Lab 11

Image Segmentation

Ahmad Amjad Mughal

121672

BSCS6C

**Code:**

import cv2

import numpy as np

#Read an image in RGB

img = cv2.imread("image.png", 1)

#Make an gauusian mask and apply binarization

blurred\_image = cv2.GaussianBlur(img,(7,7),0)

retval ,binarized\_image = cv2.threshold(blurred\_image,40,255,cv2.THRESH\_BINARY)

#Apply a filter morphological Closing operation

filter = np.ones((3,3),np.uint8)

closed\_image = cv2.morphologyEx(binarized\_image, cv2.MORPH\_CLOSE, filter)

cv2.imshow("Binarized", closed\_image);

#Apply connected components built-in function

retval, markers = cv2.connectedComponents(closed\_image)

#Give Color some label that locates object

label\_hue = np.uint8(110\*markers/np.max(markers))

#Black Area

blank\_ch = 255\*np.ones\_like(label\_hue)

#Merge an image using components and black area

labeled\_image = cv2.merge([label\_hue, blank\_ch, blank\_ch])

#Convert Binarized image into coloured image

labeled\_image = cv2.cvtColor(labeled\_image, cv2.COLOR\_HSV2RGB)

#Assign label 0 to colour 0

labeled\_image[label\_hue==0] = 0

unique\_colors = np.unique(labeled\_image.reshape(-1, labeled\_image.shape[2]), axis=0)

#Give a list of all colours assign to different objects

print "Colors available in labeled image:"

for x in xrange(unique\_colors.shape[0]):

print str(x+1)+"=> B:"+str(unique\_colors[x,0])+" G:"+str(unique\_colors[x,1])+" R:"+str(unique\_colors[x,2])+" "

output\_image = np.zeros\_like(labeled\_image)

#Assign that colour to specific object

for x in xrange(labeled\_image.shape[0]):

for y in xrange(labeled\_image.shape[1]):

if (labeled\_image[x,y,0] == int(0) and labeled\_image[x,y,1] == int(84) and labeled\_image[x,y,2] == int(255)):

output\_image[x,y,0:3] = labeled\_image[x,y,0:3]

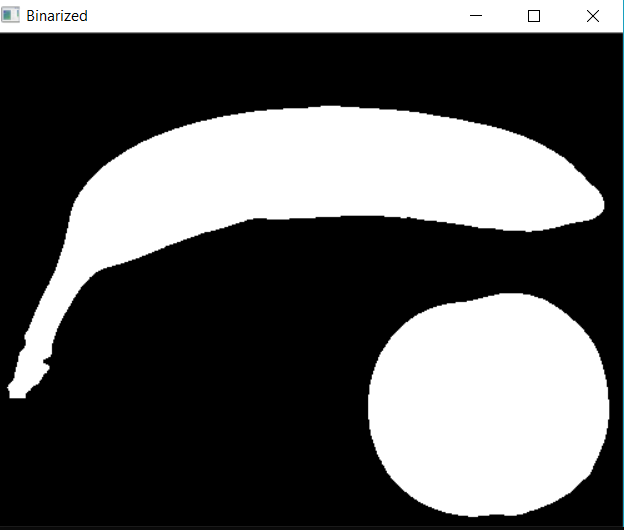
#show the output

cv2.imshow("Selected", labeled\_image)

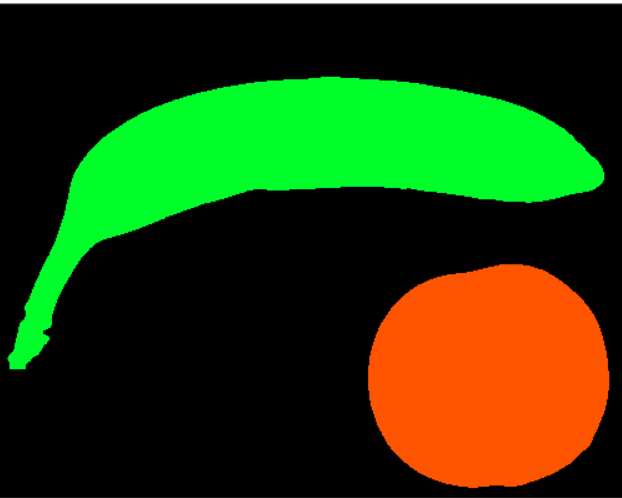
cv2.waitKey(0)

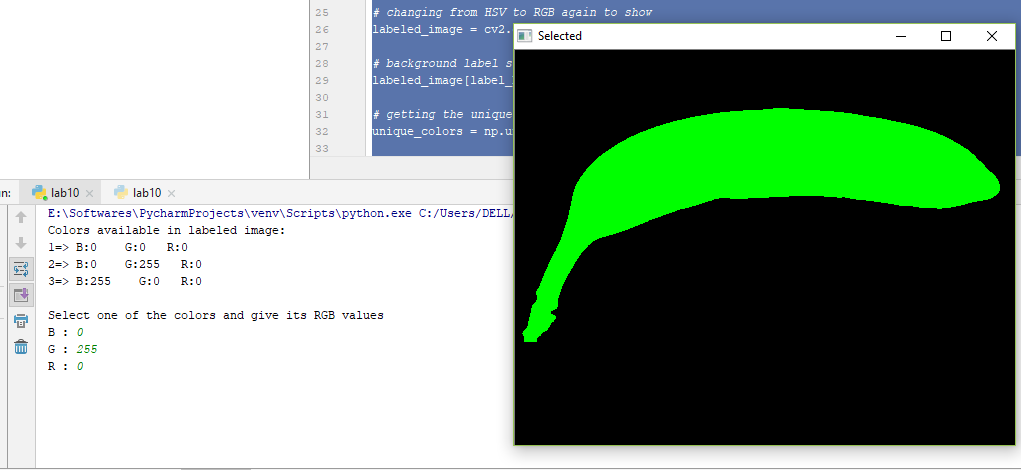
**Outputs**

**Binarized Image**

****

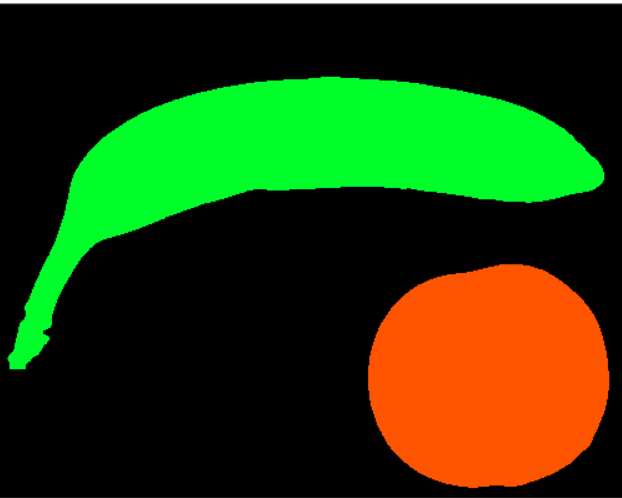
**Resultant Image**

****



**Selected Image RGB (0,255,0) Green**

**1 Object**

****

**Selected Image RGB (42,255,0) Orange**

**1 Object**

**Conclusion**

We performed image segmentation and this can be done via using prior knowledge of morphological operations, binarization and connected components. This is not a trivial lab as it demands separation of each object.