Guided Assignment On Supervised Learning

Name: **Shreya Shrivastava**

Course: **Artificial Intelligence And Machine Learning**

Batch Four

Duration: 12 Months

Problem statement: 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:

The libraries as well as things required in order for the program to work:

I. **Python 3.6** : 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-external-command/. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic. Second 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/

II. **OpenCV :** OpenCV can be downloaded from the following url: https://sourceforge.net/projects/opencvlibrary/. It is strongly recommended to download OpenCV in a virtual environment.

III. **ADDITIONAL PACKAGES :** You will also need to download and install below 2 packages- numpy and pytesseract after you install either python or anaconda from the steps above. If you have chosen to install python 3.6,then run the following commands in command prompt/terminal to install these packages :

NUMPY: pip install -U numpy

PYTESSERACT: pip install -U pytesseract

If using Anaconda then run the following commands in anaconda prompt to install these packages:

NUMPY: conda install -c anaconda numpy

PYTESSERACT: conda install -c pytesseract

IV. METHODS USED:

A. OTSU THRESHOLDING

B. OPTICAL CHARACTER RECOGNITION

THE PROJECT :

1. Importing the libraries, loading the image, pre-processing it and creating mask.

```
import cv2
import numpy as np
import pytesseract

# Load image, create mask, grayscale, Otsu's threshold
image = cv2.imread('/Users/shreyashrivastava/Desktop/sample.png')
mask = np.zeros(image.shape, dtype=np.uint8)
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)[1]
```

2. Finding contours and also getting the information about the image by using the image_to_data function.

```
# Filter for ROI using contour area and aspect ratio
cnts = cv2.findContours(thresh, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
cnts = cnts[0] if len(cnts) == 2 else cnts[1]
custom_config = r'--oem 3 --psm 6'
details = pytesseract.image_to_data(thresh, output_type=pytesseract.Output.DICT, config=custom_config, lang='eng')
```

3.Performing OCR.

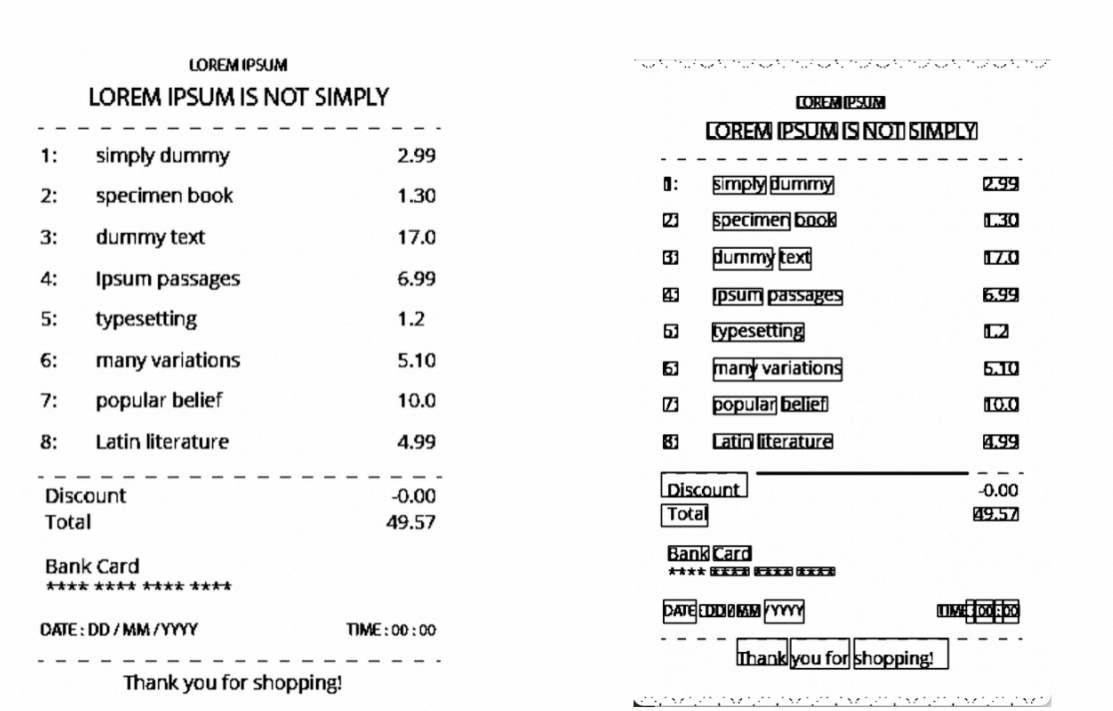
```
total_boxes = len(details['text'])
for i in range(total_boxes):
    if int(details['conf'][i])>5:
        (x, y, w, h) = (details['left'][i], details['top'][i],
                        details['width'][i],  details['height'][i])
        thresh = cv2.rectangle(thresh, (x, y), (x + w, y + h), (0, 255, 0), 3)
        cv2.imshow('captured text', thresh)
        cv2.waitKey(0)
        cv2.destroyAllWindows()
        cv2.waitKey(1)
```

4. Storing the text in the file.

```
for c in cnts:
    area = cv2.contourArea(c)
    peri = cv2.arcLength(c, True)
    approx = cv2.approxPolyDP(c, 0.05 * peri, True)
    x,y,w,h = cv2.boundingRect(approx)
    aspect_ratio = w / float(h)
    if area > 2000 and aspect_ratio > .5:
        mask[y:y+h, x:x+w] = image[y:y+h, x:x+w]


data = pytesseract.image_to_string(mask, lang='eng', config='--psm 6')
print(data)
with open('result_text.txt',  'w', newline="") as file:
    file.write(data)
```

5. FINAL RESULTS.



ORIGINAL IMAGE VS AFTER PERFORMING OCR

```
LOREM IPSUM
LOREM IPSUM IS NOT SIMPLY

1: simply dummy 2.99
2: specimen book 1.30
3: dummy text 17.0
4: Ipsum passages 6.599
5: typesetting 1.2
6: many variations 5.10
7: popular belief 10.0
8: Latin literature 4.59
Discount SSS
Total 49.57
Bank Card



DATE :DD / MM /YVYY TIME : 00 : 00
Thank you for shopping!
```