EXPERIMENT NO.7

To perform histogram equalization.

Date of Performance	
Date of Checking	

Title: To read the image, plot its histogram and equalization of histogram for image enhancement.

Objectives:

- To study histogram of various image.
- To equalize and enhance histogram of given image

Aim: Write a program for histogram plotting and equalization.

Software: Scilab

Theory:

In addition to standard mathematical functions, it is sometimes advantageous to construct a transfer function for a specific image. We desire a specific algorithm that gives reproducible and optimal results. The most popular of these methods is called histogram equalization. To understand it, we must begin with the image brightness histogram. The conventional histogram plot shows the number of pixels in the image having each of the 256 possible values of stored brightness. Peaks in the histogram correspond to the more common brightness values, which often correspond to particular structures that are present. Valleys indicate brightness values that are less common in the image. The data can also be plotted as a cumulative curve, which is simply the integral or summation of the values. If this curve is used as the display transfer function, the result is a display in which all of the available 256 brightness values are equally used. The histogram of this processed image shows this uniform distribution and a linear cumulative plot. This procedure is called histogram equalization.

Generally, images have unique brightness histograms. Even images of different areas of the same sample or scene, in which the various structures present have consistent brightness levels wherever they occur, will have different histograms, depending on the area covered by each structure. Changing the overall illumination or camera settings will shift the peaks in the histogram. In addition, most real images exhibit some variation in brightness within features (e.g., from the edge to the center) or in different regions.

The objective of histogram equalization is to map an input image to an output image such that its histogram is uniform after the mapping.

Let r represent the gray levels in the image to be enhanced and s is the enhanced output with a transformation of the form s=T(r).

Assumption:

1. *T(r)* is single-valued and monotonically increasing in the interval *[0, 1],* which preserves the order from black to white in the gray scale.
2. *0 ≤ T(r) ≤1* for *0 ≤ r ≤1*, which guarantees the mapping is consistent with the allowed range of pixel values.

If $P_r(r)$ and $T(r)$are known and $T^{-1}(s)$ satisfies condition (a), the pdf of the transformed gray levels is

$$P_s(s) = P_r(r)\frac{dr}{ds}\Big|_{r=T^{-1}(s)}$$

If $s = T(r) = \int_0^r P_r(w)dw \quad for\ 0 \leq r \leq 1$, then we have

$\frac{ds}{dr} = P_r(r)$ And hence $P_s(s) = 1 \quad for\ 0 \leq s \leq 1$

Using a transformation function equal to the cumulative distribution of r produces an image whose gray levels have a uniform density, which implies an increase in the dynamic range of the pixels. In order to be useful for digital image processing, equations should be formulated in discrete form:

$$P_r(r_k) = \frac{n_k}{n} \quad \text{And} \quad S_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n} \quad where\ k = 0,1,2..........L-1$$

A plot of $P_r(r_k)$ versus r_k is actually a histogram, and the technique used for obtaining a uniform histogram is known as histogram equalization or histogram linearization.

Algorithm:

1. Start
2. Read input image and convert it to gray scale.
3. Plot the image along with its original histogram using inbuilt function.
4. Consider the temporary array, say (temp[])and store values of number of pixels having corresponding levels of intensity into this array with two nested for loops. Initialized with respect to size of the image(M*N).
5. Divide the array (temp (i)) with the size of the image to obtain the normalized histogram along with the probabilities of occurrences of gray scales.
6. Now perform cumulative addition for every value in the array i.e.

$$S_k = G_{max} \sum_{i=0}^{k} P(ri)$$

Where $P(ri) = normalised\ histogram$
G_{max}=maximum value of gray level present in the image.

7. Here G_{max}=255 and to implement the above function, initialize a for loop with respect to total number of gray levels (0-255).
8. Round off the decimal values to the closest gray level.
9. Display the histogram equalized image.
10. Also plot the histogram of this image and compare it with the histogram of original image plotted in step 3.
11. End.

Conclusion:

____

____

____

____