### **Assignment Number: 03**

### **Name**: Mihir Unmesh Patil **Roll No**: TYCOC213 **Batch**: C/C-3

### **Title:**

### **Plot the histogram of an image and perform histogram equalization.**

### 

### **Aim:**

### The aim of this project is to demonstrate the concept and implementation of **Histogram Equalization**, a technique used to improve the contrast of images. The technique will be applied using Python and OpenCV to adjust the intensity distribution in grayscale images, making the image more visually informative.

### **Objectives:**

### **Histogram Equalization**: To enhance the contrast of an image by spreading out the most frequent intensity values, improving the overall visibility of details in an image.

### **Comparison of Histograms**: To visualize the effect of histogram equalization on the distribution of pixel intensities before and after the process.

### **Image Visualization**: To display the original image and the equalized image, showing their pixel-level differences.

### **Image Enhancement**: To improve the visual quality of an image in low-contrast or underexposed conditions.

### These objectives are aimed at understanding how histogram equalization works to enhance the visual quality of an image, especially in terms of pixel intensity distribution.

### 

### **Theory:**

#### **1. Histogram Equalization:**

### Histogram equalization is a method used to enhance the contrast of an image by modifying its intensity distribution. It aims to ensure that the image has a uniform histogram, where the pixel intensities are evenly distributed across the available range. This is achieved by transforming the intensity levels such that the cumulative distribution function (CDF) of the image's pixel intensities is spread over the entire range of possible values (usually 0-255 for an 8-bit grayscale image).

### This technique works as follows:

### Compute the histogram of the image (frequency distribution of pixel intensities).

### Calculate the cumulative distribution function (CDF) of the histogram.

### Use the CDF to map the original pixel intensities to new ones that spread the values more evenly.

### Replace the original pixel intensities with the new values, resulting in an image with improved contrast.

### Histogram equalization is especially useful for improving images with poor contrast or images that are overexposed or underexposed. The enhanced image will have a greater range of intensities, making features in the image more distinguishable.

### 

#### **2. How Histogram Equalization Works:**

### **Step 1**: Compute the histogram of the original image.

### **Step 2**: Calculate the cumulative distribution function (CDF).

### **Step 3**: Normalize the CDF.

### **Step 4**: Map the original image intensities to new intensities based on the normalized CDF.

### **Step 5**: Display the original and equalized images along with their histograms for comparison.

### By spreading out the most frequent pixel intensities, the technique can reveal hidden details, making the image more informative and easier to analyze.

### 

### **Methodology:**

### The implementation of histogram equalization was done using Python and the OpenCV library, following the steps outlined below:

### **Loading the Image**: The input image is read in grayscale mode using OpenCV’s cv2.imread() function.

### **Plotting the Original Histogram**: The histogram of the original image is computed using cv2.calcHist() to understand the distribution of pixel intensities.

### **Performing Histogram Equalization**: OpenCV’s cv2.equalizeHist() function is used to apply histogram equalization to the image.

### **Plotting the Equalized Histogram**: The histogram of the equalized image is computed to compare it with the original.

### **Displaying the Results**: Both the original and equalized images are displayed using cv2\_imshow() (for Google Colab compatibility). The corresponding histograms are plotted using matplotlib.

### Additional steps include printing a portion of the matrix of the image (before and after equalization) to observe the change in pixel values.

### **Code Implementation:**

### [Assignment\_03\_IVP.ipynb](https://colab.research.google.com/drive/1ar_sFxWtek6lgX0r3F2AmUFVsnq-JAH5?usp=sharing)

### 

### **Conclusion:**

### This project demonstrates the practical application of **Histogram Equalization** using OpenCV to enhance the contrast of images. By applying this technique, images with poor contrast are improved, making underlying details more visible. The process involves adjusting pixel intensities so that they are more evenly distributed across the available intensity range, which significantly enhances the visual quality of the image. This technique is widely used in various image processing applications such as medical imaging, satellite imaging, and even in fields like computer vision and machine learning where better image features lead to more accurate predictions.

### Histogram equalization has proven to be a valuable tool for image enhancement, especially in scenarios with uneven lighting or low-contrast images.

### 