Academic Year 2024-25



**DEPARTMENT OF INFORMATION TECHNOLOGY**

# COURSE CODE: DJS22ITL603

# COURSE NAME: Image Processing and Computer Vision Laboratory CLASS: T Y B. TECH

**NAME: Anish Sharma**

**Sap ID: 60003220045**

**EXPERIMENT NO. 4 CO/LO:** Apply Image Enhancement Techniques.

**AIM / OBJECTIVE:** To perform Histogram Transformations using OpenCV/Pillow Library

**EXERCISE**

Perform histogram transformations on any Image for optimizing the image characteristics using PIL/OpenCV:

1. Read a grayscale Image
2. Display the intensity values in an Image
3. Perform Intensity transformation for image processing (vary the brightness/contrast levels in image)
4. Compare the histogram generated for normal, bright, low, and High contrast images.
5. Perform histogram equalization on an Image.
6. Apply the same to a colored image too.

import cv2

import numpy as np

import matplotlib.pyplot as plt

def display\_image(title, img):

    """Displays an image using matplotlib."""

    plt.figure(figsize=(6,6))

    plt.title(title)

    plt.axis('off')

    plt.imshow(img, cmap='gray')

    plt.show()

def plot\_histogram(image, title):

    """Plots histogram of an image."""

    plt.figure(figsize=(6,4))

    plt.hist(image.ravel(), bins=256, range=[0,256], color='black', alpha=0.7)

    plt.title(title)

    plt.xlabel('Pixel Intensity')

    plt.ylabel('Frequency')

    plt.show()

def adjust\_brightness\_contrast(image, alpha, beta):

    """Adjusts brightness and contrast of an image."""

    adjusted = cv2.convertScaleAbs(image, alpha=alpha, beta=beta)

    return adjusted

def histogram\_equalization(image):

    """Performs histogram equalization."""

    return cv2.equalizeHist(image)

# Load images

gray\_image = cv2.imread('/content/oldman.jpg', cv2.IMREAD\_GRAYSCALE)

color\_image = cv2.imread('/content/oldmancolored.jpg')

gray\_from\_color = cv2.cvtColor(color\_image, cv2.COLOR\_BGR2GRAY)

# Display original images

display\_image('Grayscale Image', gray\_image)

display\_image('Converted Grayscale from Color', gray\_from\_color)

# Adjust brightness and contrast

bright\_image = adjust\_brightness\_contrast(gray\_image, 1.2, 50)

dark\_image = adjust\_brightness\_contrast(gray\_image, 0.8, -50)

high\_contrast = adjust\_brightness\_contrast(gray\_image, 2.0, 0)

low\_contrast = adjust\_brightness\_contrast(gray\_image, 0.5, 0)

display\_image('Brightened Image', bright\_image)

display\_image('Darkened Image', dark\_image)

display\_image('High Contrast Image', high\_contrast)

display\_image('Low Contrast Image', low\_contrast)

# Plot histograms

plot\_histogram(gray\_image, 'Histogram - Original')

plot\_histogram(bright\_image, 'Histogram - Bright Image')

plot\_histogram(dark\_image, 'Histogram - Dark Image')

plot\_histogram(high\_contrast, 'Histogram - High Contrast')

plot\_histogram(low\_contrast, 'Histogram - Low Contrast')

# Perform histogram equalization

equalized\_image = histogram\_equalization(gray\_image)

display\_image('Histogram Equalized Image', equalized\_image)

plot\_histogram(equalized\_image, 'Histogram - Equalized Image')

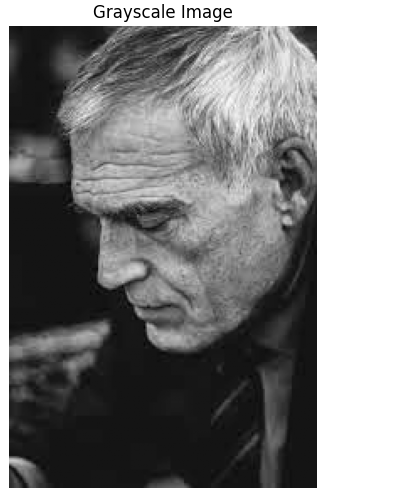
# Apply histogram equalization to a colored image

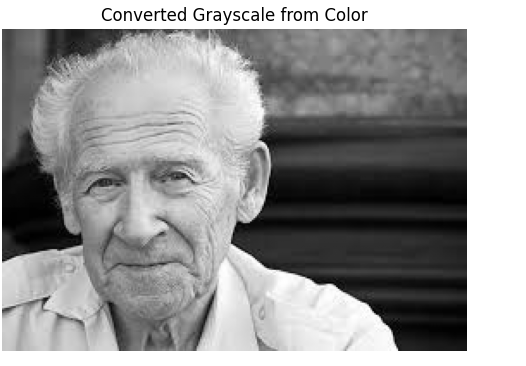
color\_yuv = cv2.cvtColor(color\_image, cv2.COLOR\_BGR2YUV)

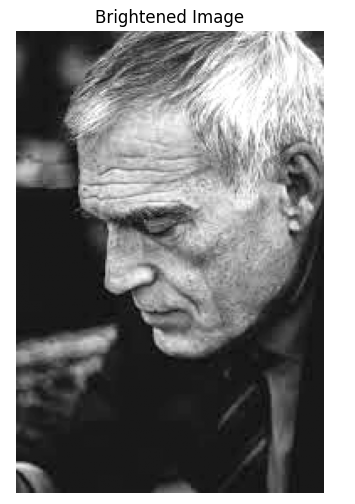
color\_yuv[:,:,0] = cv2.equalizeHist(color\_yuv[:,:,0])

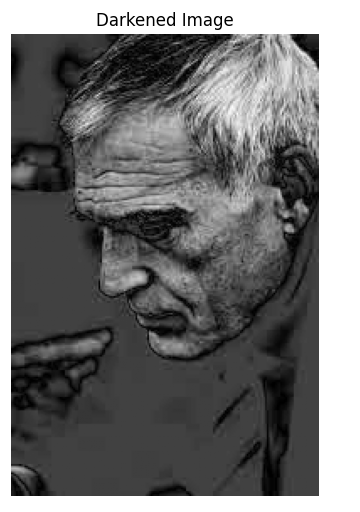
equalized\_color = cv2.cvtColor(color\_yuv, cv2.COLOR\_YUV2BGR)

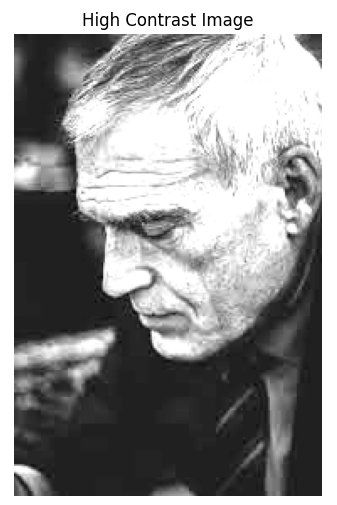
display\_image('Histogram Equalized Color Image', cv2.cvtColor(equalized\_color, cv2.COLOR\_BGR2RGB))

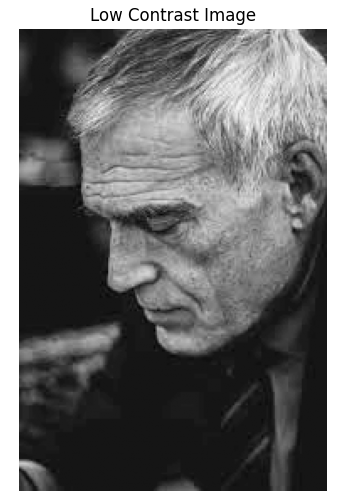


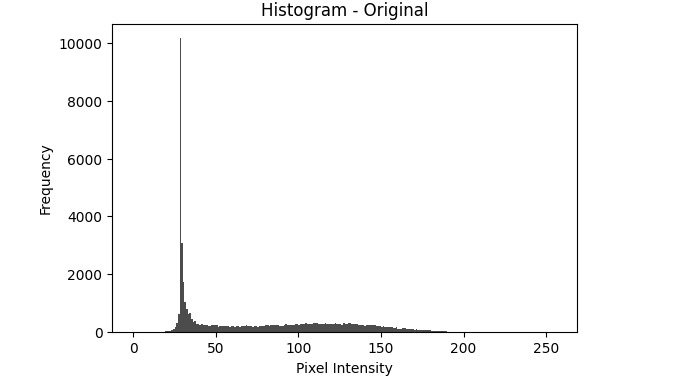


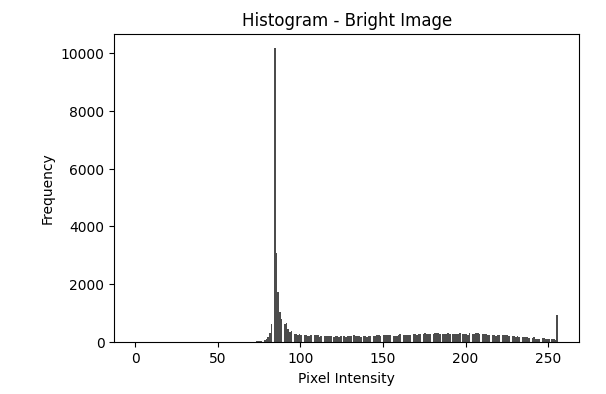


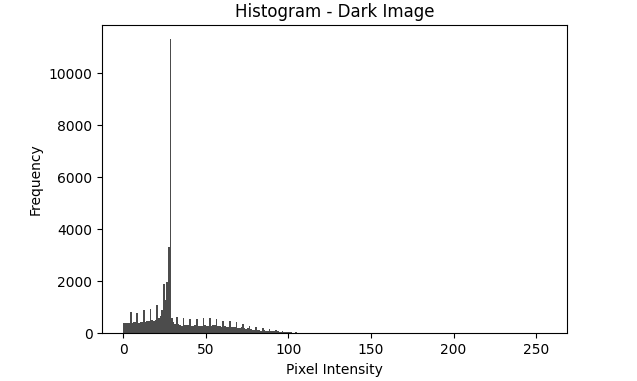


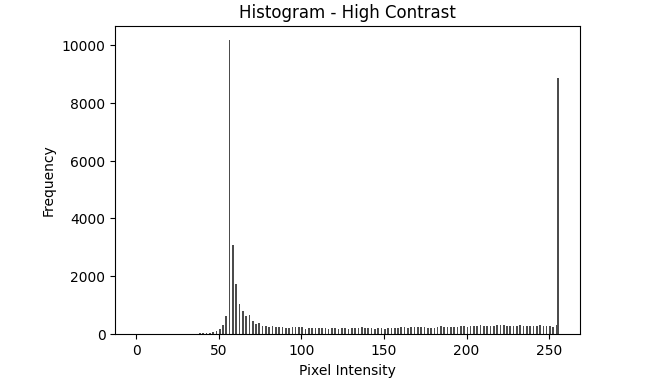


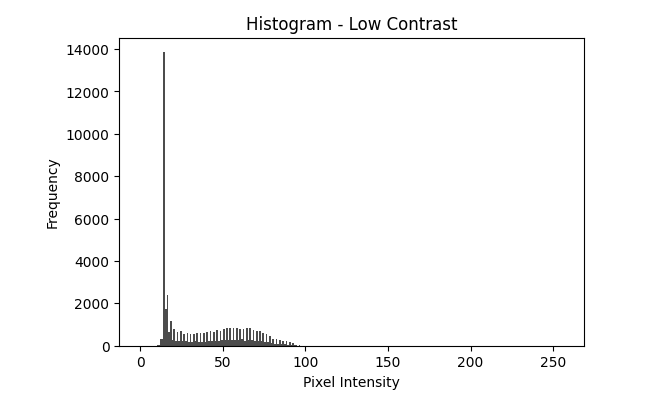


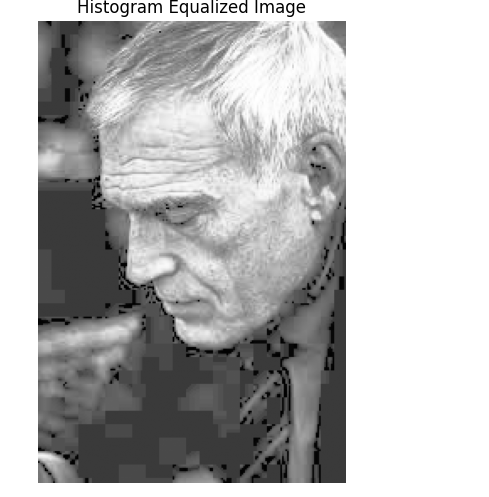


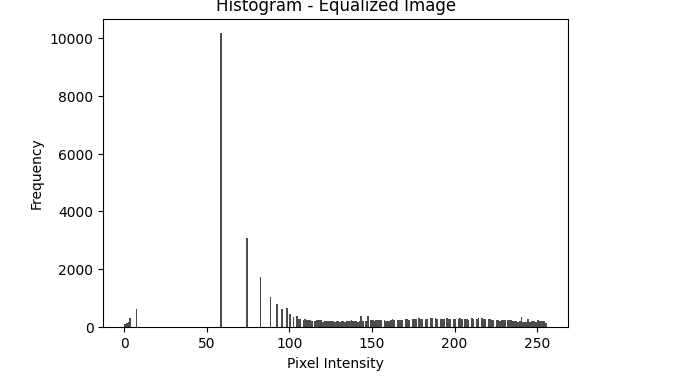


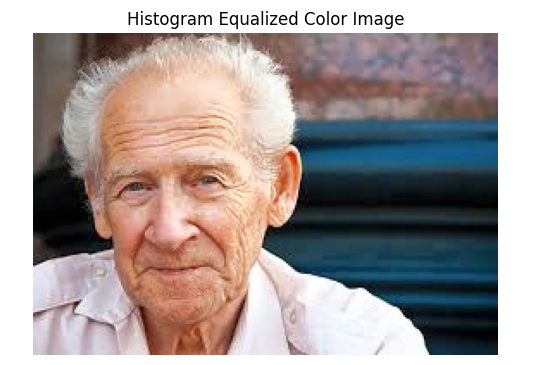












**QUESTIONS:**

* What happens if histogram equalization is applied twice?
* Describe what you can infer from the Histograms generated for different images.
* Describe the shortcomings (if any) in histogram equalization techniques.

**REFERENCES:**

**Website References:**

1. Towards Data Science, “Histogram Equalization: A Simple Way to Improve the Contrast of Your Image,” *Towards Data Science*. Available: [https://towardsdatascience.com/histogramequalization-a-simple-way-to-improve-the-contrast-of-your-image-bcd66596d815.](https://towardsdatascience.com/histogram-equalization-a-simple-way-to-improve-the-contrast-of-your-image-bcd66596d815)
2. OpenCV, “Histogram Equalization,” *OpenCV Documentation*. Available: [https://docs.opencv.org/3.4/d4/d1b/tutorial\_histogram\_equalization.html.](https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html)