Extract Text From Image Using Python With Code

Table of Content

Introduction

This article is a how-to guide/ tutorial on how to implement image Optical Character Recognition (OCR) in python using the Tesseract engine.

Tesseract is an open-source OCR engine that allows to extract text from images.

Extracting text from images is a very popular task in the operations units of the business (extracting information from invoices and receipts) as well as in other areas.

OCR is an electronic computer-based approach to convert images with text into machine-encoded text, which can then be extracted and used in text format.

To continue following this tutorial we will need the following modules:

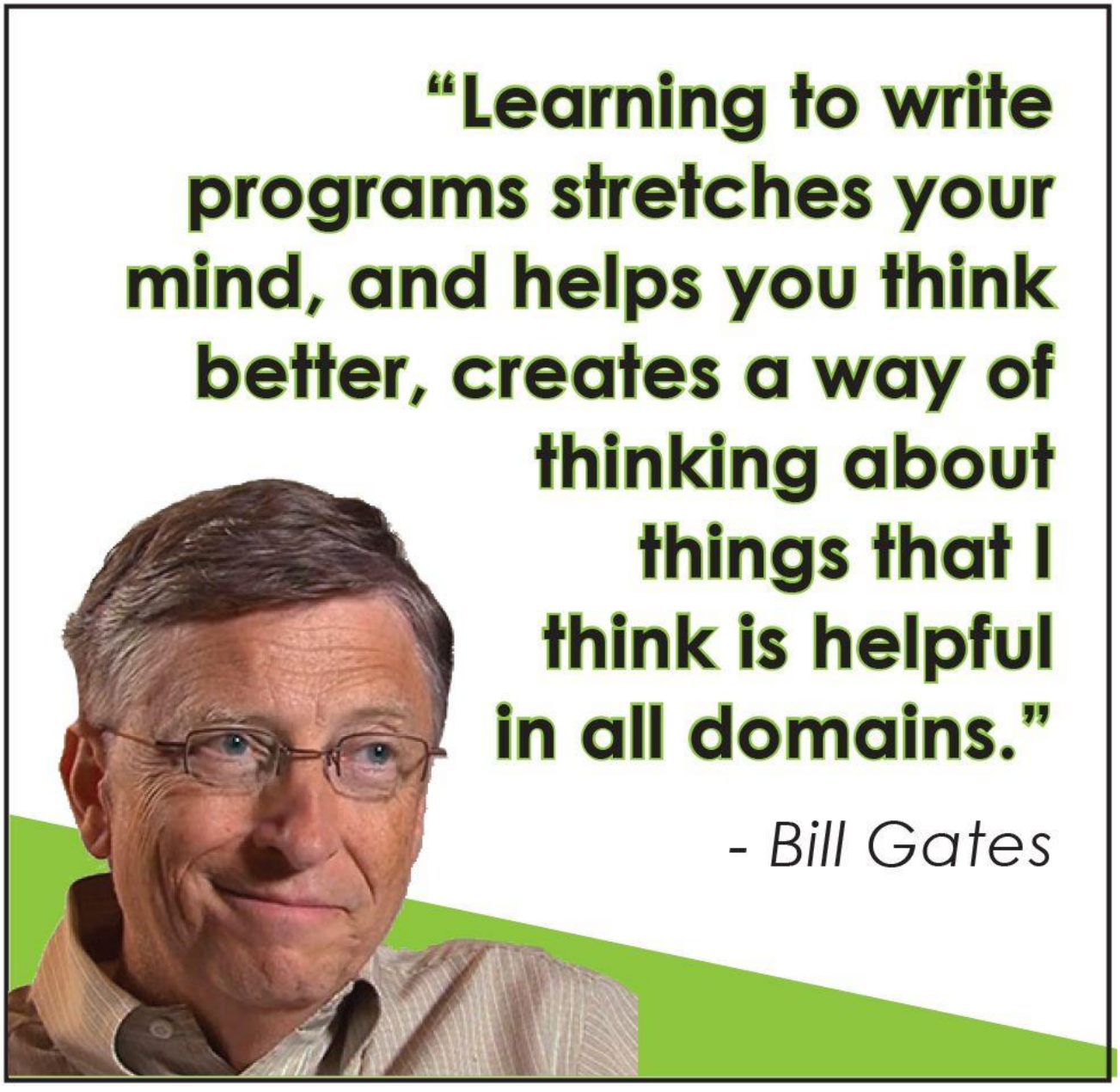
- Tesseract for Windows
- Python libraries: pytesseract, pillow, and OpenCV

Tutorial Requirements

This tutorial requires having Tesseract OCR installed on Windows. To install Tesseract on Windows, please follows the directions from the tutorial:

How to Install and Run Tesseract OCR for Windows in 4 Easy Steps.

Image Used to Extract Text



Extract Text From Image Using Tesseract and Pillow Code

```python
# Import required libraries.
from PIL import Image
from pytesseract import pytesseract, TesseractError

# If you don't have tesseract executable in your PATH, then
# define the path to Tesseract.exe.
path_to_tesseract = r'C:\Program Files\Tesseract-OCR\tesseract.exe'

# Define path to image
path_to_image = 'Images/bill gate quote.jpg'

# Point tesseract_cmd to tesseract.exe
pytesseract.tesseract_cmd = path_to_tesseract

# Open image with PIL
img = Image.open(path_to_image)

# Extract text from image
try:
    text = pytesseract.image_to_string(img, lang='eng')
    print(text)
except TesseractError as err:
    print(err.args[1])
except RuntimeError as err:
    print(err.args[1])
```

Extract Text From Image Using Tesseract and OpenCV Code

```python
# Import required libraries.
import cv2
from pytesseract import pytesseract, TesseractError

# If you don't have tesseract executable in your PATH, then
# define the path to Tesseract.exe.
path_to_tesseract = r'C:\Program Files\Tesseract-OCR\tesseract.exe'

# Define path to image.
path_to_image = 'Images/bill gate quote.jpg'

# Point tesseract_cmd to tesseract.exe
pytesseract.tesseract_cmd = path_to_tesseract

# Open image using OpenCV.
img = cv2.imread(path_to_image)

# Extract text from image.
try:
    text = pytesseract.image_to_string(img, lang='eng')
    print(text)
except TesseractError as err:
    print(err.args[1])
except RuntimeError as err:
    print(err.args[1])
```

Text Extracted From Image Using Tesseract with Pillow or OpenCV

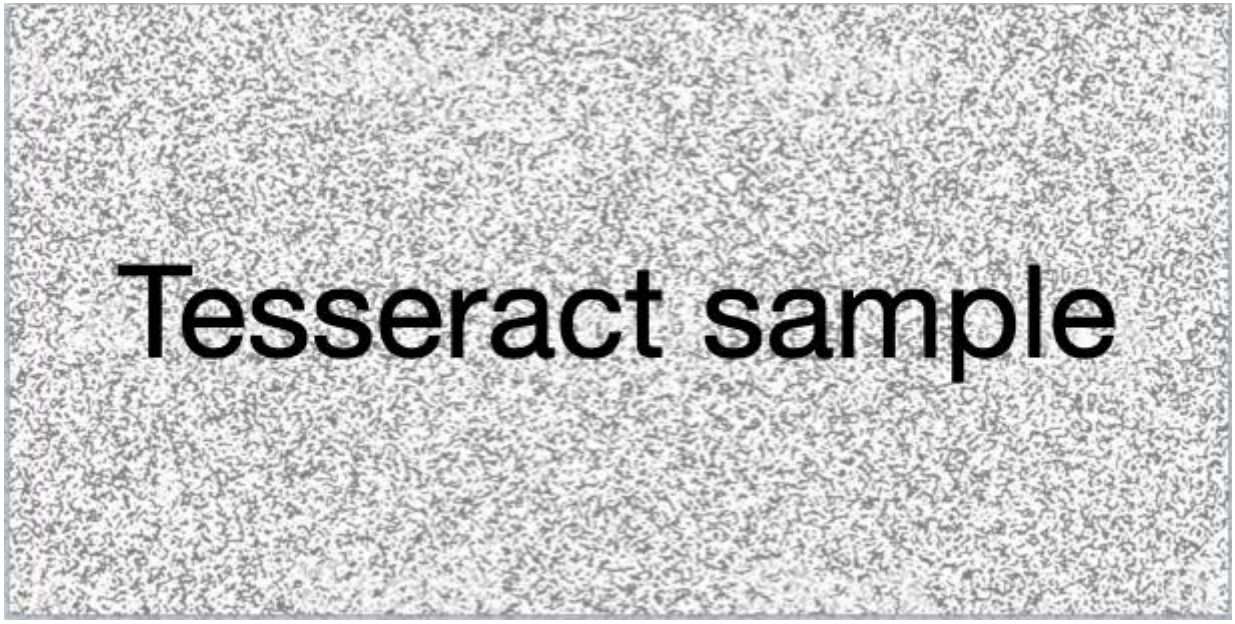
“Learning to write
programs stretches your
mind, and helps you think
better, creates a way of
thinking about

things that |
think is helpful
in all domains.”

- Bill Gates

Error. Instead of an “I”, we got “|“

Noise Image Sample



If we use the previous code to extract the text from this image, the result will be empty. It is not possible to extract text from this image without pre-processing the image before passing it to Tesseract.

To avoid all the ways that tesseract output accuracy can drop, you need to make sure the image is appropriately pre-processed.

Preprocessing the Image for Tesseract Code

```python
# Import required libraries.
import cv2
from pytesseract import pytesseract, TesseractError

# If you don't have tesseract executable in your PATH, then
# define the path to Tesseract.exe.
path_to_tesseract = r'C:\Program Files\Tesseract-OCR\tesseract.exe'

# Define path to image.
path_to_image = 'Images/tesseract.png'

# Point tesseract_cmd to tesseract.exe
pytesseract.tesseract_cmd = path_to_tesseract

# Open image using OpenCV.
image = cv2.imread(path_to_image)

# Apply threshold to image.
image2 = cv2.threshold(image, 100, 255, cv2.THRESH_BINARY)[1]

# Extract text from image.
try:
    text = pytesseract.image_to_string(image2, lang='eng')
    print(text)
except TesseractError as err:
    print(err.args[1])
except RuntimeError as err:
    print(err.args[1])
```

Preprocessing the Image

Comparing Preprocessed Image with Normal Image

Conclusion

Tesseract works best when there is a clear segmentation of the foreground text from the background. In practice, it can be extremely challenging to guarantee these types of setups. There are a variety of reasons you might not get good quality output from Tesseract like if the image has noise in the background. The better the image quality (size, contrast, lightning) the better the recognition result. It requires a bit of preprocessing to improve the OCR results, images need to be scaled appropriately, have as much image contrast as possible, and the text must be horizontally aligned.

References

1. How to OCR with Tesseract, OpenCV, and Python
https://nanonets.com/blog/ocr-with-tesseract/

2. Extract Text from the Image using Python
https://python-bloggers.com/2022/05/extract-text-from-image-using-python/

3. Reading Text from the Image using Tesseract
https://www.geeksforgeeks.org/reading-text-from-the-image-using-tesseract/