STEP 0: INSTALLING AND IMPORTING THE DEPENDENCIES

- Vishal BImal Francis
- 21122069

In [2]:
```
# !pip install easyocr
# !pip install imutils
```

In [12]:
```
#conda install pytorch torchvision torchaudio pytorch-cuda=11.7 -c pytorch -c nvidia
```

In [1]:
```
import cv2
import matplotlib
import numpy
import imutils
import easyocr
```

STEP 1: LOADING THE IMAGE

In [252…
```
#READ IN IMAGE IN GRAYSCALE
filename="C:\\Users\\visha\\OneDrive\\Desktop\\images number plate\\image02.jpg"
# filename="C:\\Users\\visha\\OneDrive\\Desktop\\numberplate.png"
img=cv2.imread(filename)
img
#we read the file in pixels(numbers in a matrix)
```

Out[252]:
```
array([[[255, 255, 255],
        [255, 255, 255],
        [255, 255, 255],
        ...,
        [255, 255, 255],
        [255, 255, 255],
        [255, 255, 255]],

       [[255, 255, 255],
        [255, 255, 255],
        [255, 255, 255],
        ...,
        [255, 255, 255],
        [255, 255, 255],
        [255, 255, 255]],

       [[255, 255, 255],
        [255, 255, 255],
        [255, 255, 255],
        ...,
        [255, 255, 255],
        [255, 255, 255],
        [255, 255, 255]],

       ...,

       [[255, 255, 255],
        [255, 255, 255],
        [255, 255, 255],
        ...,
        [255, 255, 255],
        [255, 255, 255],
        [255, 255, 255]],

       [[255, 255, 255],
        [255, 255, 255],
        [255, 255, 255],
        ...,
        [255, 255, 255],
        [255, 255, 255],
        [255, 255, 255]],

       [[255, 255, 255],
        [255, 255, 255],
        [255, 255, 255],
        ...,
        [255, 255, 255],
        [255, 255, 255],
        [255, 255, 255]]], dtype=uint8)
```

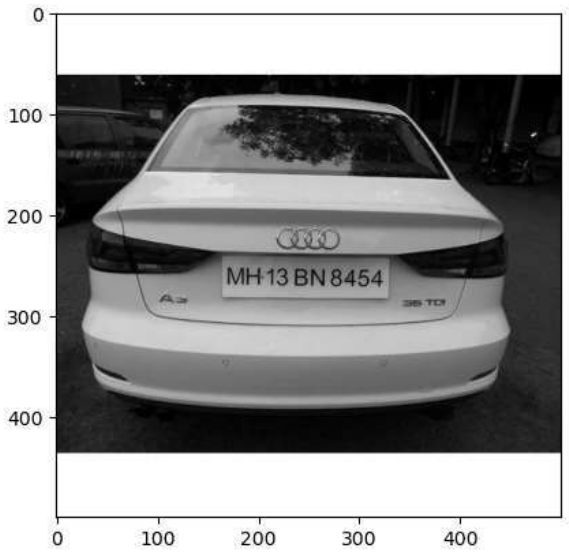
STEP 2: APPLY FILTERS

In [253…
```
#WE APPLY GRAY FILTER TO THE IMAGE
gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
gray
```

Out[253]:
```
array([[255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       ...,
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255]], dtype=uint8)
```

In [254…
```
# TO SEE THE IMAGE WE NEED TO CONVERT THE GRAYSCLE TO RGB
plt.imshow(cv2.cvtColor(gray,cv2.COLOR_BGR2RGB))
```

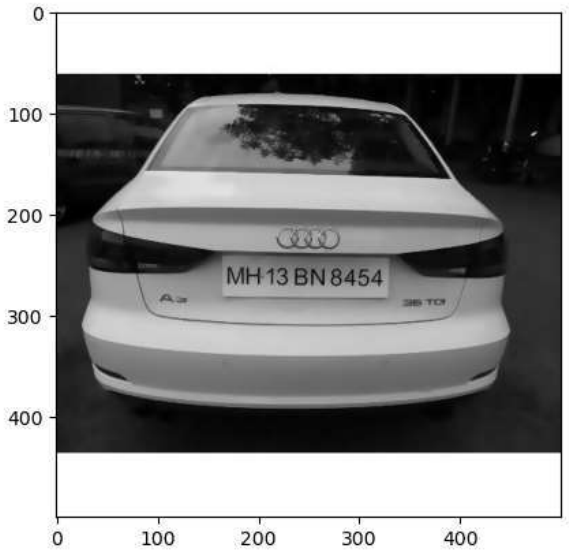
Out[254]:
```
<matplotlib.image.AxesImage at 0x2c2e7f58c40>
```



In [255…
```
# import torch
# import torchvision
# torch.cuda.is_available()
```

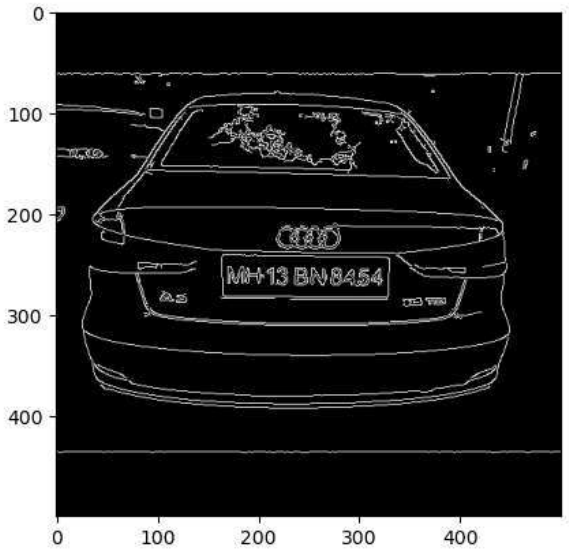
In [256…
```
# APPLY FILTER AND FIND EDGE FOR LOCALIZATION
# bilateral filter is used for smoothening images and reducing noise,
bfilter=cv2.bilateralFilter(gray,11,17,17)
plt.imshow(cv2.cvtColor(bfilter,cv2.COLOR_BGR2RGB))
```

Out[256]:
```
<matplotlib.image.AxesImage at 0x2c281765b20>
```



In [257…
```
#edge detection
edged=cv2.Canny(bfilter,30,200)
plt.imshow(cv2.cvtColor(edged,cv2.COLOR_BGR2RGB))
```

Out[257]:
```
<matplotlib.image.AxesImage at 0x2c296da06a0>
```



STEP 3: FIND CONTOURS AND APPLY MASK

In [258…
```
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]
```

In [259…
```
#contours in the image
location=None
for contour in contours:
    approx=cv2.approxPolyDP(contour,10,True)
    #approx any polygon from the contour
    if len(approx)==4:
        location=approx
        break
```

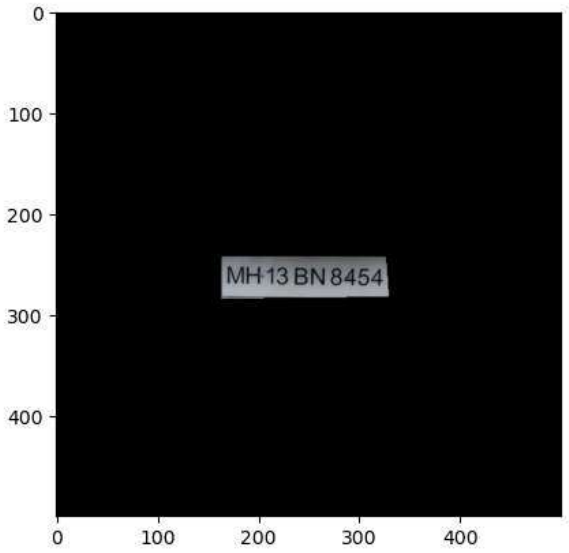
In [260…
```
location
```

Out[260]:
```
array([[[164, 243]],

       [[164, 284]],

       [[329, 282]],

       [[326, 243]]], dtype=int32)
```

In [261…
```
#masking using numpy
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)
```

In [262…
```
plt.imshow(cv2.cvtColor(new_image,cv2.COLOR_BGR2RGB))
```

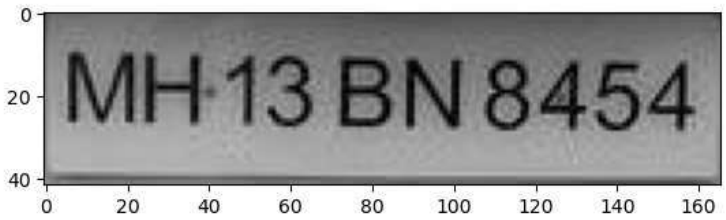
Out[262]:
```
<matplotlib.image.AxesImage at 0x2c2a318fc10>
```



In [263…
```
(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]
plt.imshow(cv2.cvtColor(cropped_image,cv2.COLOR_BGR2RGB))
```

Out[263]: <matplotlib.image.AxesImage at 0x2c2b85ad160>



STEP 4: USE EASY OCR TO READ TEXT

In [264…
```
reader=easyocr.Reader(['en'])
result=reader.readtext(cropped_image)
result
```

Out[264]: [([[2, 6], [162, 6], [162, 34], [2, 34]], 'MH13 BN8454', 0.9481776243811364)]

STEP 5: RENDER RESULT

In [265…
```
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), thicknes
res = cv2.rectangle(img, tuple(approx[0][0]), tuple(approx[2][0]), (0,255,0),3)
plt.imshow(cv2.cvtColor(res, cv2.COLOR_BGR2RGB))
```

Out[265]: <matplotlib.image.AxesImage at 0x2c2dab57820>

In [ ]: