

Walchand College of Engineering, Sangli

Computer Science & Engineering

Third Year

Course: Image Processing

PRN NO : 2020BTECS00207

Name : Kshirsagar Mayuri Manojkumar

Lab course coordinator : Ms. P.D.Mundada

Batch: - T5

Experiment No:8

Q 1) Perform segmentation by global thresholding over 4 different images. (For selection of threshold plot and display histogram of image also, and from the result choose the threshold).

Function used:

Cv2.threshold()

Cv2.calcHist()

Syntax & meaning:

Cv2.threshold(src, thresholdvalue, maxval, thresholdtechnique)

It uses parameters as follows->

Src – It contains the input image.

Thresholdvalue – Value of threshold below and above which pixel values change accordingly.

Maxval – Maximum value assigned to the pixel.

Thresholdingtechnique – Type of thresholding to be applied

`cv2.calcHist()` it is in-built function in OpenCV to find the histogram.

`cv2.calcHist(images, channels, mask, histSize, ranges[, hist[, accumulate]])`

It uses parameters as follows->

images : it is the source image of type uint8 or float32.

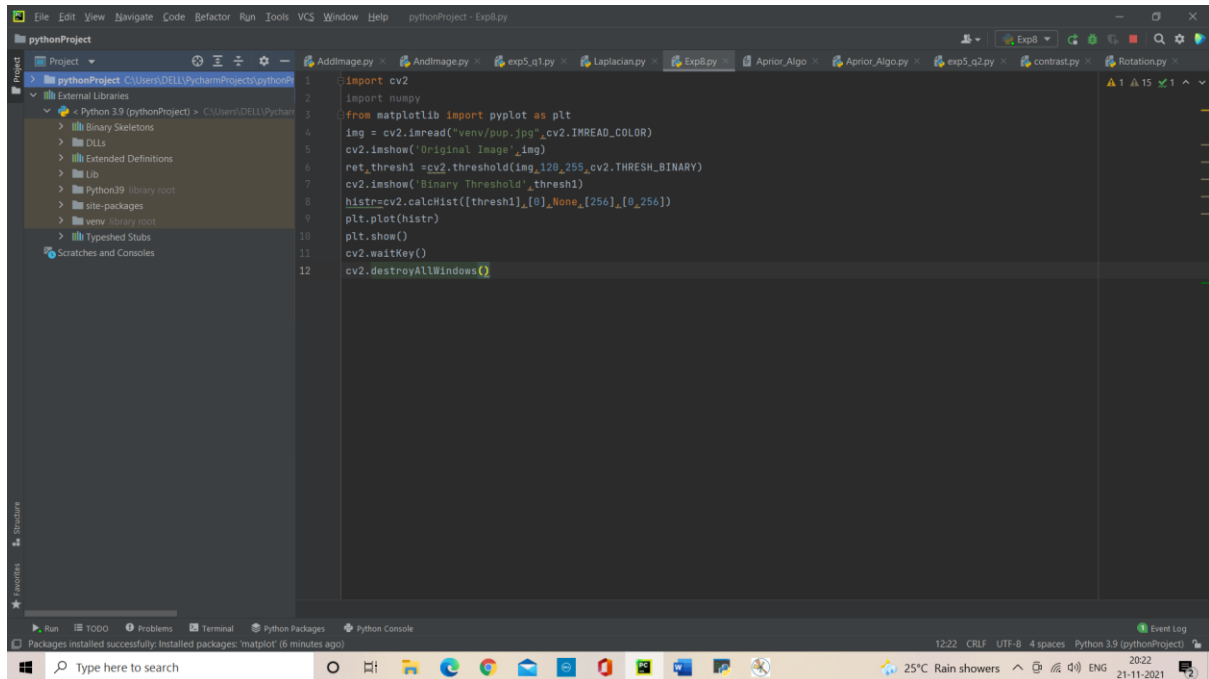
channels : it is the index of channel for which we calculate histogram. For grayscale image ,its value is [0] and color image,you can pass [0],[1] or [2]to calculate histogram of blue ,green or red channel respectively.

mask : mask image. To find histogram of full image, it is given as “None”.

histSize : this represents our BIN count. For full scale, we pass [256].

ranges : this is our RANGE. Normally, it is [0,256].

CODE:

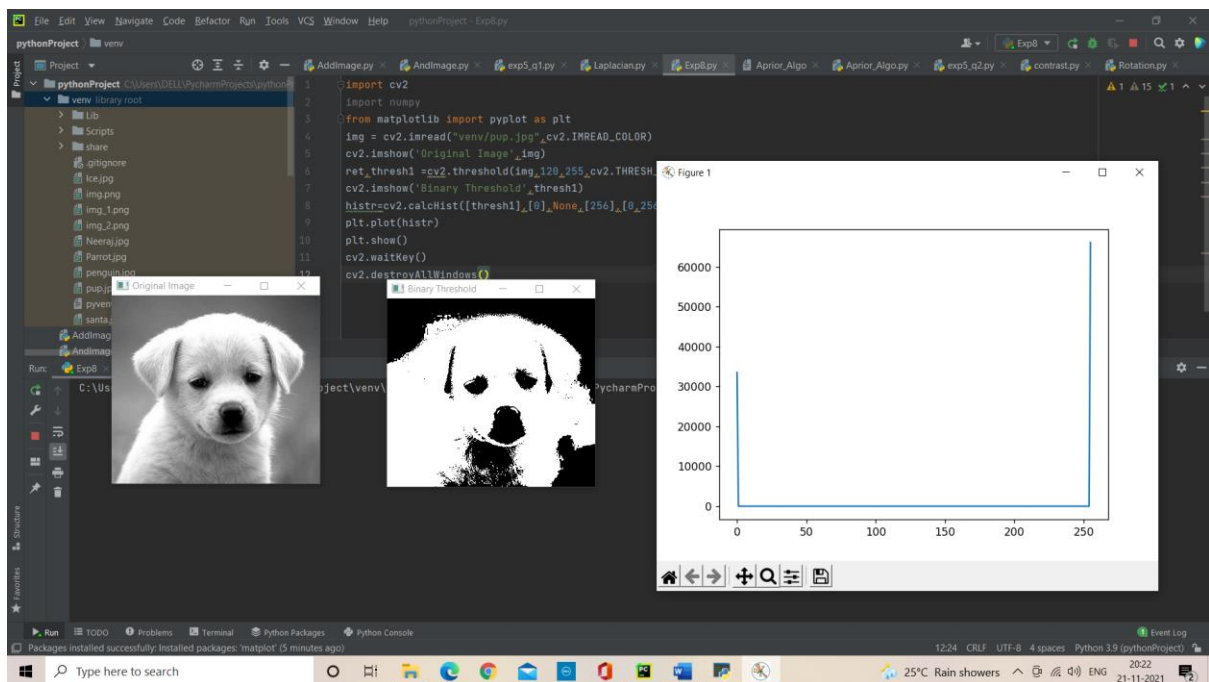


```

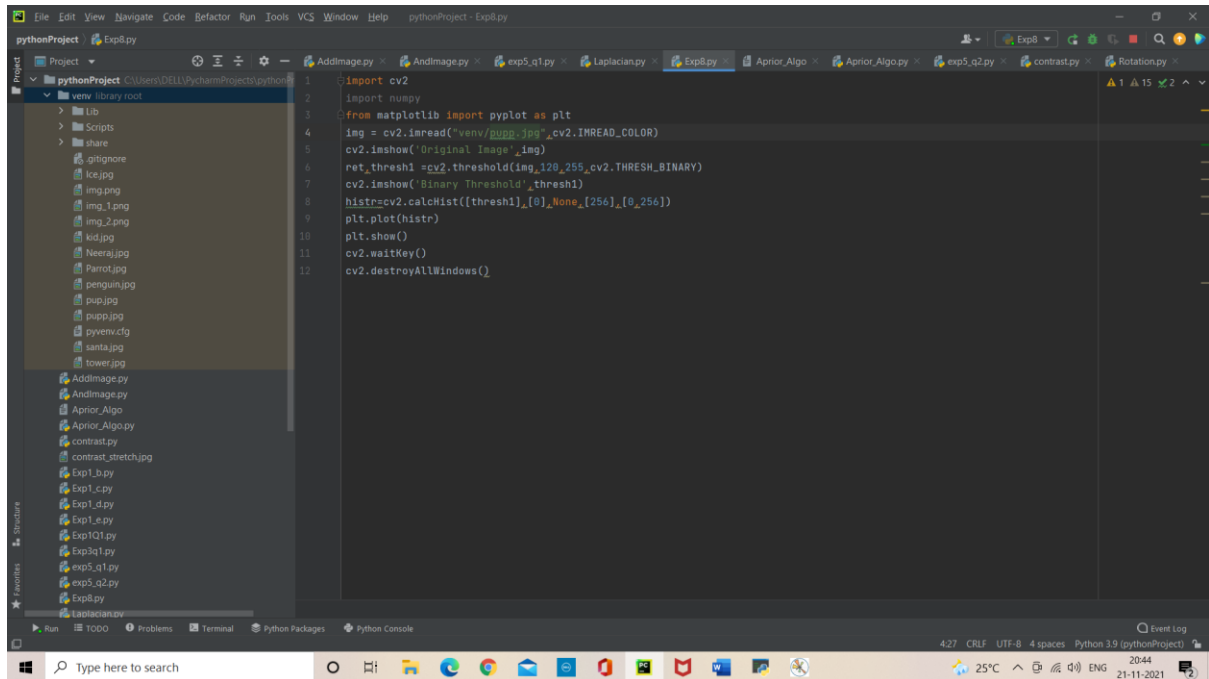
1  import cv2
2  import numpy
3  from matplotlib import pyplot as plt
4  img = cv2.imread("venv/pup.jpg",cv2.IMREAD_COLOR)
5  cv2.imshow('Original Image',img)
6  ret,thresh1 =cv2.threshold(img,120,255,cv2.THRESH_BINARY)
7  cv2.imshow('Binary Threshold',thresh1)
8  histr=cv2.calcHist([thresh1],[0],None,[256],[0,256])
9  plt.plot(histr)
10 plt.show()
11 cv2.waitKey()
12 cv2.destroyAllWindows()
  
```

The screenshot shows the PyCharm IDE with a project named 'pythonProject'. The code in 'Exp8.py' performs the following steps: imports cv2 and numpy, imports pyplot from matplotlib, reads an image 'pup.jpg' in color, displays it as 'Original Image', applies a binary threshold (120, 255) to create 'Binary Threshold', calculates the histogram for the thresholded image, plots the histogram, displays the plot, waits for a key press, and finally destroys all windows.

OUTPUT:



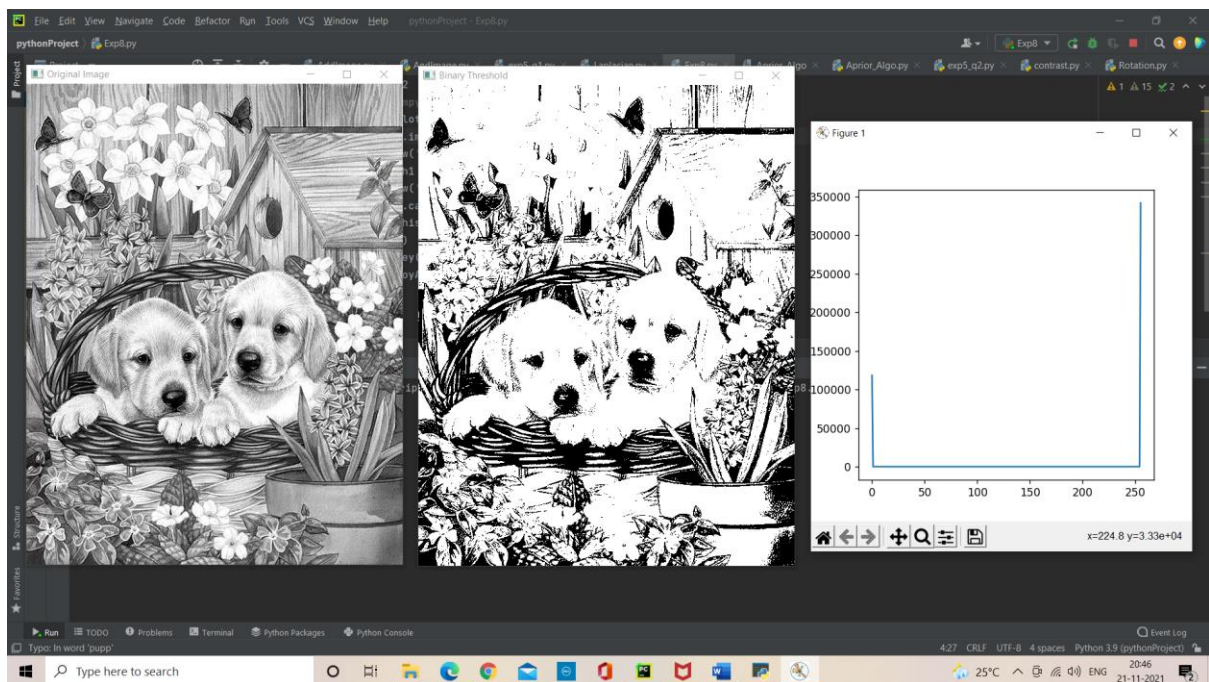
CODE:



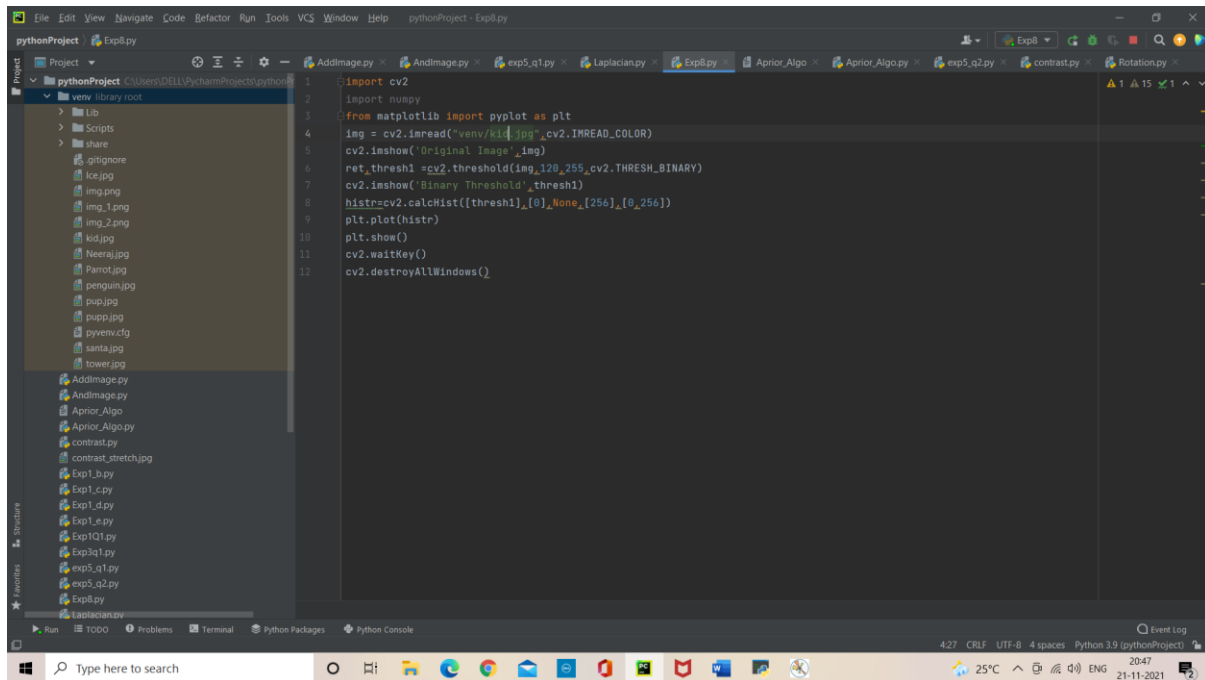
```

1 import cv2
2 import numpy
3 from matplotlib import pyplot as plt
4 img = cv2.imread("venv/gugg.jpg", cv2.IMREAD_COLOR)
5 cv2.imshow('Original Image', img)
6 ret, thresh1 = cv2.threshold(img, 120, 255, cv2.THRESH_BINARY)
7 cv2.imshow('Binary Threshold', thresh1)
8 hist=cv2.calcHist([thresh1],[0],None,[256],[0,256])
9 plt.plot(hist)
10 plt.show()
11 cv2.waitKey()
12 cv2.destroyAllWindows()
  
```

OUTPUT:



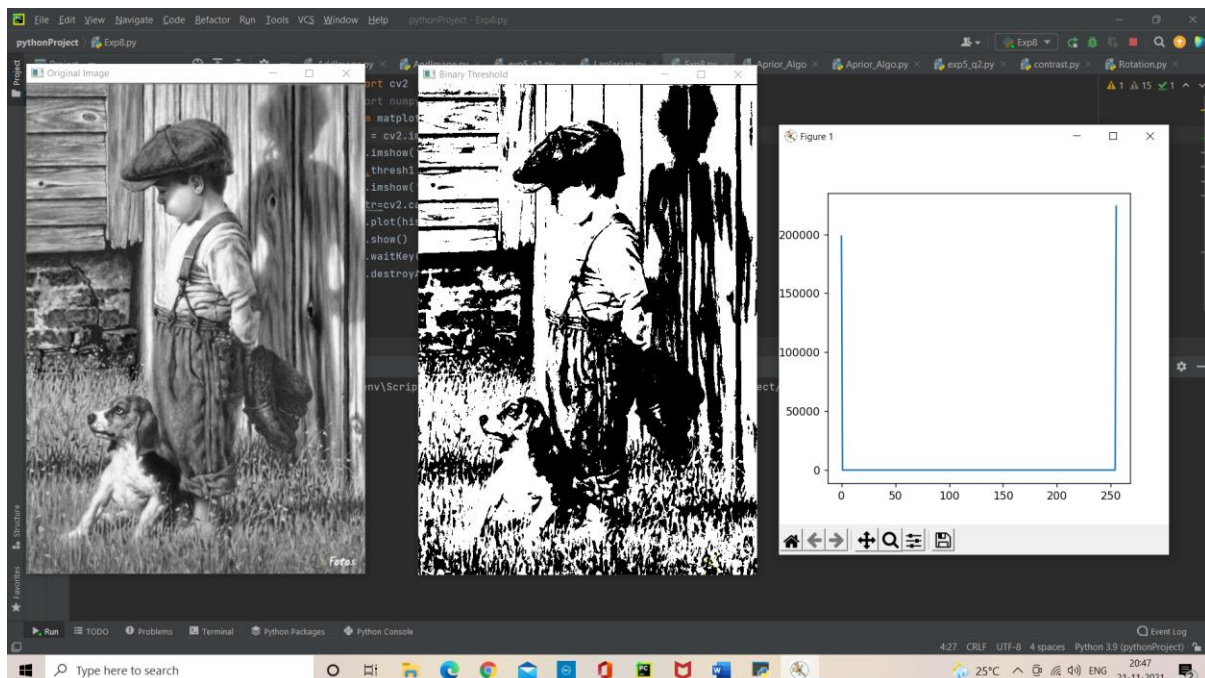
CODE:



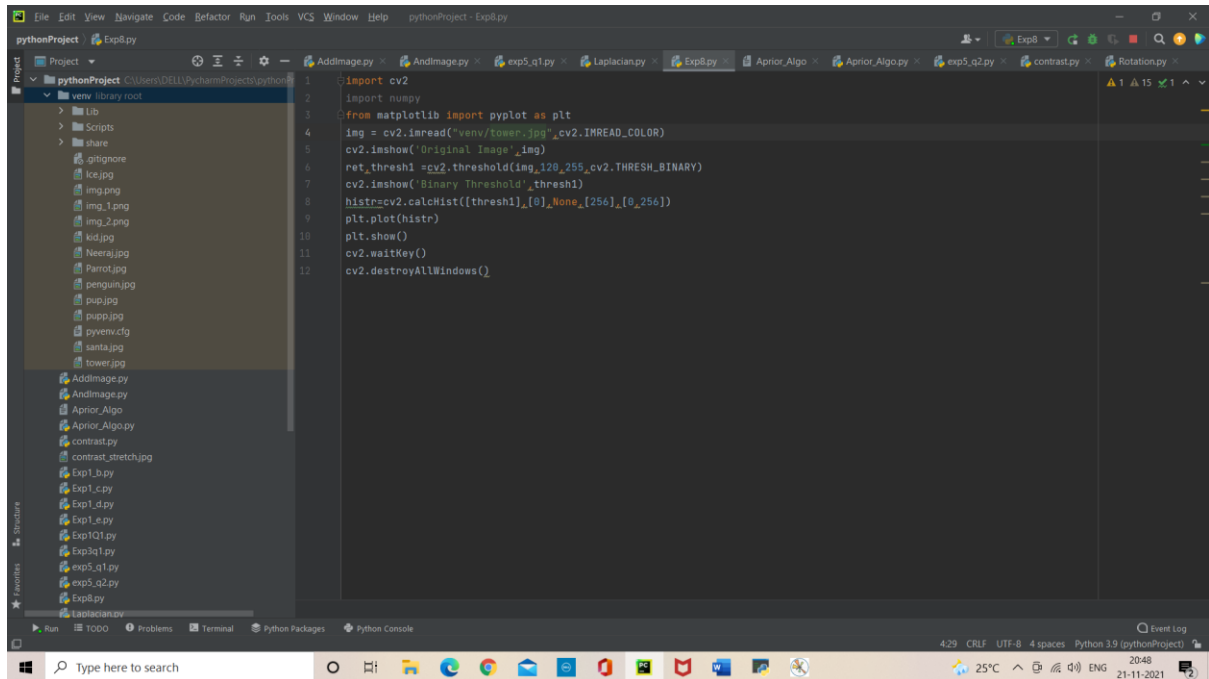
```

1 import cv2
2 import numpy
3 from matplotlib import pyplot as plt
4 img = cv2.imread("venv\kid.jpg", cv2.IMREAD_COLOR)
5 cv2.imshow('Original Image', img)
6 ret, thresh1 = cv2.threshold(img, 128, 255, cv2.THRESH_BINARY)
7 cv2.imshow('Binary Threshold', thresh1)
8 histr = cv2.calcHist([thresh1], [0], None, [256], [0, 256])
9 plt.plot(histr)
10 plt.show()
11 cv2.waitKey()
12 cv2.destroyAllWindows()
  
```

OUTPUT:



CODE:



```

1  import cv2
2  import numpy
3  from matplotlib import pyplot as plt
4  img = cv2.imread("venv/tower.jpg", cv2.IMREAD_COLOR)
5  cv2.imshow('Original Image', img)
6  ret, thresh1 = cv2.threshold(img, 120, 255, cv2.THRESH_BINARY)
7  cv2.imshow('Binary Threshold', thresh1)
8  hist=cv2.calcHist([thresh1],[0],None,[256],[0,256])
9  plt.plot(hist)
10 plt.show()
11 cv2.waitKey()
12 cv2.destroyAllWindows()
  
```

OUTPUT:

