import cv2

import numpy as np

import matplotlib.pyplot as plt

%matplotlib inline

import pytesseract

img=cv2.imread('C:/Users/yashb/Pictures/license plates/1.jpg')

img=cv2.cvtColor(img,cv2.COLOR\_BGR2RGB)

img=cv2.resize(img,(650,480))

plt.imshow(img)

img\_gray=cv2.imread('C:/Users/yashb/Pictures/license plates/1.jpg',cv2.IMREAD\_GRAYSCALE)

img\_gray=cv2.resize(img\_gray,(650,480))

plt.imshow(img\_gray,cmap='gray')

med\_val=np.median(img\_gray)

med\_val

gray\_blur = cv2.bilateralFilter(img\_gray,15,100,100)

lower=int(max(0,0.7\*med\_val))

upper=int(min(255,1.3\*med\_val))

edges=cv2.Canny(gray\_blur,lower,upper+100)

plt.imshow(edges,cmap='gray')

contours,hierarchy =cv2.findContours(edges.copy(),cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)

contours = sorted(contours,key=cv2.contourArea, reverse = True)[:30]

for contour in contours:

peri = cv2.arcLength(contour, True)

approx = cv2.approxPolyDP(contour, 0.01 \* peri, True)

if len(approx) == 4:

contour\_with\_license\_plate= approx

x,y,w,h=cv2.boundingRect(approx)

lp\_detected=cv2.rectangle(img.copy(),(x,y),(x+w,y+h),(0,255,0),5)

plt.imshow(lp\_detected,cmap='gray')

license\_plate=img.copy()[y:y+h,x:x+w]

break

print("\n")

plt.imshow(license\_plate)

license\_plate = cv2.bilateralFilter(license\_plate.copy(),30,100,100)

plt.imshow(license\_plate)

text=pytesseract.image\_to\_string(license\_plate)

print(text)