gray scale), Tesseract provides results with 100% accuracy. But in the case of some complex images Tesseract provides better accuracy results if the images are in the gray scale mode as compared to color images. To prove this hypothesis, OCR of same color images and gray scale images is performed and in both cases different result are achieved.

SOURCE CODE :

```
 import pytesseract
pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract'
text=pytesseract.image_to_string(r"C:\Users\subhasuji\Desktop\first\ppo.jpg")
f = open("recognized.txt", "w")
f.write(text)
f.close()
```

OUTPUT :




*recognized - Notepad
File Edit Format View Help
OCR PROJECT
BY
SIVAGANESH

CONCLUSION :

Although Tesseract is command-based tool but as it is open source and it is available in the form of Dynamic Link Library, it can be easily made available in graphics mode. The results obtained in above sections are obtained by extracting vehicle number from vehicle number plate. So above results do not confirm that Tesseract is always better or faster than Transym but is it more accurate in extracting text from the vehicle number plate. The input images are specific, which are vehicle number plates, so in these specific images Tesseract provides better accuracy and in other kinds on images Transym might provide better accuracy than Tesseract. As we are interested in extracting vehicle number from vehicle number plate, we have considered both tools for serving this specific purpose.

REFERENCES :

[1] ARCHANA A. SHINDE, D. 2012.Text Pre-processing

and Text Segmentation for OCR. International Journal of Computer Science Engineering and Technology, pp. 810- 812.

[2] ANAGNOSTOPOULOS,C.,ANAGNOSTOPOULOS, I., LOUMOS, V, & KAYAFAS, E. 2006. A License Plate Recognition Algorithm for Intelligent Transportation System Applications.., IEEE Transactions on Intelligent Transportation Systems, pp. 377- 399.

[3] Y. WEN, Y. L. 2011. An Algorithm for License Plate

Recognition Applied to Intelligent Transportation System., IEEE Transactions on Intelligent Systems, pp. 1-16.

[4] XIN FAN, G. L. 2009. Graphical Models for Joint Segmentation and Recognition of License Plate Characters. IEEE Signal Processing Letters, pp. 10-13.

[5] HUI WU, B. L. 2011. License Plate Recognition system. International Conference on Multimedia Technology (ICMT). pp. 5425 - 5427.

[6] PAN, Y.-F., HOU, X., & LIU, C.-L. 2008.A Robust System to Detect and Localize Texts in Natural Scene Images.The Eighth IAPR International Workshop on Document Analysis Systems.