Name: Jay Goyal

Roll no.: C017

Semester: VI

Program: B.Tech

Branch: EXTC

Date of performance: 18th December

Date of Submission: 25th December

Experiment Number: 2

Aim =

a. To write a program in PYTHON to obtain image negative of an image

b. To write a program in PYTHON to obtain thresholding of an image

c. To write a program in PYTHON to obtain grey level slicing of an image without background

d. To write a program in PYTHON to obtain grey level slicing of an image with backgroun

Conclusion: Outcome:

From this experiment we learnt the concept of image enhancement in spatial domain using point processing methods and the topics covered where Point processing in spatial domain Image Negative Thresholding Grey level slicing without background Grey level slicing with background

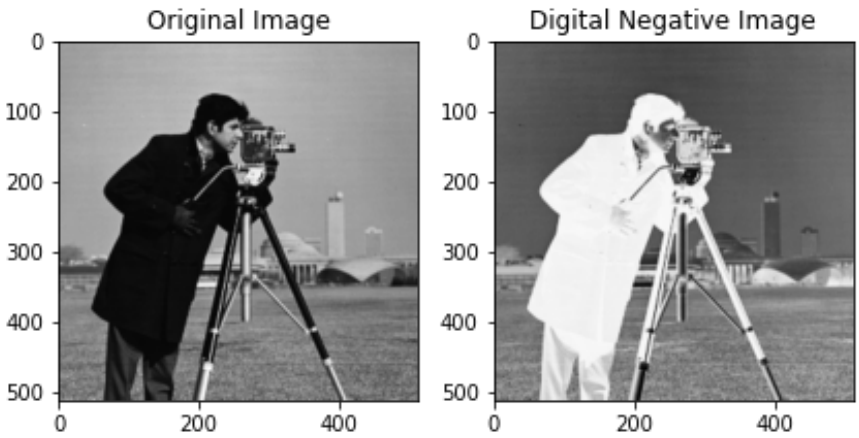
Collab Link: https://colab.research.google.com/drive/1wdhNh5eQgj67AT607m8AEjeqeosuEv1H?usp=sharing

```
#Importing all the required Libraries
import cv2
import matplotlib.pyplot as plt
import numpy as np
from skimage import io


##Image negative
img= cv2.imread('/content/cameraman.tif',0) # save the image in the same folder as this python fi
                                # In imread('cameraman.png',0)
                                # here 0 stands for grayscale image and 1 satnds for colour image


#Obtaining size of the original image (row and columns of image)
[m,n]=img.shape
print(m,n)


#To find the maximum grey level value in the image
L= img.max()    #here L is only a variable which is equivalent to (L-1) in our original formula
print(L)


# Maximum grey level value minus the original image gives the negative image
```

```
img_neg= L-img
cv2.imwrite('Cameraman_Negative.png', img_neg) # converting the np array img_neg to a png image
```

```
True
```

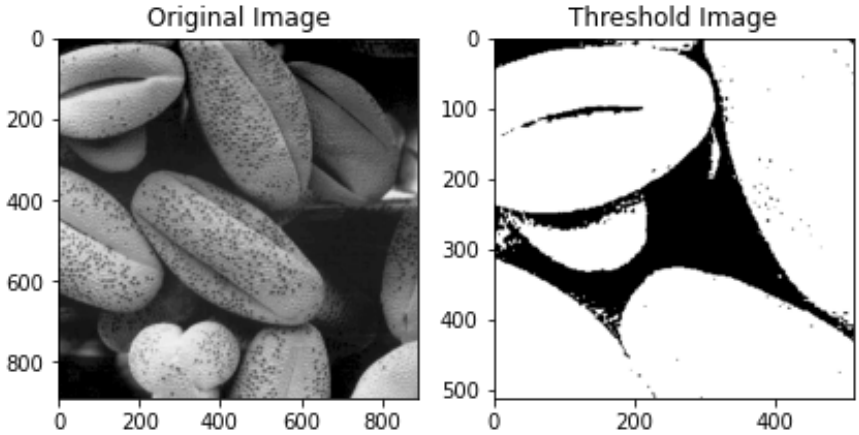
```
import matplotlib.pyplot as plt
plt.subplot(1,2,1)
plt.title('Original Image')
plt.imshow(img, cmap='gray')

plt.subplot(1,2,2)
plt.title('Digital Negative Image')
plt.imshow(img_neg,cmap='gray')

plt.tight_layout()
```




```
# 2. THRESHOLDING:  s= L-1 for r > threshold AND s= 0 for r < threshold
# Let threshold = T
# Let pixel value in the original be denoted by r
# Let pixel value in the new image be denoted by s
# If r<T, s= 0
# If r>T, s=255

#Reading the original image(0 means grayscale image)
img1= cv2.imread('/content/Fig0310(b)(washed_out_pollen_image).tif',0)

#Asking the threshold value from the user
T= int(input('Enter the threshold value: '))

# creating an array of zeros
img_thresh= np.zeros((m,n), dtype=int)

for i in range(m):
    for j in range(n):
        if img1[i,j]< T:
            img_thresh[i,j]= 0
        else:
            img_thresh[i,j] = 255

# Convert array to  png image
cv2.imwrite('/content/Fig0310(b)(washed_out_pollen_image).tif', img_thresh)
```

```
Enter the threshold value: 100
True
```

```
plt.subplot(1,2,1)
plt.title('Original Image')
plt.imshow(img1, cmap='gray')

plt.subplot(1,2,2)
plt.title('Threshold Image')
plt.imshow(img_thresh,cmap='gray')

plt.tight_layout()
```

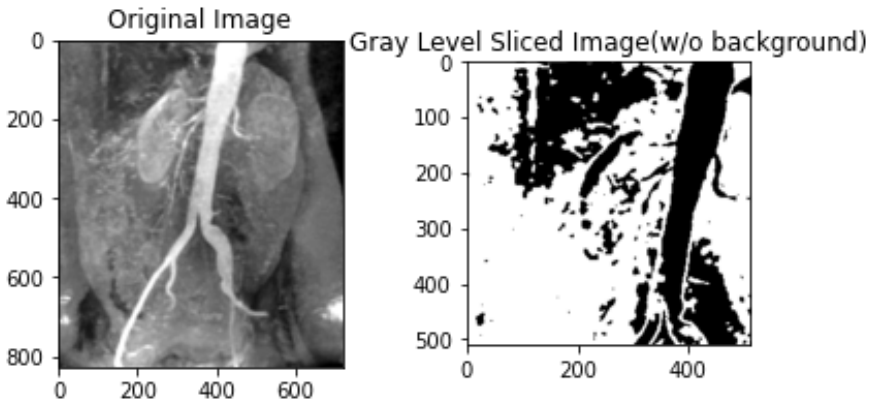



```
# Grey level slicing without background
# Let threshold values be  T1 and T2
# Let pixel value in the original be denoted by r
# Let pixel value in the new image be denoted by s
# If T1<r<T2, s= 255
# Else s = 0

#Reading the original image(0 means grayscale image)
img2= cv2.imread('/content/Fig0312(a)(kidney).tif',0)

T1= int(input('Enter the lower threshold value: ')) #Ask the lower threshold value from the user
T2= int(input('Enter the upper threshold value: ')) #Ask the upper threshold value from the user

# create a array of zeros
img_gls= np.zeros((m,n), dtype=int)

for i in range(m):
    for j in range(n):
        if T1 < img2[i,j] < T2:
            img_gls[i,j]= 255
        else:
            img_gls[i,j] = 0


cv2.imwrite('Fig0312(a)(kidney).png', img_gls) # Convert array to  png image
```

```
 Enter the lower threshold value: 80
 Enter the upper threshold value: 160
 True
```

```
plt.subplot(1,2,1)
plt.title('Original Image')
```

```
plt.imshow(img2, cmap='gray')

plt.subplot(1,2,2)
plt.title('Gray Level Sliced Image(w/o background)')
plt.imshow(img_gls,cmap='gray')

plt.subplots_adjust()
plt.tight_layout()
```




```
# Grey level slicing with background
# Let threshold values be  T1 and T2
# Let pixel value in the original be denoted by r
# Let pixel value in the new image be denoted by s
# If T1<r<T2, s= 255
# Else s = r

T1= int(input('Enter the lower threshold value: ')) #Ask the lower threshold value from the user
T2= int(input('Enter the upper threshold value: ')) #Ask the upper threshold value from the user

# create a array of zeros
img_gls1= np.zeros((m,n), dtype=int)

for i in range(m):
    for j in range(n):
        if T1 < img2[i,j] < T2:
            img_gls1[i,j]= 255
        else:
            img_gls1[i,j] = img2[i,j]


cv2.imwrite('Fig0312(a)(kidney)1.png', img_gls1) # Convert array to  png image
```

```
 Enter the lower threshold value: 80
 Enter the upper threshold value: 160
 True
```

```
plt.subplot(2,1,1)
plt.title('Original Image')
plt.imshow(img2, cmap='gray')

plt.subplot(2,1,2)
plt.title('Gray Level Sliced Image(with background)')
plt.imshow(img_gls1,cmap='gray')
```

```
plt.tight_layout()
```