VEERMATA JIJABAI
TECHNOLOGICAL
INSTITUTE, MUMBAI

Yash Rajput
211060042
Third Year
Electronics Engineering

IMAGE AND VIDEO PROCESSING LAB - EXPERIMENT 1

Date: 11th Mar, 2024

To understand Histogram Equalisation using MATLAB

AIM

To use the functions of MATLAB to perform Histogram Equalisation and explore other MATLAB functions related to image and video processing

APPARATUS:

- MATLAB 2023B
- Image in tiff format

THEORY:

Histogram equalization is a technique commonly used in image processing to enhance the contrast of an image by redistributing the intensity values. The basic idea behind histogram equalization is to transform the histogram of an image in such a way that the intensity values are spread out evenly across the entire dynamic range

The process involves computing the cumulative distribution function (CDF) of the image's histogram and then mapping the original intensity values to new values based on this CDF. This mapping effectively stretches the intensity values to cover the entire available range, resulting in improved contrast.

Steps to perform Histogram Equalization:

1. **Compute the Histogram:** Calculate the input image's histogram, which represents the frequency of occurrence of each intensity level.
2. **Compute the Cumulative Distribution Function (CDF):** Calculate the cumulative sum of the histogram values, which represents the cumulative distribution of intensities. This step essentially normalizes the histogram.
3. **Normalize the CDF:** Scale the cumulative distribution function to span the entire range of possible intensity values (usually 0 to 255 for an 8-bit image).
4. **Map the Intensity Values:** Use the normalized CDF to map each original intensity value to a new value. This mapping is typically done using linear interpolation.
5. **Generate the Equalized Image:** Replace each pixel's intensity value in the original image with its corresponding mapped value.

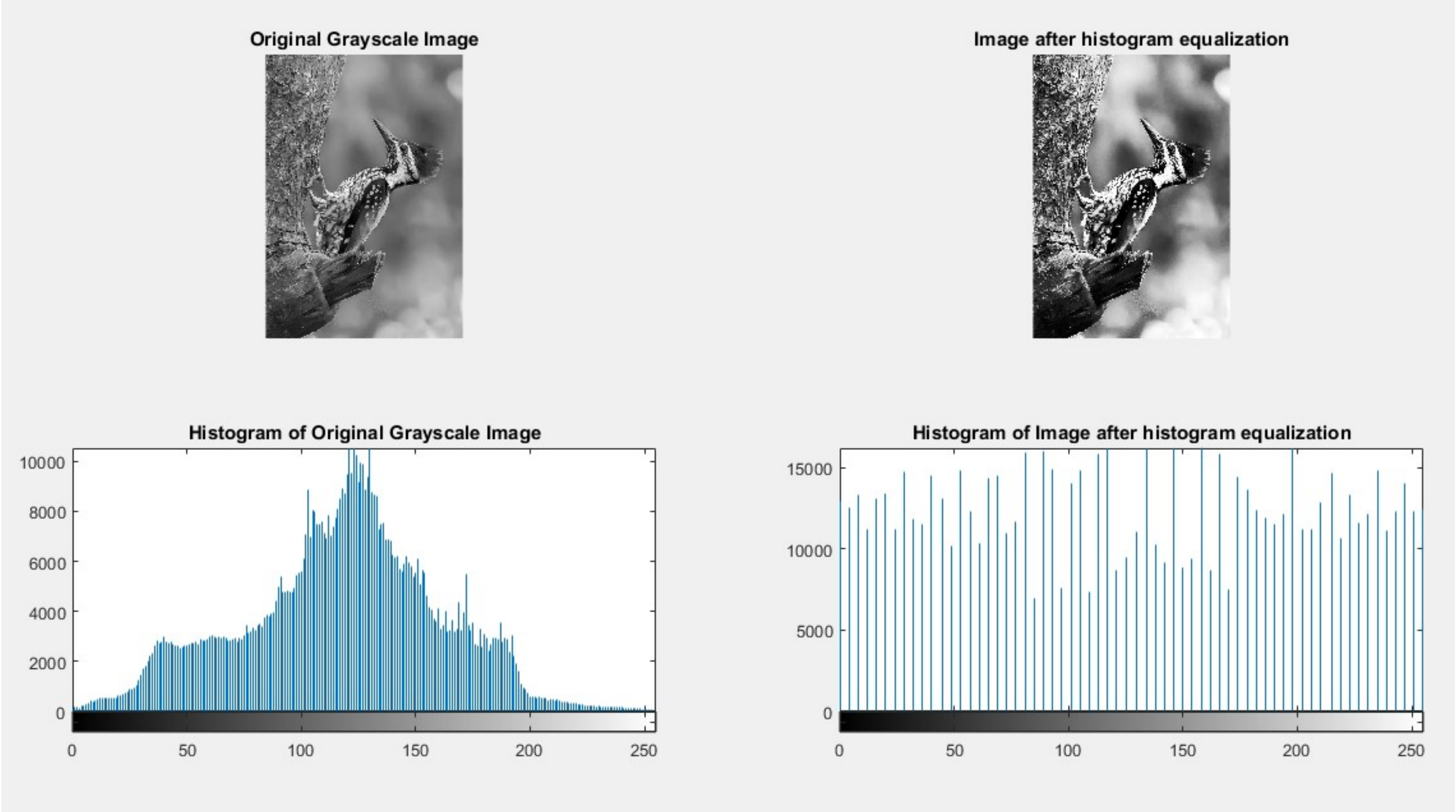

Figure 1 : Histogram Equalization

The **Wiener filter** is a popular technique used in signal processing and image processing for noise reduction and signal enhancement. Named after Norbert Wiener, the Wiener filter aims to minimize the mean square error between the original signal and the estimated signal while taking into account the statistical properties of the noise.

How does the Wiener filter work:

1. **Signal and Noise Modeling:** The Wiener filter assumes that both the signal and the noise are stochastic processes with known statistical properties. Specifically, it requires knowledge of the power spectral densities (PSDs) of the signal and the noise.

2. **Frequency Domain Representation:** The Wiener filter operates in the frequency domain, where signals and noise are represented by their Fourier transforms. By analyzing the frequency content of the signal and the noise, the Wiener filter can design an optimal filter to minimize the effects of noise.

3. **Filter Design:** The Wiener filter is designed to be an optimal linear filter that minimizes the mean square error between the original signal and the estimated signal. It achieves this by attenuating the noisy frequency components while preserving the important features of the signal.

4. **Adaptive Filtering:** In some applications, such as image processing, the characteristics of the signal and the noise may vary spatially or temporally. In such cases, an adaptive Wiener filter can be used, which adjusts its filter coefficients based on the local statistics of the signal and the noise.

The Wiener filter is widely used in various applications, including audio signal processing, image restoration, and communication systems. However, it's important to note that the effectiveness of the Wiener filter depends on the accuracy of the signal and noise models and the degree of correlation between them. In practice, trade-offs may need to be made between noise reduction and preservation of signal details.

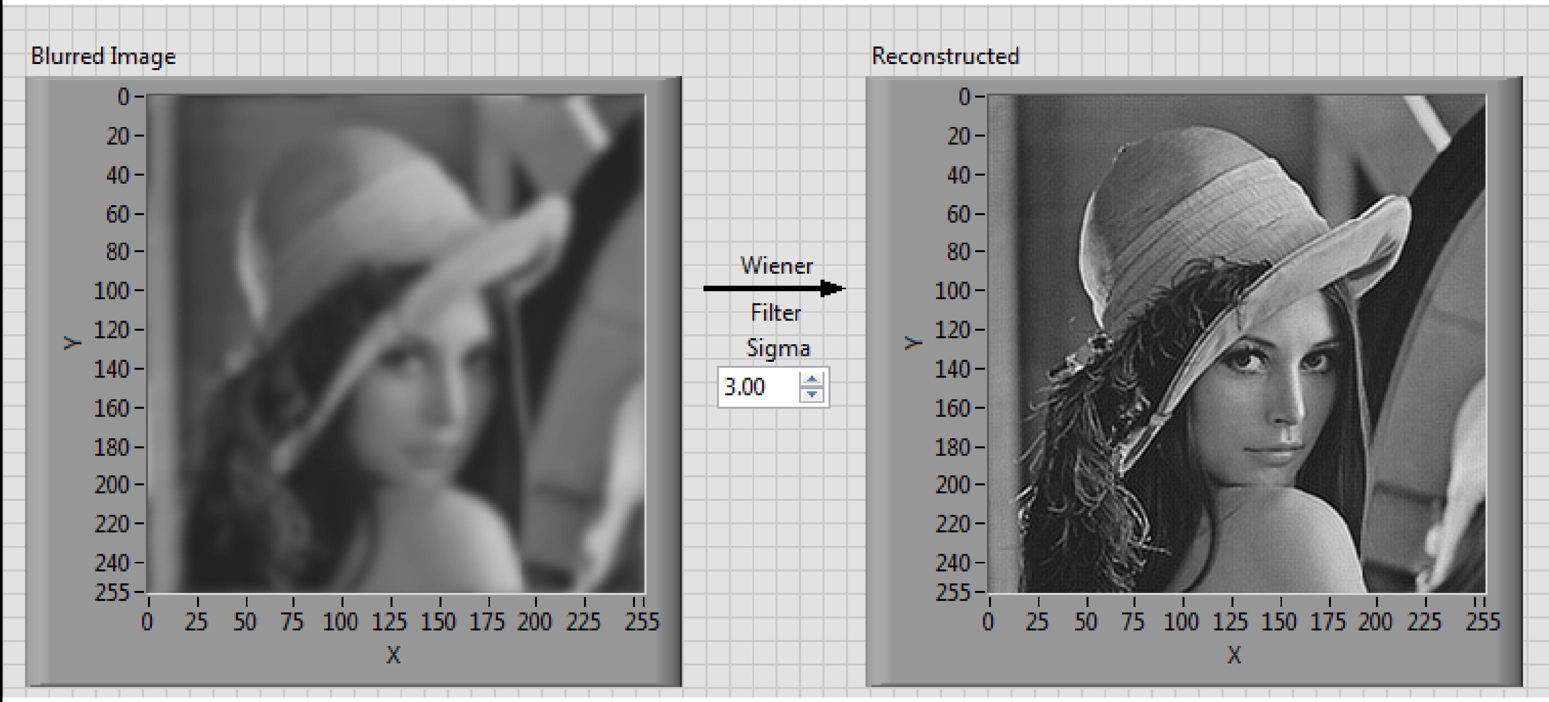

Figure 2: Wiener Filter

CODE:

Histogram Equalization

```
% Read the original image
originalImage = imread("images.tiff");

% Perform histogram equalization
equalizedImage = histeq(originalImage);

% Display the original image and its histogram
subplot(2,2,1), imshow(originalImage);
title('Original Image');
subplot(2,2,2), imhist(originalImage);
title('Original Histogram');

% Display the equalized image and its histogram
subplot(2,2,3), imshow(equalizedImage);
title('Equalised Image');
subplot(2,2,4), imhist(equalizedImage);
title('Equalised Histogram');

% Open a separate figure to allow interactive exploration of the original image
figure(2)
imageViewer(originalImage)
```

Weiner Filter

```
% Read the original image
originalImage = imread('images.tiff');
imshow(originalImage)
title('Original Image')

% Define the point spread function (PSF) for motion blur
PSF = fspecial('motion', 20, 90);

% Apply motion blur to the original image
blurredImage = imfilter(originalImage, PSF, 'conv', 'circular');

% Perform deconvolution using Wiener filter
restoredImage = deconvwnr(blurredImage, PSF);

% Display the blurred image
subplot(2, 2, 1)
imshow(blurredImage)
title('Blurred Image')

% Display the restored image after applying Wiener filter
subplot(2, 2, 2)
imshow(restoredImage)
title('Restored Blurred Image')
```

OUTPUT:

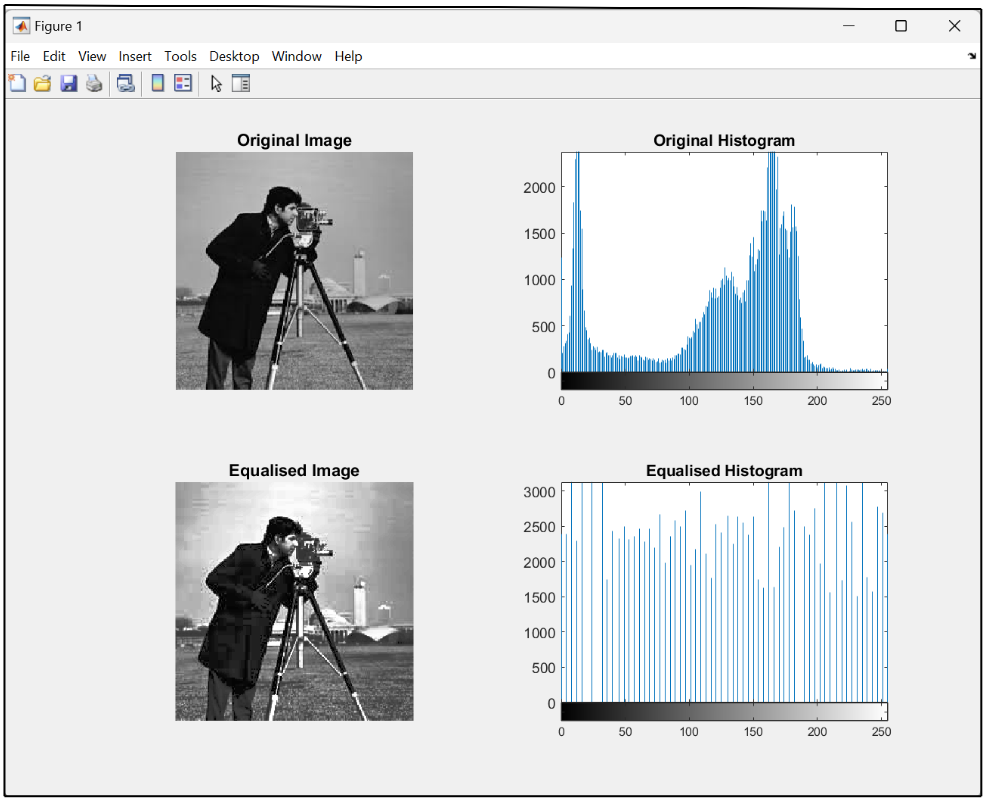

Figure 3 : MATLAB Output of Histogram Equalization Code

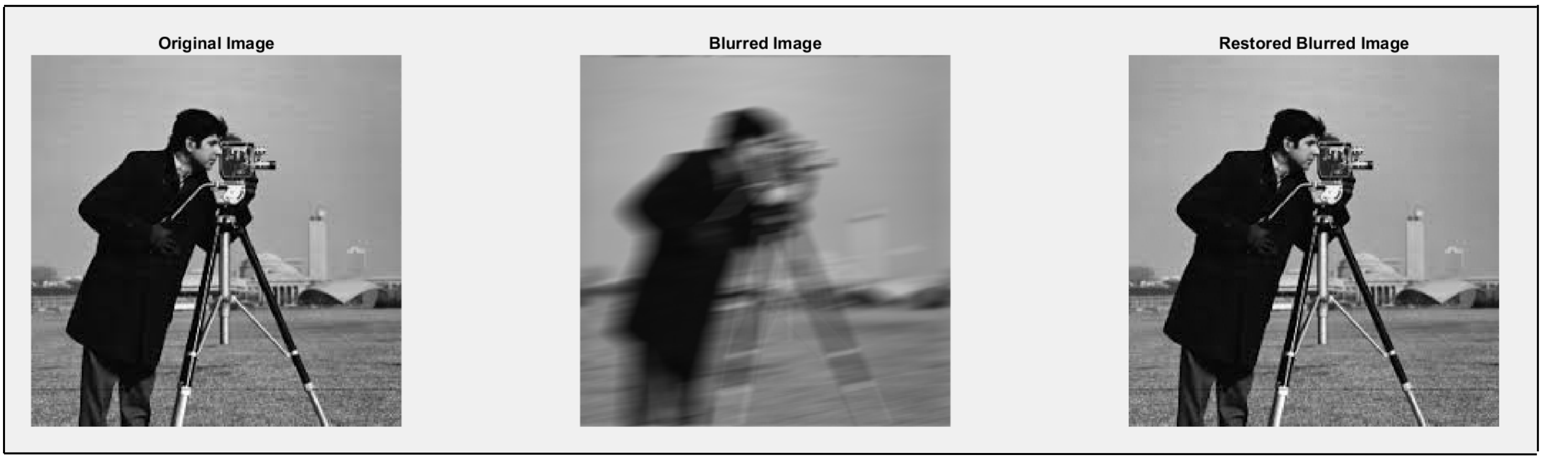

Figure 4 : MATLAB Output of Wiener Filter Code

CONCLUSION:

MATLAB's Histogram Equalization (***histeq*** & ***imhist***) offers a convenient way to:

- Enhance image contrast: Improves visual impact of low-contrast images.
- Visualize the process: See how contrast is boosted through histograms.

Remember:

- Noise amplification: Existing noise might be amplified.
- Customizable contrast: Specify a target histogram for specific needs.

Overall, MATLAB's histeq makes histogram equalization accessible for effective image contrast enhancement.

MATLAB's **Wiener filtering** functions (***deconvwnr*** & ***wiener2***) effectively:

- Deblur images: Removes blur and restores details.
- Reduce noise: Suppresses noise, leading to cleaner images.

However, consider:

- Noise estimation: Accuracy is crucial for optimal performance.
- Ringing artefacts: Might appear around sharp edges.
- Limitations: May not fully recover severely degraded images.

Overall, Wiener filtering in MATLAB is a powerful tool for image restoration, but understanding its nuances is key.