**Assignment-3**

**Build CNN Model for Classification Of Flowers**

| Assignment Date | 12 September 2022 |
| --- | --- |
| Student Name | Jeyanthi C |
| Team ID | PNT2022TMID04987 |
| Maximum Marks | 2 Marks |

# 

# **Importing required lib**

# [ ] import cv2

# import numpy as np

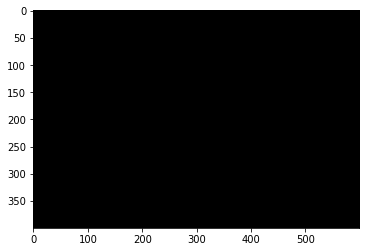
import matplotlib.pyplot as plt

# **Create black image**

[ ] img1 = np.zeros((400,600,3),np.uint8)

plt.imshow(img1)

<matplotlib.image.AxesImage at 0x7fc494447490>



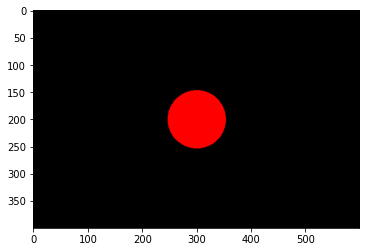
**Drawing functions**

[ ] # Drawing circle

circle = cv2.circle(img1,(300,200),50,(255,0,0),-1) #(0,0,0)-->(R,G,B)

plt.imshow(img1)

<matplotlib.image.AxesImage at 0x7fc494252ad0>

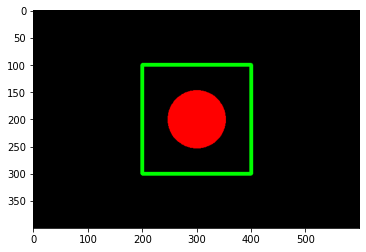


[ ] # Drawing rectangle

rectangle = cv2.rectangle(img1,(200,100),(400,300),(0,255,0),5)

plt.imshow(img1)

<matplotlib.image.AxesImage at 0x7fc4941d9fd0>



[ ] # Drawing lines

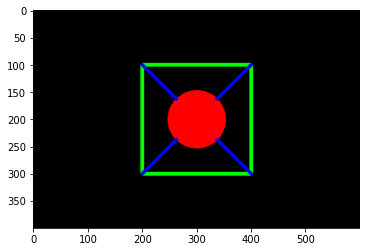
line1 = cv2.line(img1,(200,100),(400,300),(0,0,255),5)

line2 = cv2.line(img1,(400,100),(200,300),(0,0,255),5)

circle = cv2.circle(img1,(300,200),50,(255,0,0),-1)

plt.imshow(img1)

<matplotlib.image.AxesImage at 0x7fc494055210>

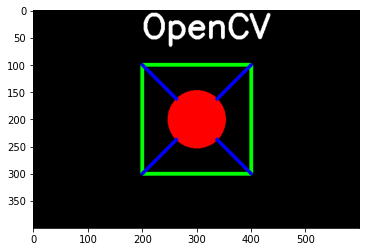


[ ] # Writing text on image

text = cv2.putText(img1,'OpenCV',(200,50),cv2.FONT\_HERSHEY\_SIMPLEX,2,(255,255,255),5)

plt.imshow(img1)

<matplotlib.image.AxesImage at 0x7fc4940b4550>



[ ] # Save the image

cv2.imwrite('output.jpg',img1)

True

[ ] # Reading the image

img = cv2.imread('/content/cat.jpg')

plt.imshow(img)

<matplotlib.image.AxesImage at 0x7fc4935bfe10>



[ ] # Converting color (BGR to RGB)

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

plt.imshow(img\_rgb)

<matplotlib.image.AxesImage at 0x7fc4935431d0>

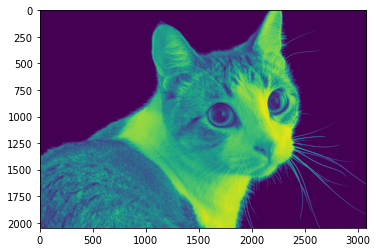


[ ] # Converting color (BGR to Gray)

img\_gray = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

plt.imshow(img\_gray)

<matplotlib.image.AxesImage at 0x7fc492cd9650>



[ ] # Visualizing the shape

img\_rgb.shape

(2048, 3072, 3)

[ ] # Reshape image

img\_resize = cv2.resize(img\_rgb,(500,400))

plt.imshow(img\_resize)

<matplotlib.image.AxesImage at 0x7fc492cb5050>



[ ] # Rotate image

img\_rotate = cv2.rotate(img\_resize,cv2.ROTATE\_90\_CLOCKWISE)

plt.imshow(img\_rotate)

<matplotlib.image.AxesImage at 0x7fc492c0c650>

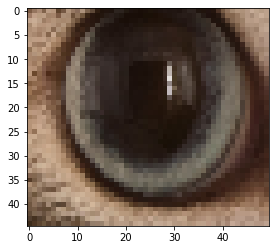


[ ] # Image crop

img\_crop = img\_resize[175:220,250:300]

plt.imshow(img\_crop)

<matplotlib.image.AxesImage at 0x7fc48a417390>



[ ] # Image Blur

img\_blur = cv2.GaussianBlur(img\_resize,(11,11),cv2.BORDER\_ISOLATED)

plt.imshow(img\_blur)

<matplotlib.image.AxesImage at 0x7fc48a21c490>

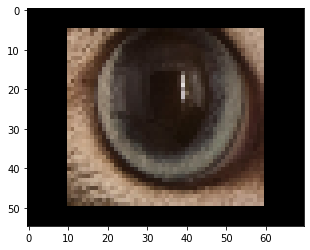


[ ] # Creating Border

im=cv2.copyMakeBorder(img\_crop,5,5,10,10,cv2.BORDER\_CONSTANT)

plt.imshow(im)

<matplotlib.image.AxesImage at 0x7fc48a1fb4d0>

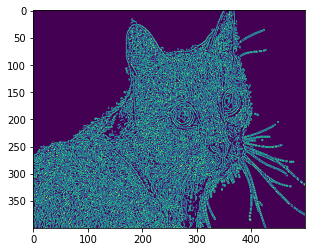


[ ] # Edge detection

img\_canny = cv2.Canny(img\_resize,20,20)

plt.imshow(img\_canny)

<matplotlib.image.AxesImage at 0x7fc489fb9d90>



[ ] # Edge detection

img2 = cv2.imread('/content/Hyundai\_KONA-EV\_Side-Profile\_640x331.png')

img2\_canny = cv2.Canny(img2,150,150)

plt.imshow(img2\_canny)

<matplotlib.image.AxesImage at 0x7fc48a0e3050>



[ ] plt.imshow(img2)

<matplotlib.image.AxesImage at 0x7fc48a4dba10>



[ ]