

| **Title:** Write a program to apply the global processing technique: Histogram equalization.on a digital image |
| --- |

**Objective:** To learn and understand the concept of histogram stretching and equalization in image enhancement operations.

**Expected Outcome of Experiment:**

| **CO** | **Outcome** |
| --- | --- |
| **CO4** | Design & implement algorithms for digital image enhancement, segmentation & restoration. |

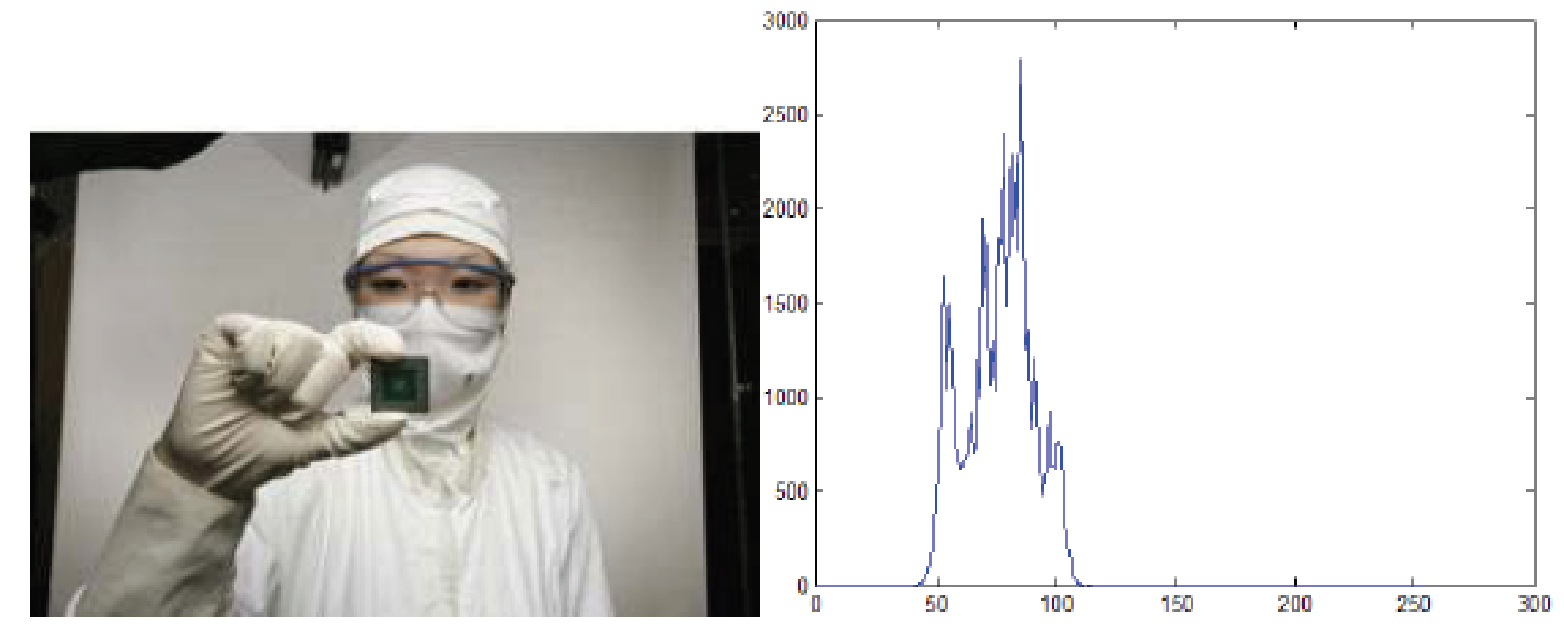
**Books/ Journals/ Websites referred:**

1. http://www.mathworks.com/support/
2. www.math.mtu.edu/~msgocken/intro/intro.html.
3. R. C.Gonsales R.E.Woods, “Digital Image Processing”, Second edition, Pearson Education
4. S.Jayaraman, S Esakkirajan, T Veerakumar “Digital Image Processing “Mc Graw Hill.
5. S.Sridhar,”Digital Image processing”, oxford university press, 1st edition."

**Pre Lab/ Prior Concepts:**

**Image histogram:**

In an image processing context, the histogram of an image normally refers to a histogram of the pixel intensity values. This histogram is a graph showing the number of pixels in an image at each different intensity value found in that image. For an 8-bit greyscale image there are 256 different possible intensities, and so the histogram will graphically display 256 numbers showing the distribution of pixels amongst those greyscale values. Histograms can also be taken of color images either individual histogram of red, green and blue channels can be taken, or a 3-D histogram can be produced, with the three axes representing the red, blue and green channels, and brightness at each point representing the pixel count. The exact output from the operation depends upon the implementation it may simply be a picture of the required histogram in a suitable image format, or it may be a data file of some sort representing the histogram statistics.

****

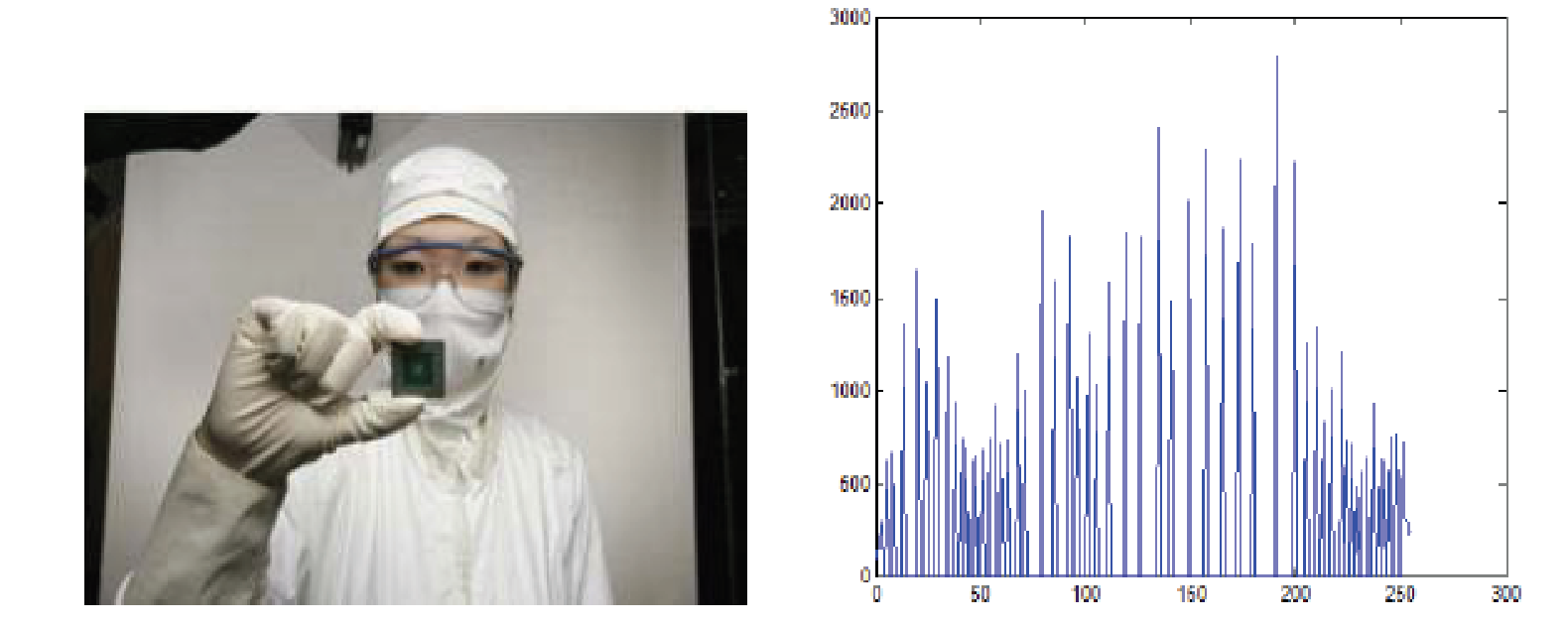
**Fig. 1 An image and its histogram**

**Histogram Equalization:**

A perfect image is one which has equal number of pixels in all its grey levels. hence our objective is not only to spread the dynamic range , but also to have equal pixels in all the grey levels. This technique is known as histogram equalization.

Basically the histogram equalization spreads out intensity values along the total range of values in order to achieve higher contrast. This method is especially useful when an image is

represented by close contrast values, such as images in which both the background and foreground are bright at the same time, or else both are dark at the same time. For example, the result of applying histogram equalization to the image in figure 1 is presented in figure 2.



**Fig. 2 New image and its equalized histogram**

**Description of cumulative histogram equalization:**

Here are the steps for implementing this algorithm.

1. Create the histogram for the image.

2. Calculate the cumulative distribution function histogram.

3. Calculate the new values through the general histogram equalization formula.

4. Assign new values for each gray value in the image.

Thus processed image is obtained by mapping each pixel with level rk into a corresponding pixel with level sk in o/p image. This transformation is called Histogram equalization

**Resources Used:** Matlab

**Implementation Details:**

**clc; clear; close all;**

**a = rgb2gray(imread('low\_cont.jpg'));**

**[rows, cols] = size(a);**

**hist = zeros(1, 256);**

**for i = 1:rows**

**for j = 1:cols**

**intensity = a(i, j) + 1;**

**hist(intensity) = hist(intensity) + 1;**

**end**

**end**

**pdf = hist / (rows \* cols);**

**cdf = cumsum(pdf);**

**new\_intensities = round(cdf \* 255);**

**output\_img = uint8(zeros(rows, cols));**

**for i = 1:rows**

**for j = 1:cols**

**output\_img(i, j) = new\_intensities(a(i, j) + 1);**

**end**

**end**

**hist\_eq = zeros(1, 256);**

**for i = 1:rows**

**for j = 1:cols**

**intensity = output\_img(i, j) + 1;**

**hist\_eq(intensity) = hist\_eq(intensity) + 1;**

**end**

**end**

**figure;**

**set(gcf, 'Position', [100, 100, 1000, 600]);**

**subplot(2, 2, 1);**

**imshow(a);**

**title('Original Image', 'FontSize', 12, 'FontWeight', 'bold', 'Color', 'k');**

**subplot(2, 2, 2);**

**bar(0:255, hist, 'FaceColor', [0 0.447 0.741], 'EdgeColor', 'none'); % Blue shade**

**title('Histogram of Original Image', 'FontSize', 12, 'FontWeight', 'bold', 'Color', 'k');**

**xlabel('Gray Level');**

**ylabel('Pixel Count');**

**grid on;**

**box on;**

**subplot(2, 2, 3);**

**imshow(output\_img);**

**title('Equalized Image', 'FontSize', 12, 'FontWeight', 'bold', 'Color', 'k');**

**subplot(2, 2, 4);**

**bar(0:255, hist\_eq, 'FaceColor', [0.85 0.325 0.098], 'EdgeColor', 'none'); % Red shade**

**title('Histogram of Equalized Image', 'FontSize', 12, 'FontWeight', 'bold', 'Color', 'k');**

**xlabel('Gray Level');**

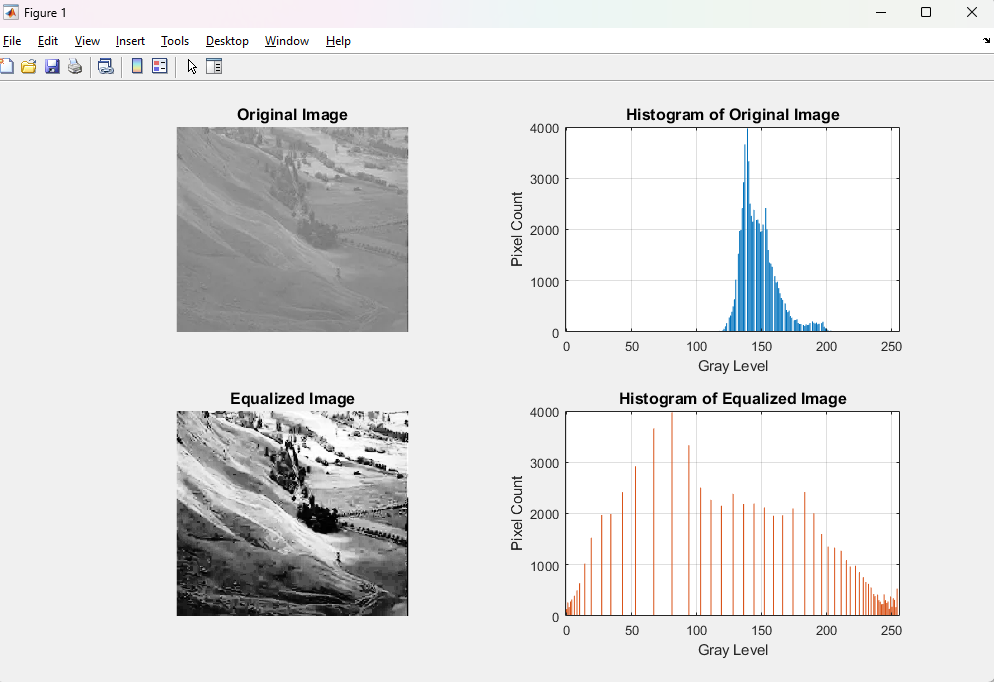
**ylabel('Pixel Count');**

**grid on;**

**box on;**

**imwrite(output\_img, 'equalized\_image.jpg');**

**OUTPUT:**



**Write Algorithm and Matlab commands used:**

1. Read the image and convert it to grayscale if needed.
2. Determine the image dimensions (rows, cols).
3. Compute the histogram of pixel intensities (0–255).
4. Normalize the histogram to get the PDF (divide by total number of pixels).
5. Compute the CDF by taking the cumulative sum of the PDF.
6. Scale and round the CDF to map original intensities to new intensities.
7. Generate the output image by replacing each original intensity with the corresponding new intensity.
8. Compute the histogram of the equalized image (if needed).
9. Display or save the results.

**Conclusion:-** Contrast stretching linearly enhances tones preserving order, while histogram equalization redistributes intensities, boosting contrast but sometimes sacrificing details.

**Date: 21 / 03 / 2025 Signature of faculty in-charge**

**Post Lab Descriptive Questions**

Compare between contrast stretching and histogram equalization.

Contrast stretching: • Uses a linear transformation to expand the range of pixel intensities to the full display range (e.g., 0 to 255). • Preserves the overall shape of the original histogram but redistributes values more evenly across the available intensity range. • Simpler and often faster, but may not always maximize image contrast in all regions.

Histogram equalization: • Uses a non-linear transformation based on the cumulative distribution of intensities. • Tries to produce a uniformly distributed histogram, enhancing contrast in darker and lighter regions. • Can introduce visual artifacts or over-enhancement in certain areas.