NAME: Hayden Cordeiro **ROLL NO.: 05** **BATCH: A**

Experiment 7

Image Negation, Image Thresholding

Objective

Write a OpenCV program for Image Negative and Image Thresholding

Theory

Image Negative

The negative of an image with gray levels in the range [0, L-1] is obtained by using the negative transformation given by the expression

$$s = (L - 1) - r$$

This is according to the transformation s = T (r). In the above transformation, the intensity of the output image decreases as the intensity of the input increases. The type of processing is particularly suited for enhancing white or gray detail embedded in dark regions of an image especially when black areas are dominant in sight.

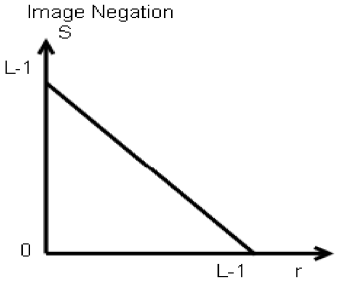


Image Thresholding

Thresholding is a simple process to separate the interested object from the background. It gives the binary image. The formula for achieving thresholding is as follows

s = 0 if r <= t

s = L-1 if r > t

Algorithm

Image Negation

1. Read i/p image
2. Read maximum gray level pixel of i/p image
3. Replace input image by (maximum – i/p) = o/p
4. Display o/p image

Image Thresholding

1. Read input image
2. Enter thresholding value t
3. If the image pixel is less than t replace it by zero.

4. If image pixel is > t replace it by 255

5. Display input image

6. Display threshold image

7. Write input image

8. Write threshold image

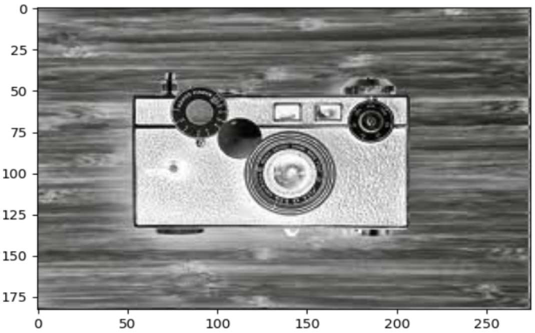
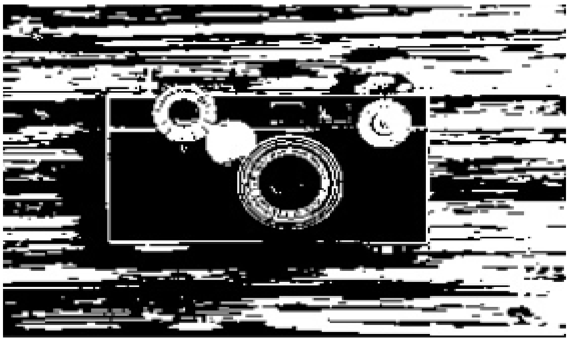
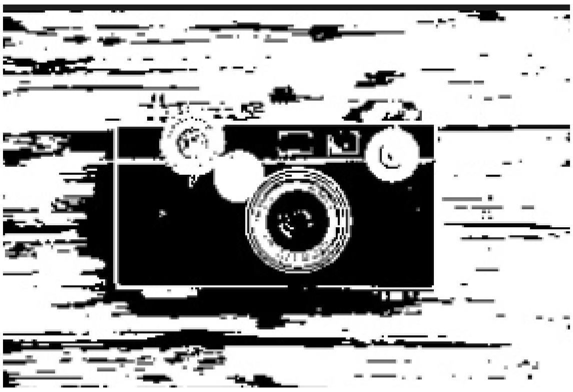
Program

IMAGE NEGATION

```
import cv2
import matplotlib.pyplot as plt
# Read an image
img_bgr = cv2.imread('cam.jpg', 1)
plt.imshow(img_bgr)
plt.show()
# Histogram plotting of the image
color = ('b', 'g', 'r')
for i, col in enumerate(color):
    histr = cv2.calcHist([img_bgr],
                         [i], None,
                         [256],
                         [0, 256])
plt.show()

# get height and width of the image
height, width, _ = img_bgr.shape
for i in range(0, height - 1):
    for j in range(0, width - 1):

        # Get the pixel value
        pixel = img_bgr[i, j]

        # Negate each channel by  subtracting it from 255
        # 1st index contains red pixel
        pixel[0] = 255 - pixel[0]

        # 2nd index contains green pixel
        pixel[1] = 255 - pixel[1]
        # 3rd index contains blue pixel
        pixel[2] = 255 - pixel[2]

        # Store new values in the pixel
        img_bgr[i, j] = pixel

# Display the negative transformed image
plt.imshow(img_bgr)
plt.show()

# Histogram plotting of the negative transformed image
color = ('b', 'g', 'r')
for i, col in enumerate(color):
    histr = cv2.calcHist([img_bgr],
                         [i], None,
                         [256],
                         [0, 256])

plt.show()
```

IMAGE THRESHOLDING:

```
# import opencv
import cv2

# Read image
src = cv2.imread("cam.jpg", cv2.IMREAD_GRAYSCALE)
# Basic threhold example
th, dst = cv2.threshold(src, 120, 255, cv2.THRESH_BINARY)
cv2.imwrite("openc2.jpg", dst)
```

Output	**IMAGE NEGATION:**  0 25 50 75 100 125 150 175 0 50 100 150 200 250 **IMAGE THRESHOLDING:** **Threshold: 150**  **Threshold: 120** 
Outcome	Ability to process images to get a negative image.Ability to process an image to highlight a specific range of Gray values depending on threshold value.
Questions	1. **How to get a negative image.** The negative of an image is achieved by replacing the intensity 'i' in the original image by 'i-1', i.e. the darkest pixels will become the brightest and the brightest pixels will become the darkest. Image negative is produced by subtracting each pixel from the maximum intensity value. For example in an 8-bit grayscale image, the max intensity value is 255, thus each pixel is subtracted from 255 to produce the output image. 2. **Explain local & global thresholding**

	Local thresholding is used to convert an image consisting of gray scale pixels to just black and white scale pixels. Usually a pixel value of 0 represents white and the value 255 represents black with the numbers from 1 to 254 representing different gray levels. **Global thresholding** determines the threshold value based on the histogram of the overall pixel intensity distribution of the image. 3. **Discuss some application of thresholding.** For differentiating between normal and abnormal blood for leukemia detection. We use thresholding as a way to select areas of interest of an image, while ignoring the parts we are not concerned with.