

## Project Report 18

### Text Detection

**Name :DHIVAKAR.R**

**COURSE: AI and ML(Aug 2020)**

Question: Implement a text detection and extraction model using OpenCV and OCR. The necessary steps that you need to perform are:

1. Image preprocessing
2. Find possible contours that can represent the textual areas.
3. Apply optical character recognition (using python-tesseract, google OCR engine.

#### Prerequisites

What things you need to install the software and how to install them:

Python 3.6 This setup requires that your machine has latest version of python. The following url <https://www.python.org/downloads/> can be referred to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: <https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-externalcommand/> . Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic.

Second and easier option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url <https://www.anaconda.com/download/> You will also need to download and install below 3 packages after you install either python or anaconda from the steps above Sklearn (scikit-learn) numpy scipy if you have chosen to install python 3.6 then run below commands in command prompt/terminal to install these packages `pip install -U scikit-learn` `pip install numpy` `pip install scipy` if you have chosen to install anaconda then run below commands in anaconda prompt to install these packages `conda install -c scikit-learn` `conda install -c anaconda numpy` `conda install -c anaconda scipy` .

Importing the required libraires

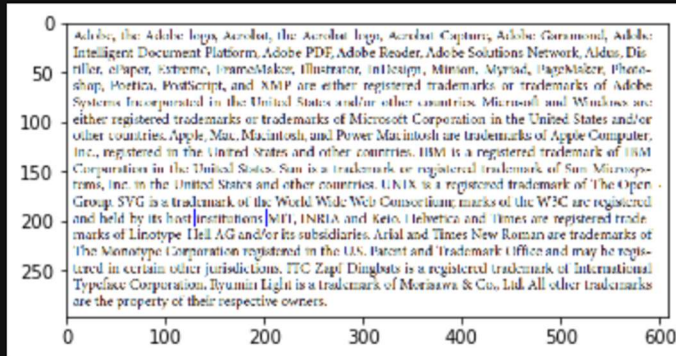
```
[11]: import pytesseract
      pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract.exe'
      import cv2
      import matplotlib.pyplot as plt
```

## Reading the image

```
[12]: ## reading the image
img = cv2.imread('promo.png')

plt.imshow(img)
```

```
[12]: <matplotlib.image.AxesImage at 0x20d639a8748>
```



## Getting the text from the image

```
[13]: text = pytesseract.image_to_string(img)
print(text)
```

Adobe, the Adobe logo, Acrobat, the Acrobat logo, Acrobat Capture, Adobe Garamond, Adobe Intelligent Document Platform, Adobe PDF, Adobe Reader, Adobe Solutions Network, Aldus, Distiller, ePaper, Extreme, FrameMaker, Illustrator, InDesign, Minion, Myriad, PageMaker, PhotoShop, Poetica, PostScript, and XMP are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Apple, Mac, Macintosh, and Power Macintosh are trademarks of Apple Computer, Inc., registered in the United States and other countries. IBM is a registered trademark of IBM Corporation in the United States. Sun is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. UNIX is a registered trademark of The Open Group. SVG is a trademark of the World Wide Web Consortium; marks of the W3C are registered and held by its host institutions MIT, INRIA and Keio. Helvetica and Times are registered trademarks of Linotype-Hell AG and/or its subsidiaries. Arial and Times New Roman are trademarks of The Monotype Corporation registered in the US. Patent and Trademark Office and may be registered in certain other jurisdictions. ITC Zapf Dingbats is a registered trademark of International Typeface Corporation. Ryumin Light is a trademark of Morisawa & Co., Ltd. All other trademarks are the property of their respective owners.

## Image to boxes

```
[14]: imgbox = pytesseract.image_to_boxes(img)
      print(imgbox)
```

```
A 7 278 25 289 0
d 17 276 25 289 0
o 25 278 32 285 0
b 32 276 37 289 0
e 33 276 50 289 0
, 44 276 49 289 0
t 55 278 67 289 0
h 63 278 69 289 0
e 68 278 74 285 0
A 82 278 90 289 0
d 79 278 104 289 0
```

```
[15]: type(imgbox)
```

```
[15]: str
```

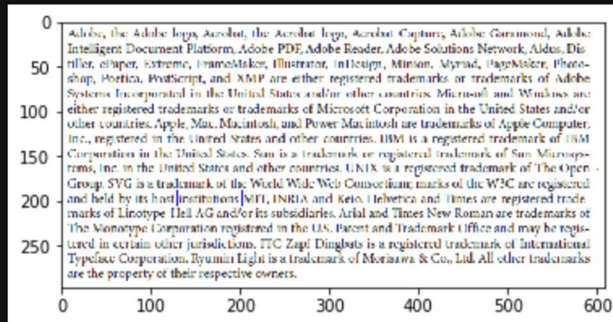
```
[16]: imgH, imgW, _ = img.shape
      img.shape
```

```
[16]: (297, 609, 3)
```

```
[17]: for boxes in imgbox.splitlines():
      boxes = boxes.split(' ')
      x,y,w,h = int(boxes[1]), int(boxes[2]), int(boxes[3]), int(boxes[4])
      cv2.rectangle(img, (x, imgH-y), (w, imgH-h), (0,0,255), 10)
```

```
[18]: plt.imshow(img1)
```

```
[18]: <matplotlib.image.AxesImage at 0x20d63960048>
```



```
[19]: plt.imshow(cv2.cvtColor(img1, cv2.COLOR_BGR2RGB))
```

```
[19]: <matplotlib.image.AxesImage at 0x20d633cd888>
```

