****

**Lab Manual**

**Computer Engineering – Artificial Intelligence**

**B. Tech. Year – II, 5th Semester, Academic Year (2023)**

**Subject Code: 01AI0504**

*Subject Name: Digital Image Processing*

*Name: Basid Al Siddik Shammo*

*Enrollment Number: 92100151052*

**Aim-** Normalization and Histogram Equalization

**Description-**

**Normalization:**

Normalization is a process of scaling the pixel values of an image to a specific range, often [0, 1] or [0, 255]. It is a linear transformation that ensures all pixel values fall within the desired range.

The main purpose of normalization is to eliminate the effect of varying intensity levels between images, making them more comparable and suitable for processing by various algorithms.

**Histogram Equalization:**

**It** is a non-linear technique that redistributes the pixel intensity values in an image's histogram to achieve a more uniform distribution.

The histogram of an image represents the frequency of occurrence of each pixel intensity value. In an ideal case, where all intensities are equally represented, the histogram will have a flat shape.

By equalizing the histogram, the contrast of the image is improved, and details that were previously hidden due to a limited range of intensities become more visible.

**Task- I**

**Code-**

import cv2

import numpy as np

import matplotlib.pyplot as plt

def normalize\_image(image):

image = image.astype(np.float32)

image\_normalized = image / 255.0

return image\_normalized

def histogram\_equalization(image):

if len(image.shape) == 3:

image = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

equalized\_image = cv2.equalizeHist(image)

return equalized\_image

if \_\_name\_\_ == "\_\_main\_\_":

image = cv2.imread("ex5\_do\_nawab-100x100.png")

normalized\_image = normalize\_image(image)

equalized\_image = histogram\_equalization(image)

plt.figure(figsize=(12, 6))

plt.subplot(1, 3, 1)

plt.imshow(cv2.cvtColor(image, cv2.COLOR\_BGR2RGB))

plt.title("Original Image")

plt.axis("off")

plt.subplot(1, 3, 2)

plt.imshow(normalized\_image, cmap="gray")

plt.title("Normalized Image")

plt.axis("off")

plt.subplot(1, 3, 3)

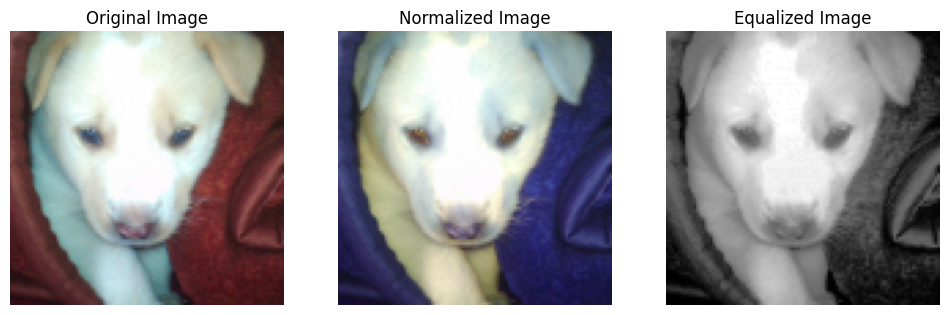
plt.imshow(equalized\_image, cmap="gray")

plt.title("Equalized Image")

plt.axis("off")

plt.show()

**Result-**

****

**Task- II**

**Code-**

import cv2

import numpy as np

import matplotlib.pyplot as plt

def normalize\_image(image):

image = image.astype(np.float32)

image\_normalized = image / 255.0

return image\_normalized

def histogram\_equalization(image):

if len(image.shape) == 3:

image = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

equalized\_image = cv2.equalizeHist(image)

return equalized\_image

if \_\_name\_\_ == "\_\_main\_\_":

image = cv2.imread("ex5\_clahe\_1.png")

equalized\_image = histogram\_equalization(image)

plt.figure(figsize=(12, 6))

plt.subplot(1, 3, 1)

plt.imshow(cv2.cvtColor(image, cv2.COLOR\_BGR2RGB))

plt.title("Original Image")

plt.axis("off")

plt.subplot(1, 3, 2)

plt.imshow(normalize\_image, cmap="gray")

plt.title("Normalized Image")

plt.axis("off")

plt.subplot(1, 3, 3)

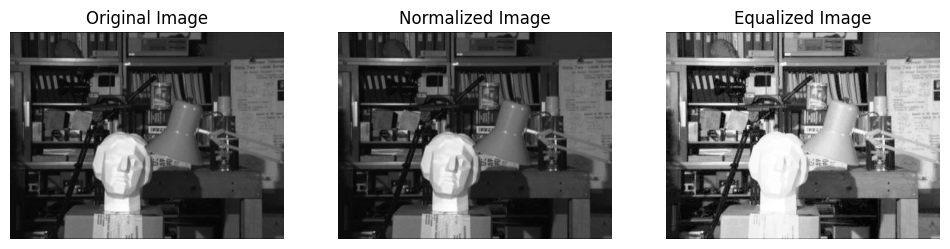
plt.imshow(equalized\_image, cmap="gray")

plt.title("Equalized Image")

plt.axis("off")

plt.show()

**Result-**



**Conclusion-**