**COMPUTER VISION  
ASSIGNMENT-1**

**AMOGH GARG – 2020UCO1688**

Importing Libraries

import cv2  
import numpy as np  
import matplotlib.pyplot as plt

Task-1 : Read an image into a variable

Task-2 : Display that image

img = cv2.imread("Img-1.jpg",cv2.IMREAD\_COLOR)  
img = cv2.resize(img, (700,700))  
cv2.imshow('Colored Image of Butterfly', img)  
cv2.waitKey(0)  
cv2.destroyAllWindows()

Task-3 : Convert the image into grayscale

img\_gs = cv2.imread("Img-1.jpg",cv2.IMREAD\_GRAYSCALE)  
img\_gs = cv2.resize(img\_gs, (700,700))  
cv2.imshow('Grayscale Image of Butterfly', img\_gs)  
cv2.waitKey(0)  
cv2.destroyAllWindows()

Task-4 : Check the height and width of that image

img\_gs.shape

(700, 700)

Task-5 : Extract RGB channels from the image

# extract red channel  
red\_channel = img[:,:,2]  
  
# create empty image with same shape as that of src image  
red\_img = np.zeros(img.shape)  
  
#assign the red channel of src to empty image  
red\_img[:,:,2] = red\_channel  
  
#save image  
cv2.imwrite('Red\_Img-1.jpg',red\_img)

True

# extract green channel  
green\_channel = img[:,:,1]  
  
# create empty image with same shape as that of src image  
green\_img = np.zeros(img.shape)  
  
#assign the red channel of src to empty image  
green\_img[:,:,1] = green\_channel  
  
#save image  
cv2.imwrite('Green\_Img-1.jpg',green\_img)

True

# extract blue channel  
blue\_channel = img[:,:,0]  
  
# create empty image with same shape as that of src image  
blue\_img = np.zeros(img.shape)  
  
#assign the red channel of src to empty image  
blue\_img[:,:,0] = blue\_channel  
  
#save image  
cv2.imwrite('Blue\_Img-1.jpg',blue\_img)

True

Task-6 : Extract middle 100 pixels from the image

# Extracting the middle hundred pixels  
cropped\_img = img[300:400,300:400]  
cv2.imshow('Center Image', cropped\_img)  
cv2.waitKey(0)  
cv2.destroyAllWindows()

Task-7 :For the image, create one more image. Keep every 10th pixel in the horizontal direction and 20th pixel in vertical direction. Display the output matrix in image form.

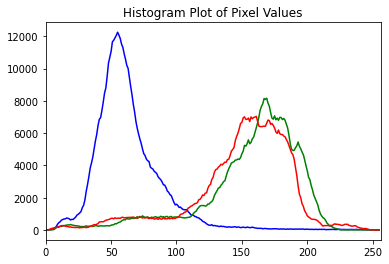
# Extracting every 10th pixel in horizontal and every 20th pixel in vertical direction  
modified\_img = img[::10,::20]  
cv2.imshow('Modified Image', modified\_img)  
cv2.waitKey(0)  
cv2.destroyAllWindows()

Task-8 :Flip the image vertically

flipped\_img = cv2.flip(img,0)  
cv2.imshow('Flipped Image', flipped\_img)  
cv2.waitKey(0)  
cv2.destroyAllWindows()

Task-9 :Draw the histogram plot of pixel values.

color = ('b','g','r')  
for i,col in enumerate(color):  
 histr = cv2.calcHist([img],[i],None,[256],[0,256])  
 plt.plot(histr,color = col)  
 plt.xlim([0,256])  
plt.title('Histogram Plot of Pixel Values')  
plt.show()



**OUTPUT IMAGES:**

****