

b3dhgnef4

April 13, 2023

```
[1]: # install and Import Dependencies
```

```
!pip install easyocr
```

```
!pip install imutils
```

Requirement already satisfied: easyocr in c:\users\mansi\anaconda3\lib\site-packages (1.6.2)

Requirement already satisfied: torch in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (2.0.0)

Requirement already satisfied: scipy in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.7.1)

Requirement already satisfied: pyclicker in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.3.0.post4)

Requirement already satisfied: PyYAML in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (6.0)

Requirement already satisfied: Shapely in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (2.0.1)

Requirement already satisfied: numpy in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.22.4)

Requirement already satisfied: ninja in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.11.1)

Requirement already satisfied: torchvision>=0.5 in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (0.15.1)

Requirement already satisfied: python-bidi in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (0.4.2)

Requirement already satisfied: Pillow in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (8.4.0)

Requirement already satisfied: opencv-python-headless<=4.5.4.60 in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (4.5.4.60)

Requirement already satisfied: scikit-image in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (0.18.3)

Requirement already satisfied: requests in c:\users\mansi\anaconda3\lib\site-packages (from torchvision>=0.5->easyocr) (2.26.0)

Requirement already satisfied: networkx in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (2.6.3)

Requirement already satisfied: jinja2 in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (2.11.3)

Requirement already satisfied: typing-extensions in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (3.10.0.2)

Requirement already satisfied: sympy in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (1.9)

Requirement already satisfied: filelock in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (3.3.1)

Requirement already satisfied: MarkupSafe>=0.23 in c:\users\mansi\anaconda3\lib\site-packages (from jinja2->torch->easyocr) (1.1.1)

Requirement already satisfied: six in c:\users\mansi\anaconda3\lib\site-packages (from python-bidi->easyocr) (1.16.0)

Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\mansi\anaconda3\lib\site-packages (from requests->torchvision>=0.5->easyocr) (2.0.4)

Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\mansi\anaconda3\lib\site-packages (from requests->torchvision>=0.5->easyocr) (1.26.7)

Requirement already satisfied: certifi>=2017.4.17 in c:\users\mansi\anaconda3\lib\site-packages (from requests->torchvision>=0.5->easyocr) (2021.10.8)

Requirement already satisfied: idna<4,>=2.5 in c:\users\mansi\anaconda3\lib\site-packages (from requests->torchvision>=0.5->easyocr) (3.2)

Requirement already satisfied: matplotlib!=3.0.0,>=2.0.0 in c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (3.4.3)

Requirement already satisfied: imageio>=2.3.0 in c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (2.9.0)

Requirement already satisfied: tifffile>=2019.7.26 in c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (2021.7.2)

Requirement already satisfied: PyWavelets>=1.1.1 in c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (1.1.1)

Requirement already satisfied: cycler>=0.10 in c:\users\mansi\anaconda3\lib\site-packages (from matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (0.10.0)

Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\mansi\anaconda3\lib\site-packages (from matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (1.3.1)

Requirement already satisfied: pyparsing>=2.2.1 in c:\users\mansi\anaconda3\lib\site-packages (from matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (3.0.4)

Requirement already satisfied: python-dateutil>=2.7 in c:\users\mansi\anaconda3\lib\site-packages (from matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (2.8.2)

Requirement already satisfied: mpmath>=0.19 in c:\users\mansi\anaconda3\lib\site-packages (from sympy->torch->easyocr) (1.2.1)

Requirement already satisfied: imutils in c:\users\mansi\anaconda3\lib\site-packages (0.5.4)

```
[2]: pip install easyocr
```

Requirement already satisfied: easyocr in c:\users\mansi\anaconda3\lib\site-packages (1.6.2)

Requirement already satisfied: ninja in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.11.1)

Requirement already satisfied: pyclipper in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.3.0.post4)

Requirement already satisfied: torchvision>=0.5 in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (0.15.1)

Requirement already satisfied: numpy in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.22.4)

Requirement already satisfied: scikit-image in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (0.18.3)

Requirement already satisfied: Pillow in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (8.4.0)

Requirement already satisfied: torch in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (2.0.0)

Requirement already satisfied: PyYAML in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (6.0)

Requirement already satisfied: Shapely in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (2.0.1)

Requirement already satisfied: python-bidi in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (0.4.2)

Requirement already satisfied: opencv-python-headless<=4.5.4.60 in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (4.5.4.60)

Requirement already satisfied: scipy in c:\users\mansi\anaconda3\lib\site-packages (from easyocr) (1.7.1)

Requirement already satisfied: requests in c:\users\mansi\anaconda3\lib\site-packages (from torchvision>=0.5->easyocr) (2.26.0)

Requirement already satisfied: sympy in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (1.9)

Requirement already satisfied: filelock in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (3.3.1)

Requirement already satisfied: jinja2 in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (2.11.3)

Requirement already satisfied: typing-extensions in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (3.10.0.2)

Requirement already satisfied: networkx in c:\users\mansi\anaconda3\lib\site-packages (from torch->easyocr) (2.6.3)

Requirement already satisfied: MarkupSafe>=0.23 in c:\users\mansi\anaconda3\lib\site-packages (from jinja2->torch->easyocr) (1.1.1)

Requirement already satisfied: six in c:\users\mansi\anaconda3\lib\site-packages (from python-bidi->easyocr) (1.16.0)

Requirement already satisfied: charset-normalizer~2.0.0 in c:\users\mansi\anaconda3\lib\site-packages (from requests->torchvision>=0.5->easyocr) (2.0.4)

Requirement already satisfied: idna<4,>=2.5 in c:\users\mansi\anaconda3\lib\site-packages (from requests->torchvision>=0.5->easyocr) (3.2)

Requirement already satisfied: certifi>=2017.4.17 in
c:\users\mansi\anaconda3\lib\site-packages (from
requests->torchvision>=0.5->easyocr) (2021.10.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in
c:\users\mansi\anaconda3\lib\site-packages (from
requests->torchvision>=0.5->easyocr) (1.26.7)
Requirement already satisfied: matplotlib!=3.0.0,>=2.0.0 in
c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (3.4.3)
Requirement already satisfied: imageio>=2.3.0 in
c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (2.9.0)
Requirement already satisfied: tifffile>=2019.7.26 in
c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr)
(2021.7.2)
Requirement already satisfied: PyWavelets>=1.1.1 in
c:\users\mansi\anaconda3\lib\site-packages (from scikit-image->easyocr) (1.1.1)
Requirement already satisfied: kiwisolver>=1.0.1 in
c:\users\mansi\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (1.3.1)
Requirement already satisfied: python-dateutil>=2.7 in
c:\users\mansi\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (2.8.2)
Requirement already satisfied: pyparsing>=2.2.1 in
c:\users\mansi\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (3.0.4)
Requirement already satisfied: cyclor>=0.10 in
c:\users\mansi\anaconda3\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.0->scikit-image->easyocr) (0.10.0)
Requirement already satisfied: mpmath>=0.19 in
c:\users\mansi\anaconda3\lib\site-packages (from sympy->torch->easyocr) (1.2.1)
Note: you may need to restart the kernel to use updated packages.

[3]: `pip install torch`

Requirement already satisfied: torch in c:\users\mansi\anaconda3\lib\site-
packages (2.0.0)
Requirement already satisfied: sympy in c:\users\mansi\anaconda3\lib\site-
packages (from torch) (1.9)
Requirement already satisfied: filelock in c:\users\mansi\anaconda3\lib\site-
packages (from torch) (3.3.1)
Requirement already satisfied: Jinja2 in c:\users\mansi\anaconda3\lib\site-
packages (from torch) (2.11.3)
Requirement already satisfied: networkx in c:\users\mansi\anaconda3\lib\site-
packages (from torch) (2.6.3)
Requirement already satisfied: typing-extensions in
c:\users\mansi\anaconda3\lib\site-packages (from torch) (3.10.0.2)
Requirement already satisfied: MarkupSafe>=0.23 in
c:\users\mansi\anaconda3\lib\site-packages (from Jinja2->torch) (1.1.1)
Requirement already satisfied: mpmath>=0.19 in

c:\users\manshi\anaconda3\lib\site-packages (from sympy->torch) (1.2.1)
Note: you may need to restart the kernel to use updated packages.

```
[4]: import cv2
      from matplotlib import pyplot as plt
      import numpy as np
      import imutils
      import easyocr
```

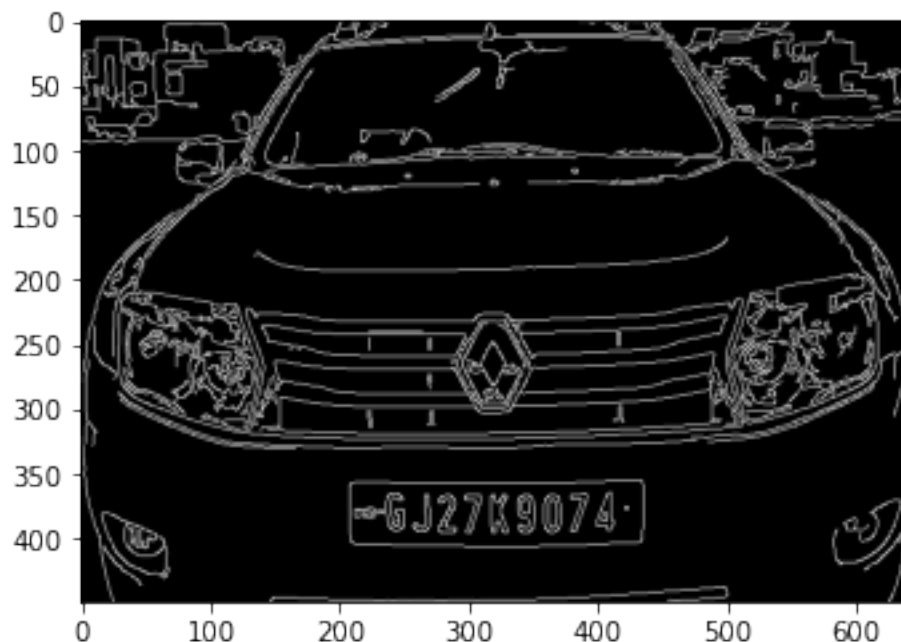
```
[5]: # Read in image, Grayscale and Blur
      img = cv2.imread('wheeler.jpg')
      gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
      plt.imshow(cv2.cvtColor(gray, cv2.COLOR_BGR2RGB))
```

[5]: <matplotlib.image.AxesImage at 0x1fad5b78be0>



```
[6]: # Apply filter and find edges for localization
      bfilter = cv2.bilateralFilter(gray, 11, 17, 17) #Noise reduction
      edged = cv2.Canny(bfilter, 30, 200) #Edge detection
      plt.imshow(cv2.cvtColor(edged, cv2.COLOR_BGR2RGB))
```

[6]: <matplotlib.image.AxesImage at 0x1fad7c4ed60>



```
[7]: # Find Contours and Apply Mask
keypoints = cv2.findContours(edged.copy(), cv2.RETR_TREE, cv2.
    ↳CHAIN_APPROX_SIMPLE)
contours = imutils.grab_contours(keypoints)
contours = sorted(contours, key=cv2.contourArea, reverse=True)[:10]
```

```
[8]: location = None
for contour in contours:
    approx = cv2.approxPolyDP(contour, 10, True)
    if len(approx) == 4:
        location = approx
        break
```

```
[9]: location
```

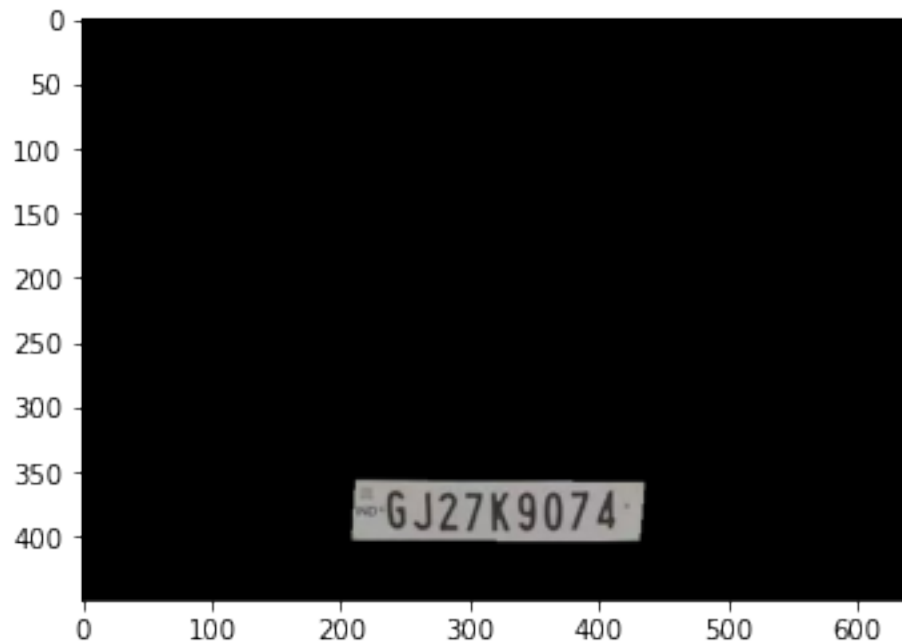
```
[9]: array([[435, 358]],
          [[212, 356]],
          [[209, 402]],
          [[431, 403]]], dtype=int32)
```

```
[10]: mask = np.zeros(gray.shape, np.uint8)
new_image = cv2.drawContours(mask, [location], 0, 255, -1)
```

```
new_image = cv2.bitwise_and(img, img, mask=mask)
```

```
[11]: plt.imshow(cv2.cvtColor(new_image, cv2.COLOR_BGR2RGB))
```

```
[11]: <matplotlib.image.AxesImage at 0x1fad7ccda60>
```



```
[12]: (x,y) = np.where(mask==255)
(x1, y1) = (np.min(x), np.min(y))
(x2, y2) = (np.max(x), np.max(y))
cropped_image = gray[x1:x2+1, y1:y2+1]
```

```
[13]: plt.imshow(cv2.cvtColor(cropped_image, cv2.COLOR_BGR2RGB))
```

```
[13]: <matplotlib.image.AxesImage at 0x1fad8d135b0>
```



```
[14]: # Use Easy OCR To Read Text
reader = easyocr.Reader(['en'])
result = reader.readtext(cropped_image)
result
```

CUDA not available - defaulting to CPU. Note: This module is much faster with a GPU.

```
[14]: [[([19, 3], [210, 3], [210, 46], [19, 46]), '6J27k9074', 0.7518029508547546]]
```

```
[15]: # Render Result
text = result[0][-2]
font = cv2.FONT_HERSHEY_SIMPLEX
res = cv2.putText(img, text=text, org=(approx[0][0][0], approx[1][0][1]+60),
    ↳fontFace=font, fontScale=1, color=(0,255,0), thickness=2, lineType=cv2.
    ↳LINE_AA)
res = cv2.rectangle(img, tuple(approx[0][0]), tuple(approx[2][0]), (0,255,0),3)
plt.imshow(cv2.cvtColor(res, cv2.COLOR_BGR2RGB))
```

```
[15]: <matplotlib.image.AxesImage at 0x1fadb5f46d0>
```



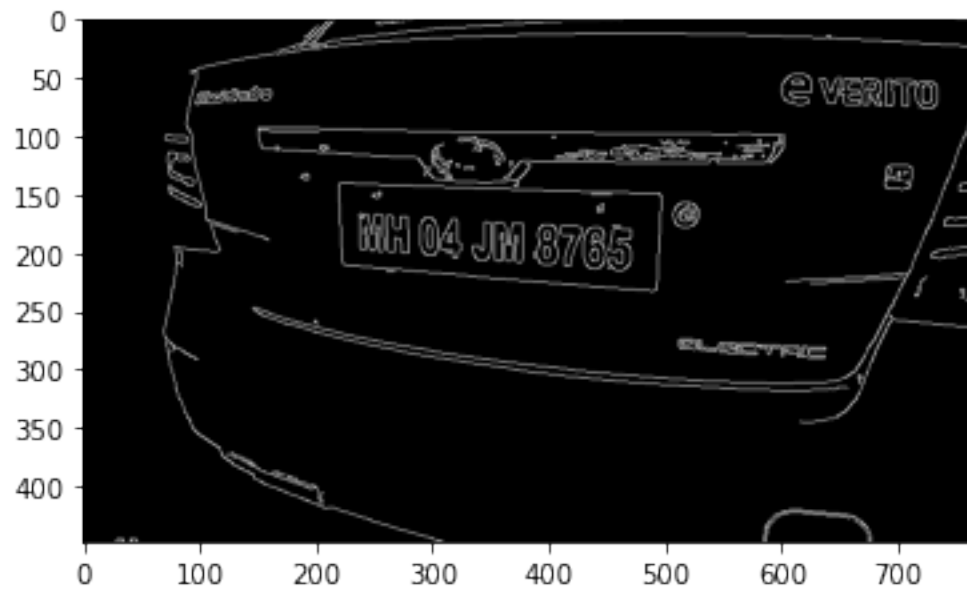
```
[16]: # Read in image, Grayscale and Blur
img = cv2.imread('wheeler3.jpg')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
plt.imshow(cv2.cvtColor(gray, cv2.COLOR_BGR2RGB))
```


[16]: <matplotlib.image.AxesImage at 0x1fade08d7f0>



```
[17]: # Apply filter and find edges for localization
bfilter = cv2.bilateralFilter(gray, 11, 17, 17) #Noise reduction
edged = cv2.Canny(bfilter, 30, 200) #Edge detection
plt.imshow(cv2.cvtColor(edged, cv2.COLOR_BGR2RGB))
```

[17]: <matplotlib.image.AxesImage at 0x1fade0fad30>



```
[18]: # Find Contours And Apply Mask
keypoints = cv2.findContours(edged.copy(), cv2.RETR_TREE, cv2.
    ↳CHAIN_APPROX_SIMPLE)
contours = imutils.grab_contours(keypoints)
contours = sorted(contours, key=cv2.contourArea, reverse=True)[:10]
```

```
[19]: location = None
for contour in contours:
    approx = cv2.approxPolyDP(contour, 10, True)
    if len(approx) == 4:
        location = approx
        break
```

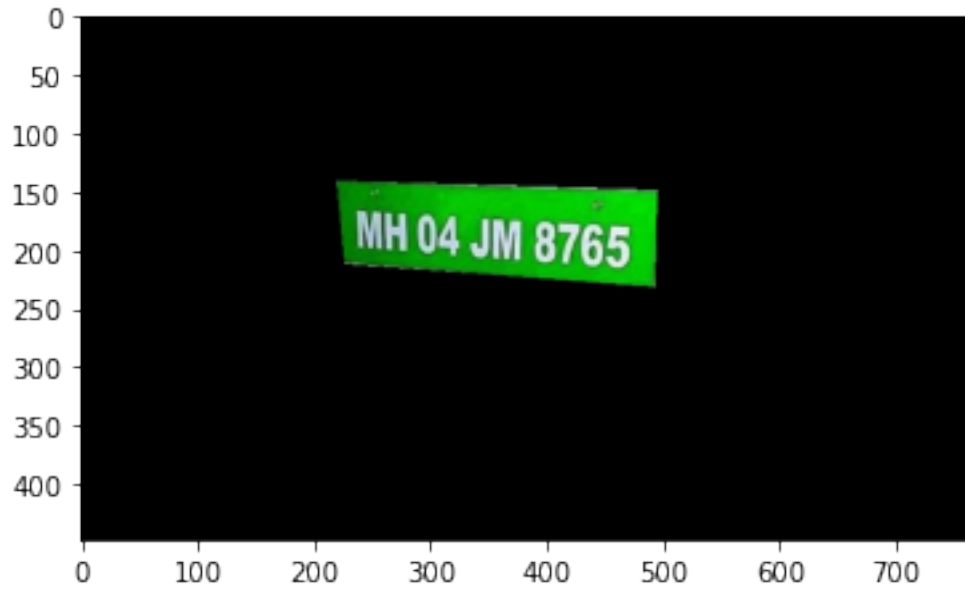
```
[20]: location
```

```
[20]: array([[220, 141]],
          [[227, 211]],
          [[493, 231]],
          [[495, 149]]], dtype=int32)
```

```
[21]: mask = np.zeros(gray.shape, np.uint8)
new_image = cv2.drawContours(mask, [location], 0, 255, -1)
new_image = cv2.bitwise_and(img, img, mask=mask)
```

```
[22]: plt.imshow(cv2.cvtColor(new_image, cv2.COLOR_BGR2RGB))
```

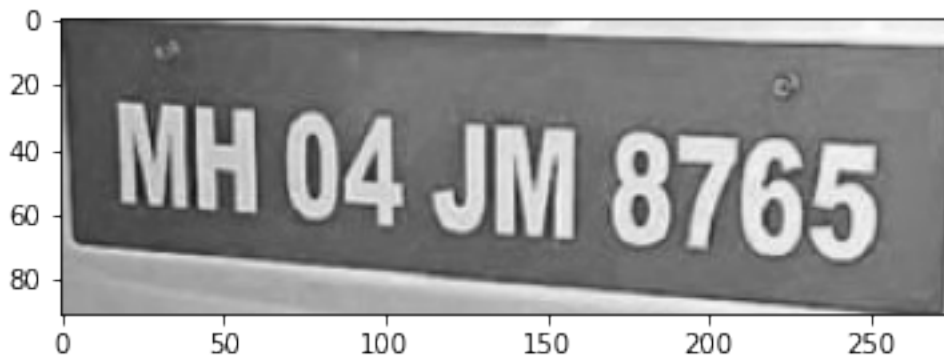
```
[22]: <matplotlib.image.AxesImage at 0x1fade169fa0>
```



```
[23]: (x,y) = np.where(mask==255)
      (x1, y1) = (np.min(x), np.min(y))
      (x2, y2) = (np.max(x), np.max(y))
      cropped_image = gray[x1:x2+1, y1:y2+1]
```

```
[24]: plt.imshow(cv2.cvtColor(cropped_image, cv2.COLOR_BGR2RGB))
```

```
[24]: <matplotlib.image.AxesImage at 0x1fade1d1af0>
```



```
[25]: # Use Easy OCR To Read Text
      reader = easyocr.Reader(['en'])
      result = reader.readtext(cropped_image)
      result
```

CUDA not available - defaulting to CPU. Note: This module is much faster with a GPU.

```
[25]: [([9, 17], [264, 17], [264, 86], [9, 86]),  
      'MH04 JM 8765',  
      0.8127615749455771)]
```

```
[26]: # Render Result  
text = result[0][-2]  
font = cv2.FONT_HERSHEY_SIMPLEX  
res = cv2.putText(img, text=text, org=(approx[0][0][0], approx[1][0][1]+60),  
                  ↪fontFace=font, fontScale=1, color=(0,255,0), thickness=2, lineType=cv2.  
                  ↪LINE_AA)  
res = cv2.rectangle(img, tuple(approx[0][0]), tuple(approx[2][0]), (0,255,0),3)  
plt.imshow(cv2.cvtColor(res, cv2.COLOR_BGR2RGB))
```

```
[26]: <matplotlib.image.AxesImage at 0x1fae29a3130>
```



```
[ ]:
```