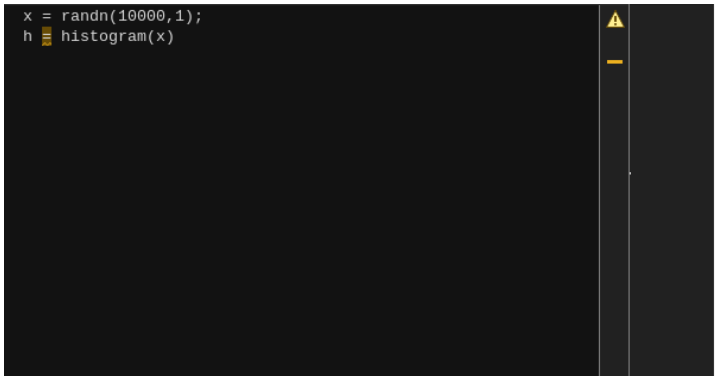
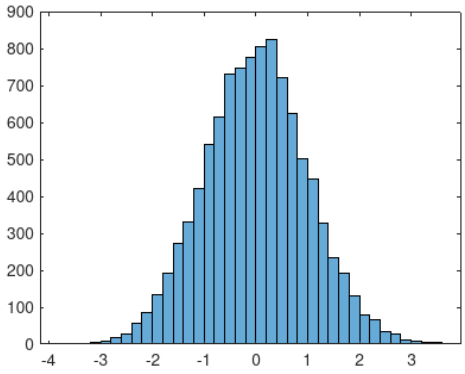
Lab 4: Histogram Equalization

Experiment No.8 Generate 10,000 Random Numbers and Create a Histogram.

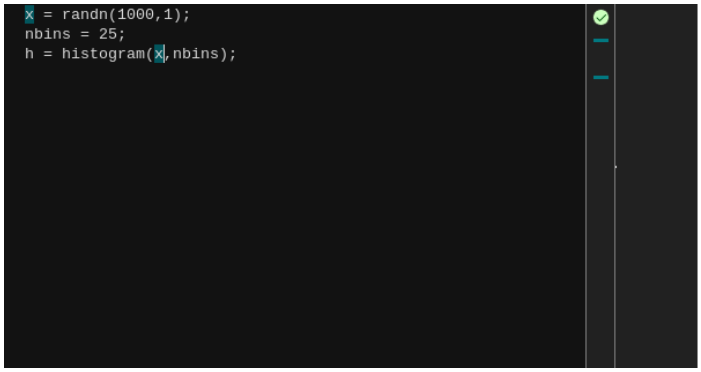
x = randn(10000,1);
h = histogram(x)

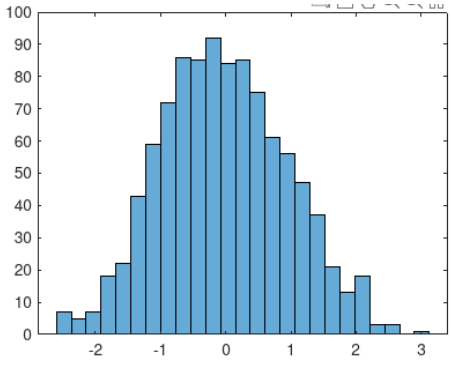




Experiment No.9 Plot a Histogram of 1,000 Random Numbers Sorted Into 25 Equally Spaced Bins.

x = randn(1000,1);
nbins = 25;
h = histogram(x,nbins);

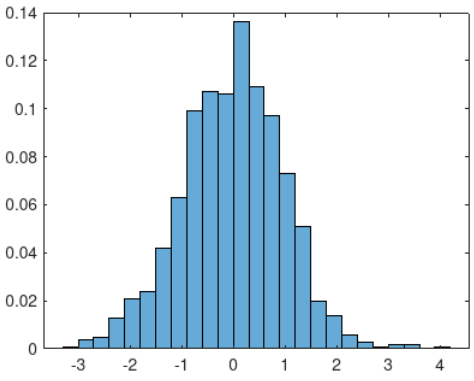




Experiment No.10 Generate 1,000 Random Numbers and Create a Histogram Using the 'Probability' Normalization.

x = randn(1000,1);
h = histogram(x,'Normalization','probability')

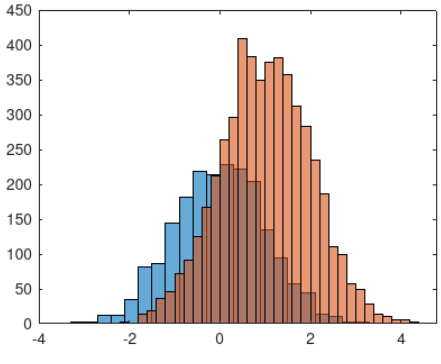
```
x = randn(1000,1);
h = histogram(x,'Normalization','probability')
```



Experiment No.11 Generate Two Vectors of Random Numbers and Plot a Histogram for Each Vector in the Same Figure.

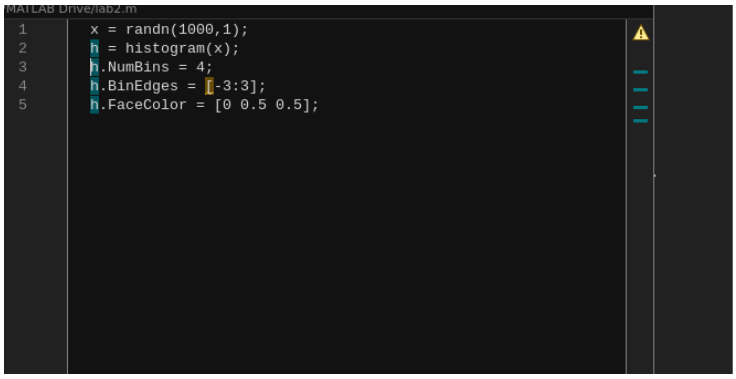
x = randn(2000,1);
y = 1 + randn(5000,1);
h1 = histogram(x);
hold on
h2 = histogram(y);

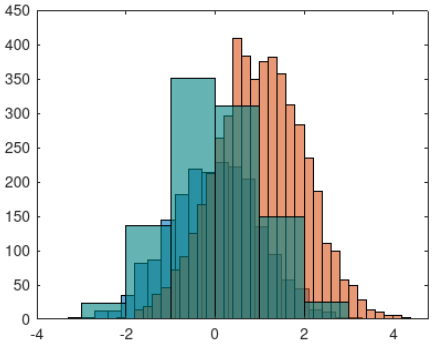
```
1  x = randn(2000,1);
2  y = 1 + randn(5000,1);
3  h1 = histogram(x);
4  hold on
5  h2 = histogram(y);
```



Experiment No.12 Generate 1,000 Random Numbers and Create a Histogram. Return the Histogram Object to Adjust the Properties of the Histogram Without Recreating the Entire Plot.

```
x = randn(1000,1);
h = histogram(x);
h.NumBins = 4;
h.BinEdges = [-3:3];
h.FaceColor = [0 0.5 0.5]; %bin color rgb
h.EdgeColor = 'r'; % edge color r = red
```





Experiment No.13 Draw Histogram from an image.

```
img = imread('at.jpg');
histogram(img);
```

Experiment No.14 Perform Histogram Equalization to Enhance an Image

Aim:

To enhance contrast using Histogram Equalization.

Syntax:

```
J = histeq(I, hgram)
J = histeq(I, n)
[J, T] = histeq(I,...)
newmap = histeq(X, map, hgram)
newmap = histeq(X, map)
[newmap, T] = histeq(X,...)
```

Theory

- histeq enhances the contrast of images by transforming the values in an intensity image, or the values in the colormap of an indexed image, so that the histogram of the output image approximately matches a specified histogram.
- J = histeq(I, hgram) transforms the intensity image I so that the histogram of the outputintensity image J with length(hgram) bins approximately matches hgram.
- **histeq** automatically scales hgram so that sum(hgram) = prod(size(I)). The histogram of J will better match hgram when length(hgram) is much smaller than the number of discrete levels in I.
- J = histeq(I, n) transforms the intensity image I, returning in J an intensity image with n discrete gray levels. A roughly equal number of pixels is mapped to each of the n levels in J, so that the histogram of J is approximately flat. (The histogram of J is flatter when n is much smaller than the number of discrete levels in I.) The default value for n is 64.
- [J, T] = histeq(I,...) returns the grayscale transformation that maps gray levels in the image I to gray levels in J.

- newmap = histeq(X, map, hgram) transforms the colormap associated with the indexed image X so that the histogram of the gray component of the indexed image (X,newmap) approximately matches hgram. The histeq function returns the transformed colormap in newmap. length(hgram) must be the same as size(map,1).
- newmap = histeq(X, map) transforms the values in the colormap so that the histogram of the gray component of the indexed image X is approximately flat. It returns the transformed colormap in newmap.
- [newmap, T] = histeq(X,...) returns the grayscale transformation T that maps the gray component of map to the gray component of newmap.

Examples

1. Enhance the contrast of an intensity image using histogram

equalization.I = imread('tire.tif');

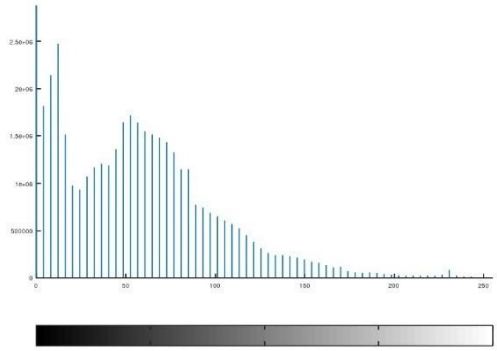
J = histeq(I);

imshow(I)

figure, imshow(J)

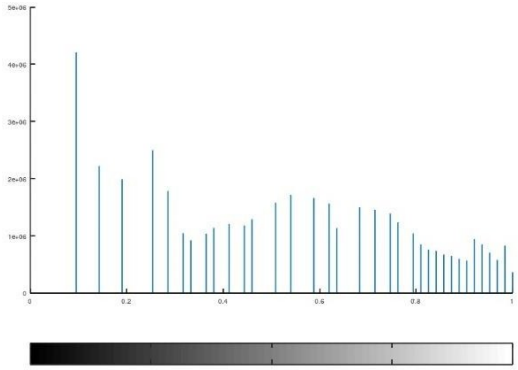
2. Display a histogram of the original image.

figure; imhist(I,64)



3. Compare it to a histogram of the processed

image.figure; imhist(J,64)



Algorithm

When you supply a desired histogram hgram, histeq chooses the grayscale transformation T to minimize where c0 is the cumulative histogram of A, c1 is the cumulative sum of hgram for all intensities k. This minimization is subject to the constraints that T must be monotonic and c1(T(a))cannot overshoot c0(a) by more than half the distance between the histogram counts at a. histeq uses the transformation b = T(a) to map the gray levels in X (or the colormap) to their new values.Ifyou do not specify hgram, histeq creates a flat hgram,

hgram = ones(1,n)*prod(size(A))/n;

Task:

- Read an image.
- Apply histogram equalization.
- Display original image, enhanced image, and their histograms in the same figure.