

EXPT:NO:4
DATE: 20.02.2024

IMPLEMENTATION OF HISTOGRAM AND HISTOGRAM EQUALIZATION TECHNIQUES USING MATLAB

AIM:

To implement the histogram and histogram equalization techniques using MATLAB:

SOFTWARE USED:

MATLAB version 2014a.

THEORY:

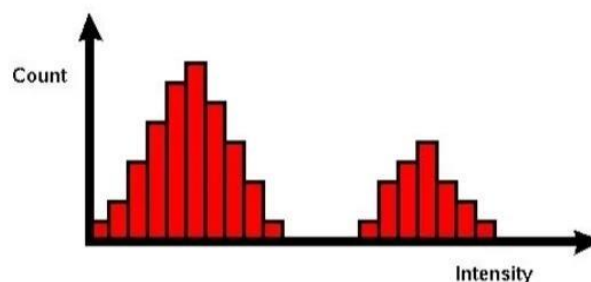
HISTOGRAM:

An image histogram is a graphical representation between number of pixels and Intensity values. It plots the number of pixels vs each intensity value.

Histogram is the representation of relative frequency of occurrence of various gray Level of an image.

histograms are utilized to understand the distribution of pixel intensities with within an image. Essentially, a histogram provides a visual representation of how frequently each intensity value occurs in the image. This information is crucial for image processing tasks, including contrast enhancement, brightness, adjustment, and image segmentation

- X axis represents the intensity values or colour values(ranging from 0 to 255)
- Y axis represents the frequency or count of pixels with those intensity or Colour values.

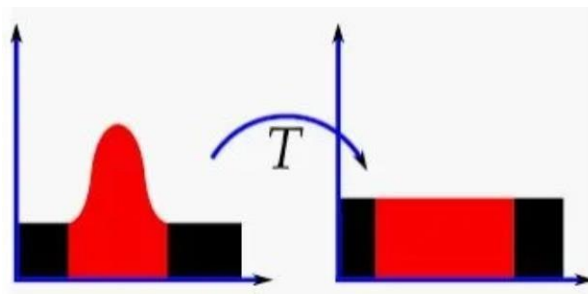


Histogram of an Image

HISTOGRAM EQUALIZATION:

Histogram Equalization is an image processing technique that adjusts the contrast of an image by using its histogram. To enhance the image's contrast, it spreads out the most frequent pixel intensity values or stretches out the intensity range of the image. By accomplishing this, histogram equalization allows the image's areas with lower contrast to gain a higher contrast.

Histogram Equalization can be used when you have images that look washed out because they do not have sufficient contrast. In such photographs, the light and dark areas blend together creating a flatter image that lacks highlights and shadows



Graphical Representation of Histogram
Equalization

TYPES OF HISTOGRAM EQUALIZATION:

- Global histogram equalization
- Local histogram equalization
- Adaptive histogram equalization
- Contrast limited adaptive histogram equalization

PROGRAM:

```
% Read the input image
input_image = imread('sample_img.jpg');
% Convert the input image to grayscale if it's a color image if
size(input_image, 3) == 3
input_image_gray = rgb2gray(input_image); else
input_image_gray = input_image;
end

% Perform histogram equalization
output_image_gray = histeq(input_image_gray);
% Compute histograms of original and equalized images
[orig_counts, orig_bins] = imhist(input_image_gray);
[eq_counts, eq_bins] = imhist(output_image_gray);
```

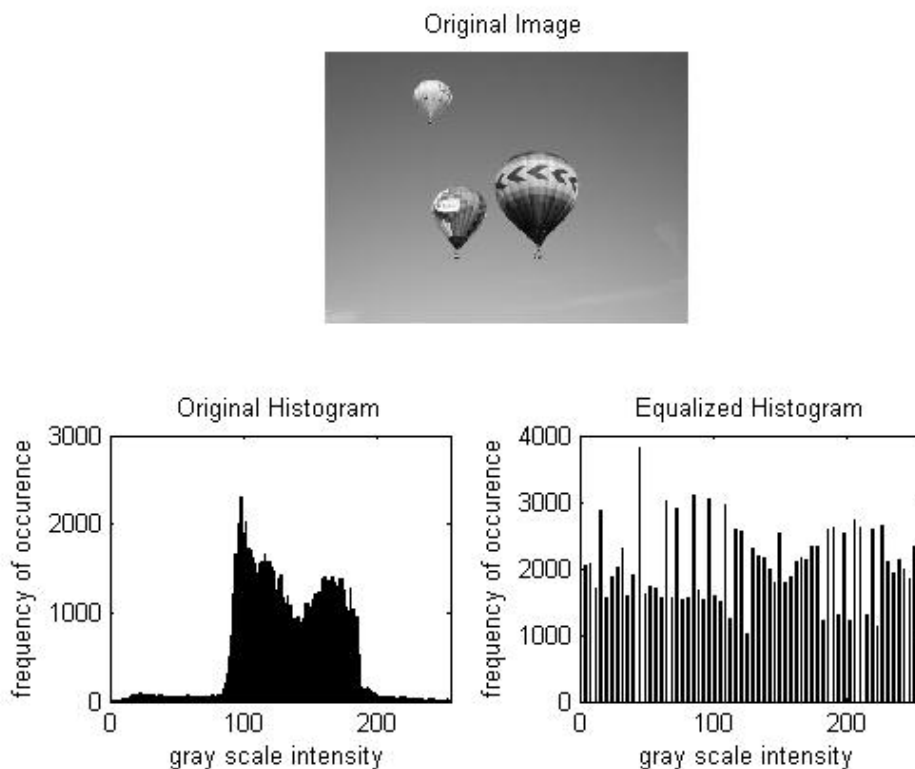
```

% Display the original and equalized images figure;
subplot(2, 2, [1, 2]);
imshow(input_image_gray);
title('Original Image');
subplot(2, 2, [3, 4]);
imshow(output_image_gray);
title('Equalized Image');

% Plot histograms subplot(2,
2, 3);
bar(orig_bins, orig_counts, 'BarWidth', 1);
xlim([0, 255]);
xlabel('gray scale intensity') ylabel('frequency
of occurrence')
title('Original Histogram');
subplot(2, 2, 4);
bar(eq_bins, eq_counts, 'BarWidth', 1);
xlim([0, 255]);
xlabel('gray scale intensity')
ylabel('frequency of occurrence')
title('Equalized Histogram');

```

OUTPUT:



RESULT:

Thus the implementation of histogram and histogram equalization is done through MATLAB software and the output is verified.