Lab1

Title: Conversion of color image into gray-scale image.

Background:

A grayscale image is a type of digital or physical image where each pixel is represented by a single value that corresponds to its level of brightness. It lacks color information and uses shades of gray to depict the varying levels of lightness or darkness in the image. Grayscale images are often used in applications like photography, medical imaging, and image processing for their simplicity and ability to convey visual information without color. These images are monochromatic and typically use 8-bit values ranging from 0 (black) to 255 (white) to represent different shades of gray.

Tools Used:

Visual studio code

Source Code:

```
import cv2
color_image = cv2.imread(‘me.jpg')
gray_image = cv2.cvtColor(color_image, cv2.COLOR_BGR2GRAY)
cv2.imshow('Color Image', color_image)
cv2.imshow('Grayscale Image', gray_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:

Conclusion:

This Python code using the OpenCV library converts a color image to grayscale. It loads the color image, applies the conversion, and saves the resulting grayscale image, with an optional display step.

Lab 2

Title: Conversion of gray-scale image into black and white image.

Background:

The conversion of a grayscale image into a black and white image, also known as binarization, is a process that involves simplifying the image to only two levels of intensity: black and white. In a black and white image, each pixel is represented by either black (usually denoted as 0) or white (usually denoted as 255), based on a certain threshold value. This transformation is often used to emphasize specific details in an image or to prepare it for further analysis. The process typically involves comparing pixel intensities to a threshold value and setting them to black or white accordingly, creating a high-contrast representation.

Tools Used:

Visual studio code

Source Code:

```
import cv2 import numpy
img=cv2.imread('me.jpg')
img = cv2.resize(img,(400, 500))
image = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
(thresh, bimage) = cv2.threshold(image, 127, 255, cv2.THRESH_BINARY)
cv2.imshow("image",bimage) cv2.waitKey(0) cv2.destroyAllWindows()
```

Output:

Conclusion:

In conclusion, this Python code utilizing OpenCV effectively converts a grayscale image into a black and white (binary) image through thresholding. It loads the grayscale image, applies a user-defined threshold, generates the binary image, and optionally displays it. This process simplifies the image to just two intensity levels, making it suitable for tasks such as image segmentation and feature extraction. If we decrease the intensity level then image become dim.

Lab 1 Output:

```
Mohan Thapa Mashrangi
Enter the transition matrix (e.g., 0.7 0.3 0.4 0.6 for a 2x2 matrix):
0.7
0.3
0.5
0.5
Transition matrix:
0.700000 0.300000
0.500000 0.500000
Enter the current state matrix (e.g., 10 for state 1):
1
0
Enter the number of steps to generate:
4
State sequence probabilities:
1.000000 0.000000
0.700000 0.300000
0.640000 0.360000
0.628000 0.372000
0.625600 0.374400
```