Digital Signal Processing

Lab7: Fourier Transform and image signal
Instructor: Eng\ Samar Shaaban
E-mail: ssa10@fayoum.edu.eg
Github Repo: https://github.com/SamarShabanCS/DSP
Slack workspace: https://fayoum-university-fci.slack.com

Z-transform

```
>> syms xx x y n
>> xx=cos(n)

xx =

cos(n)

>> x=ztrans(xx)

x =

(z*(z - cos(1)))/(z^2 - 2*cos(1)*z + 1)

>> y=iztrans(x)

y =

cos(n)
```

Z-transform

inverse Z-transform

Example 4:-

Determine the z-transform of the following signals.

$$x[n] = [1,\ 2,\ 5,\ 7,\ 0,\ 1]$$

$$\boxed{X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n}}$$

Ans:

```matlab
%% z-transform
clc; close all; clear;
x=[9 5 4 2 1];
b=0;
n=length(x);
y=sym('z');
for i=1:n
    b=b+x(i)*y^(1-i);
end
disp(b);
```

x[n] = δ[n] + 2δ[n-1] + 5δ[n-2] + 7δ[n-3] + δ[n-5]

$$X(z) = 1 + 2z^{-1} + 5z^{-2} + 7z^{-3} + z^{-5}$$

Fourier Transform

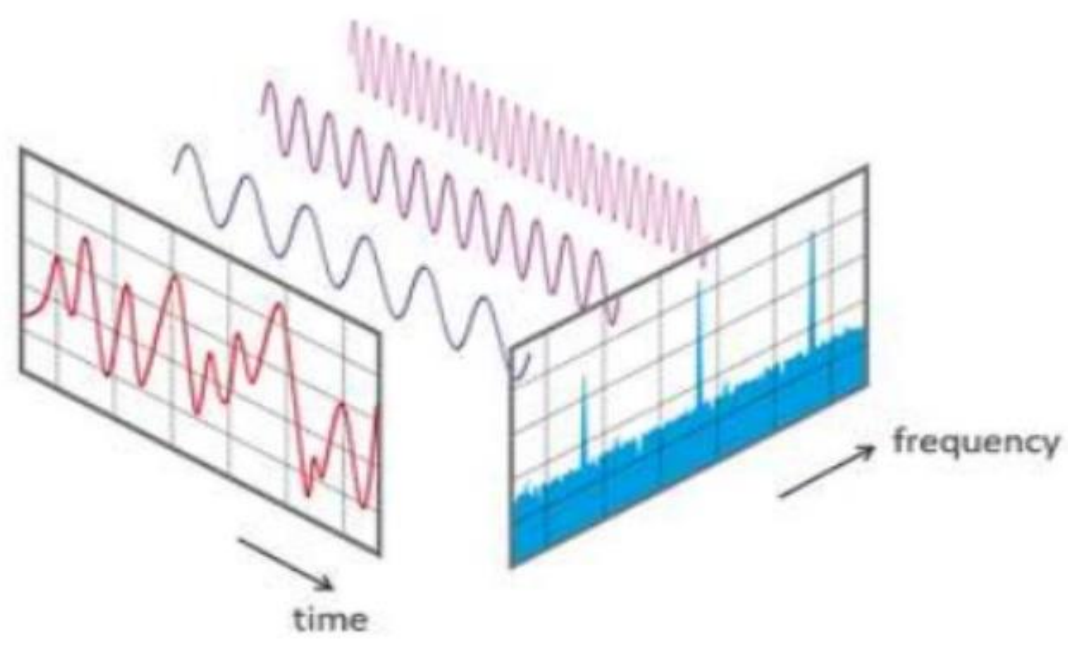
Get FFT and plot absolute spectrum

What is Fourier Transform?

- Fourier Transform is a tool to transforming a wave function or signal from a time domain(how a signal changes over time) into frequency domain (how much of the signal lies within each given frequency band over a range of frequencies)

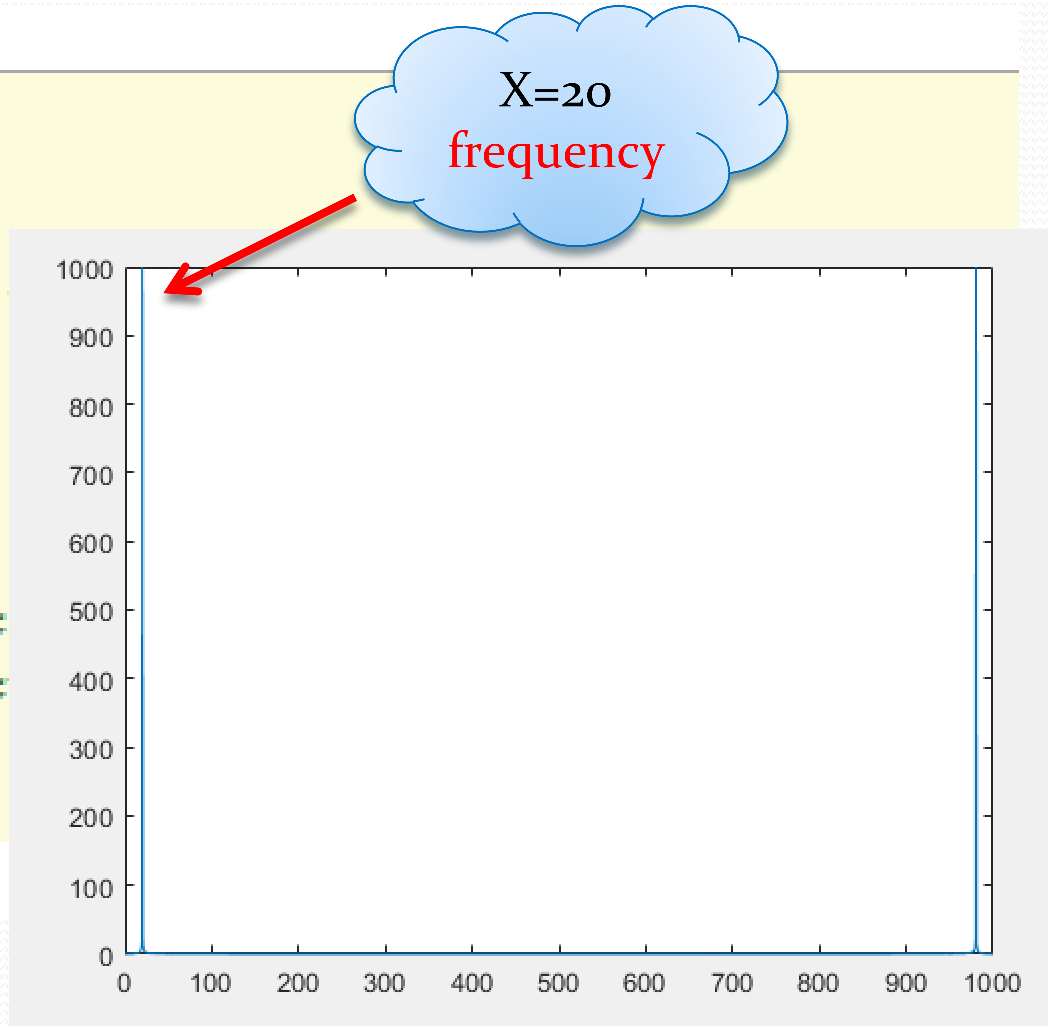
- ***Why?***

- To analyze signal easier
- To separate compound signal
- Using FFT() function



Example

```
close all; clear; clc;
%% simple fourier transform
fs=1000;
time=2;
t=0:1/fs:time;  %discretize
x=sin(2*pi*20*t);
plot(t,x);

N=fs*time;
f=linspace(0,fs,N+1);  %disc
ff=abs(fft(x));        %calc
plot(f,ff);
```



It is your turn

- Construct a complex signal compound of three sin signals with frequencies [10,50,90], respectively.
- Plot each signal(plot the 4 signal using subplot).
- Apply Fast Fourier Transform to separate the complex signal into its primitive signals.
- Plot the output from Fast Fourier Transform.

Sol:

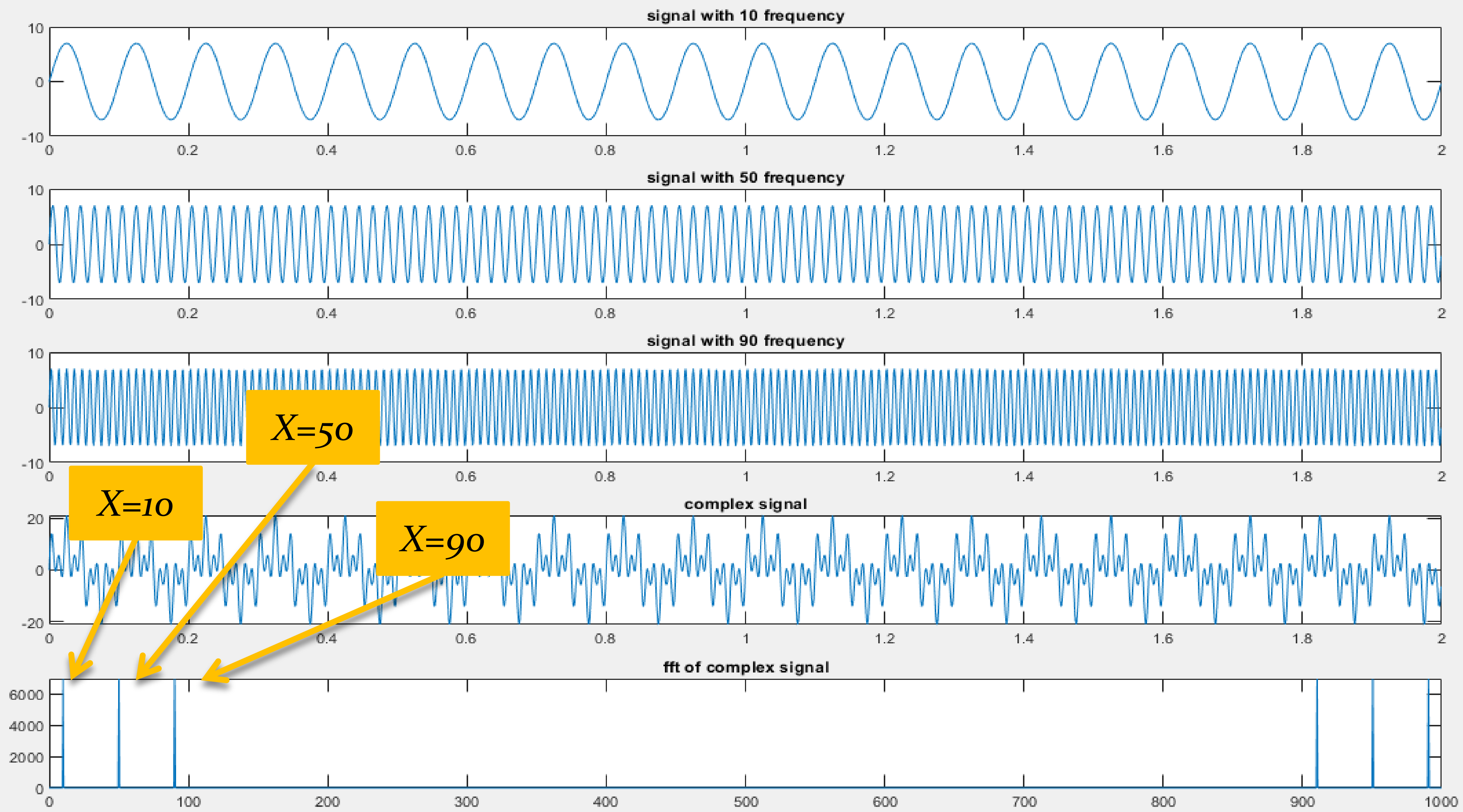
```
close all; clear; clc;
%% complex signal consist of [10,50,90]frequencies
fs=1000;
time=2;
ts=1/fs;
N=fs*time;
t=0:ts:time-ts;  %discretize time
freqs=[10 50 90];
Amp=7;
x_complex=0;
for i=1:length(freqs)
    x(i,:)=Amp*sin(2*pi*freqs(i)*t);
    x_complex=x_complex+x(i,:);
end
```

```
%apply FFT

f=linspace(0,fs,N);
fft_complex_1=fft(x_complex);
fft_complex_2=abs(fft_complex_1);
```

```
%plotting signals
subplot(5,1,1)
plot(t,x(1,:));
title('signal with 10 frequency');

subplot(5,1,2)
plot(t,x(2,:));
title('signal with 50 frequency');

subplot(5,1,3)
plot(t,x(3,:));
title('signal with 90 frequency');

subplot(5,1,4)
plot(t,x_complex);
title('complex signal');
```

```
%plot fft of complex_signal
subplot(5,1,5)
plot(f,fft_complex_2);
title('fft of complex signal');
```

Sol: Plotting the signals



2D signal: Image

What is an image?

- An image is a 2D function *f(x,y), where x and y are* ***spatial coordinates*** and the magnitude of *f at any point is called* the **intensity** of the image at that point

- When *x, y and the intensity are discrete quantities we call the* image a **digital image**

- The elements of a digital image are referred to as **pixels**

Reading Images in MATLAB

```
x = imread ('c:\lab.jpg');
```

Image Write

```
imwrite (x,'c:\lab.jpg');
```

MATLAB Image Types

- Indexed Images
- Grayscale Images
- Binary Images
- RGB Images

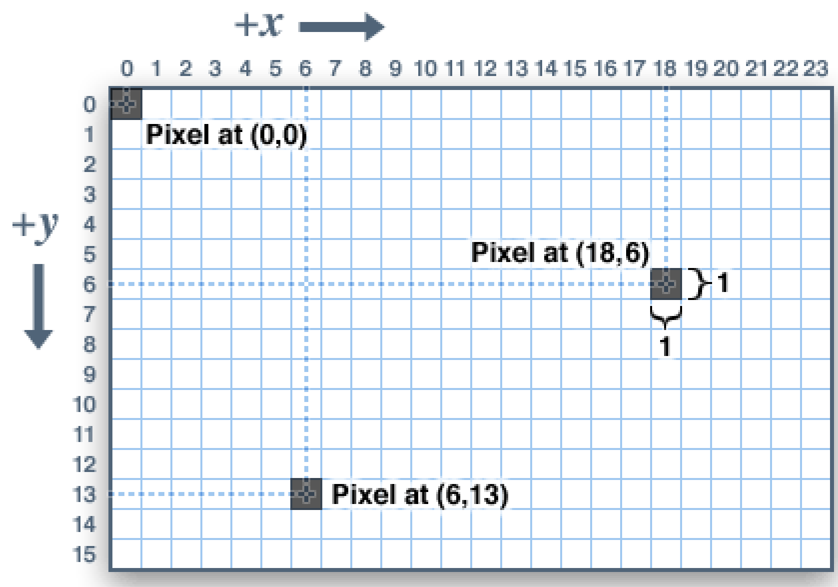
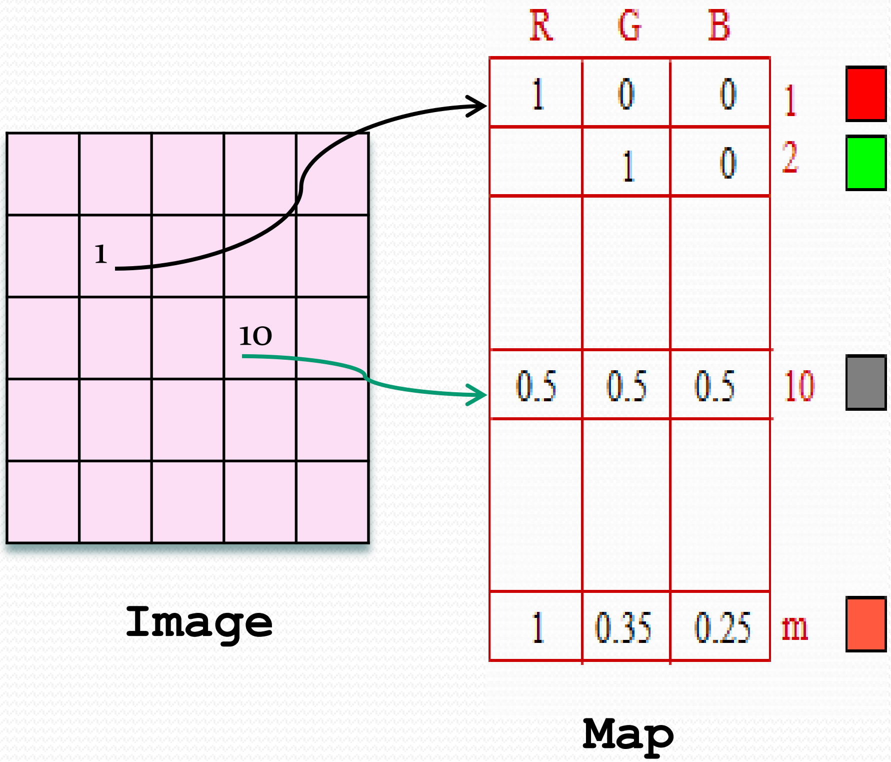


Image data in MATLAB can be logical, double, uint8, uint16

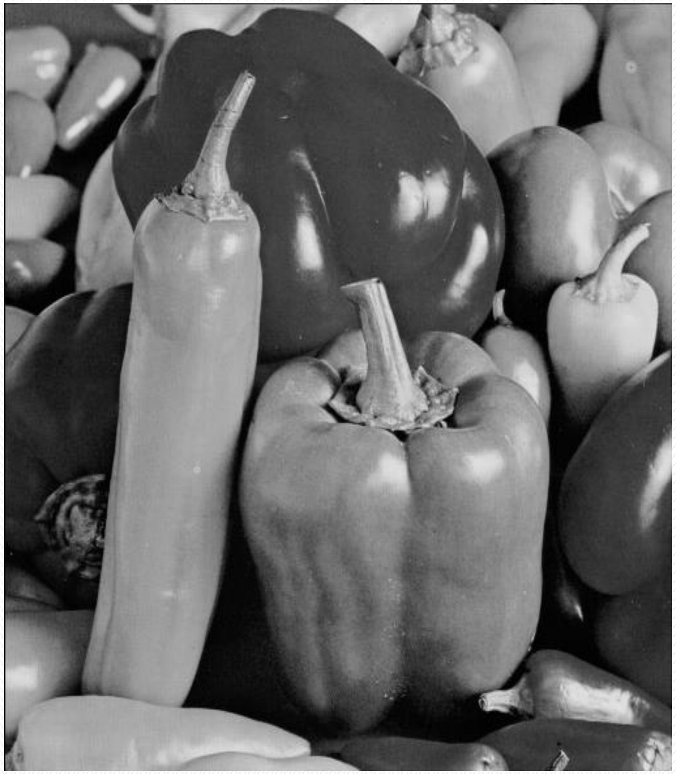
Indexed Images

- Consists of a data matrix, **I** and a colormap matrix, **C**
  - C is an m-by-3 matrix, with each row specifying the R, G, and B components of a single color.
  - Values in C are floating point numbers in the range [0, 1]
  - Color of each pixel is determined by using the corresponding value of I as an index into the colormap



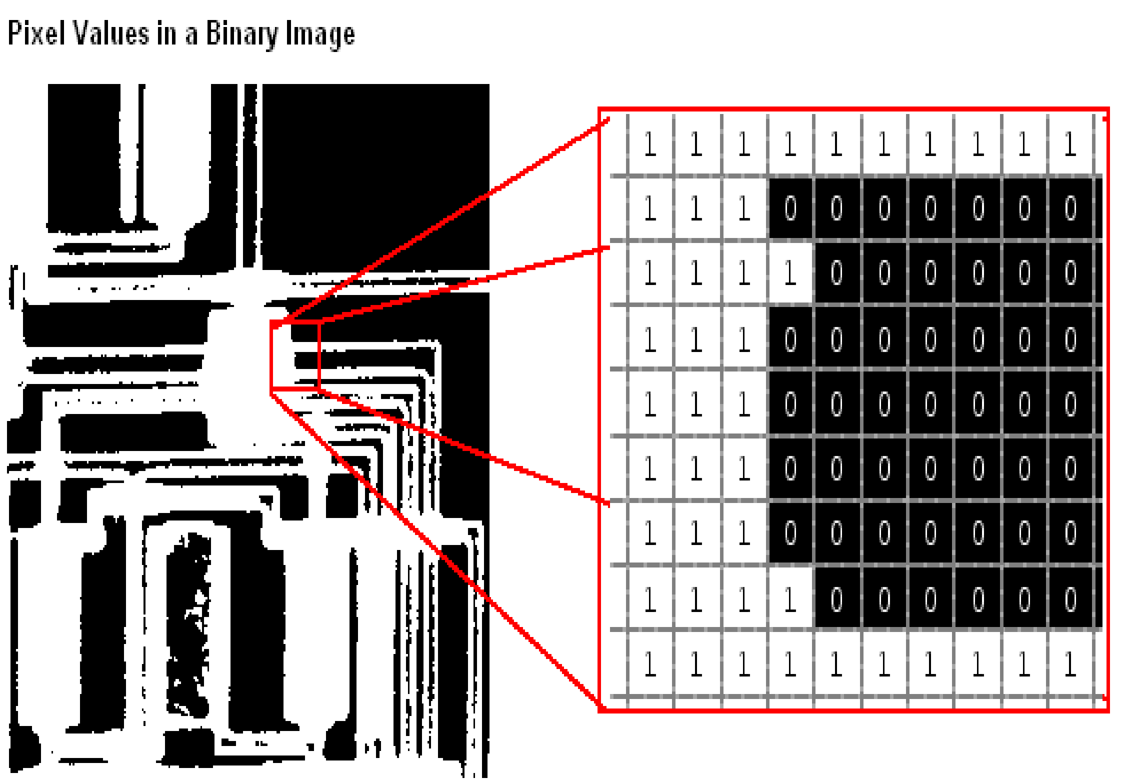
Grayscale Images

- Each pixel is usually stored as a byte (value between 0 to 255)



Binary Images

- In a binary image, each pixel assumes one of only two discrete values:1 or 0.



RGB Images

- Consist of a m-by-n-by-3 data array, I, containing the R, G, and B components for each individual pixel
  - I(:, :, 1) is the red component of the image
  - I(:, :, 2) is the green component of the image
  - I(:, :, 3) is the blue component of the image
- To display a true color image, do the following

  >> imshow(I)

Displaying Image

- To display a true color image: **>> imshow(I)**
- If another image is to be displayed using imshow MATLAB replaces the image in the screen with the new image.
- To keep the first image and output a second we use function figure to display both images.

figure, imshow(x); title('Image');

Displaying Image

- Displaying multiple images at the same figure:

- >> subplot(2,1,1)
- >> imshow(x)
- >> title('Image1')
- >> subplot(2,1,2)
- >> imshow(y)
- >> title('Image2')