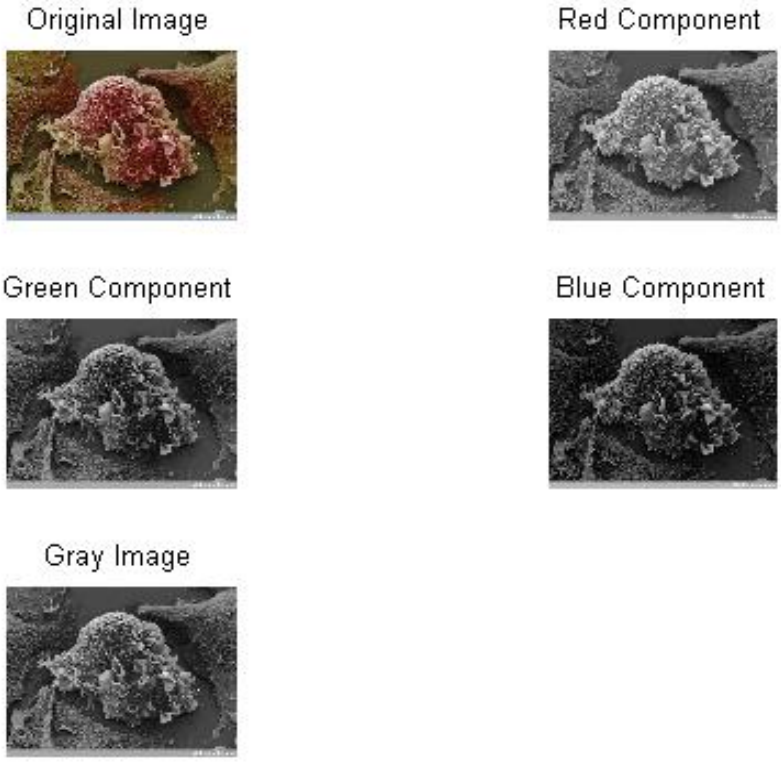
DIGITAL IMAGE PROCESSING LAB MANUAL

1. Simulation and Display of an Image, Negative of an Image(Binary & Gray Scale)
2. Implementation of Relationships between Pixels
3. Implementation of Transformations of an Image
4. Contrast stretching of a low contrast image, Histogram, and Histogram Equalization
5. Display of bit planes of an Image
6. Display of FFT(1-D & 2-D) of an image
7. Computation of Mean, Standard Deviation, Correlation coefficient of the given Image
8. Implementation of Image Smoothening Filters(Mean and Median filtering of an Image)
9. Implementation of image sharpening filters and Edge Detection using Gradient Filters
10. Image Compression by DCT,DPCM, HUFFMAN coding
11. Implementation of image restoring techniques
12. Implementation of Image Intensity slicing technique for image enhancement
13. Canny edge detection Algorithm

Prepared by Abhishek Appaji

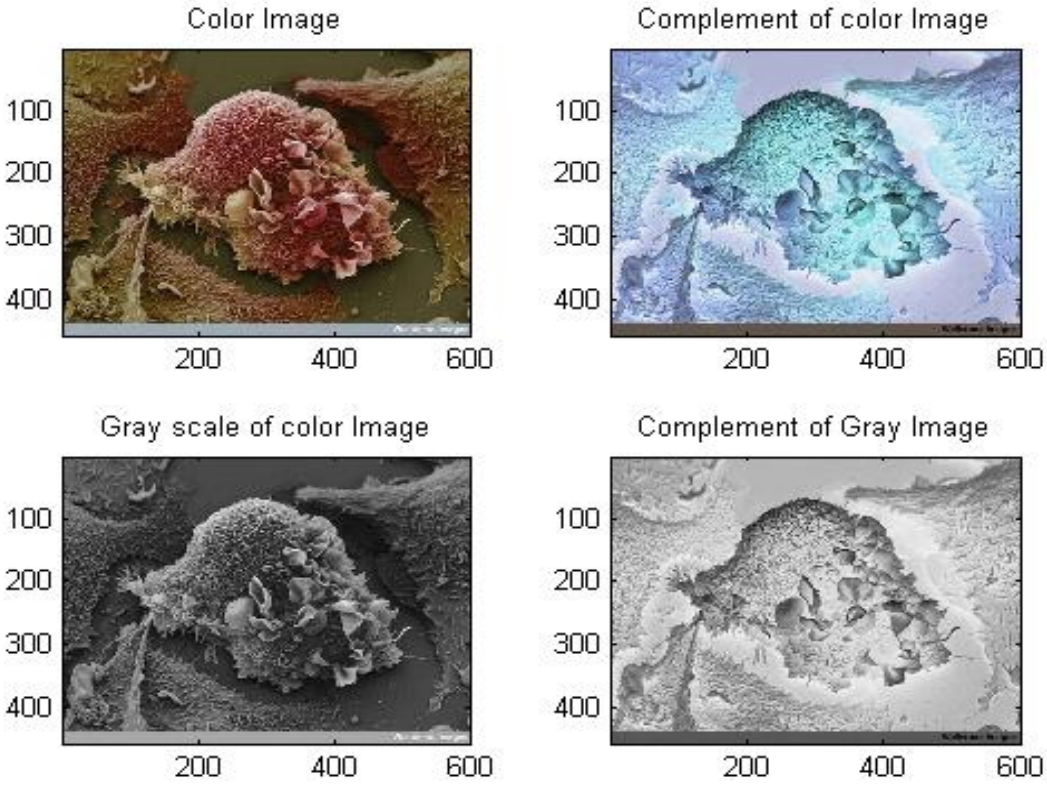
1. Simulation and Display of an Image, Negative of an Image(Binary & Gray Scale)

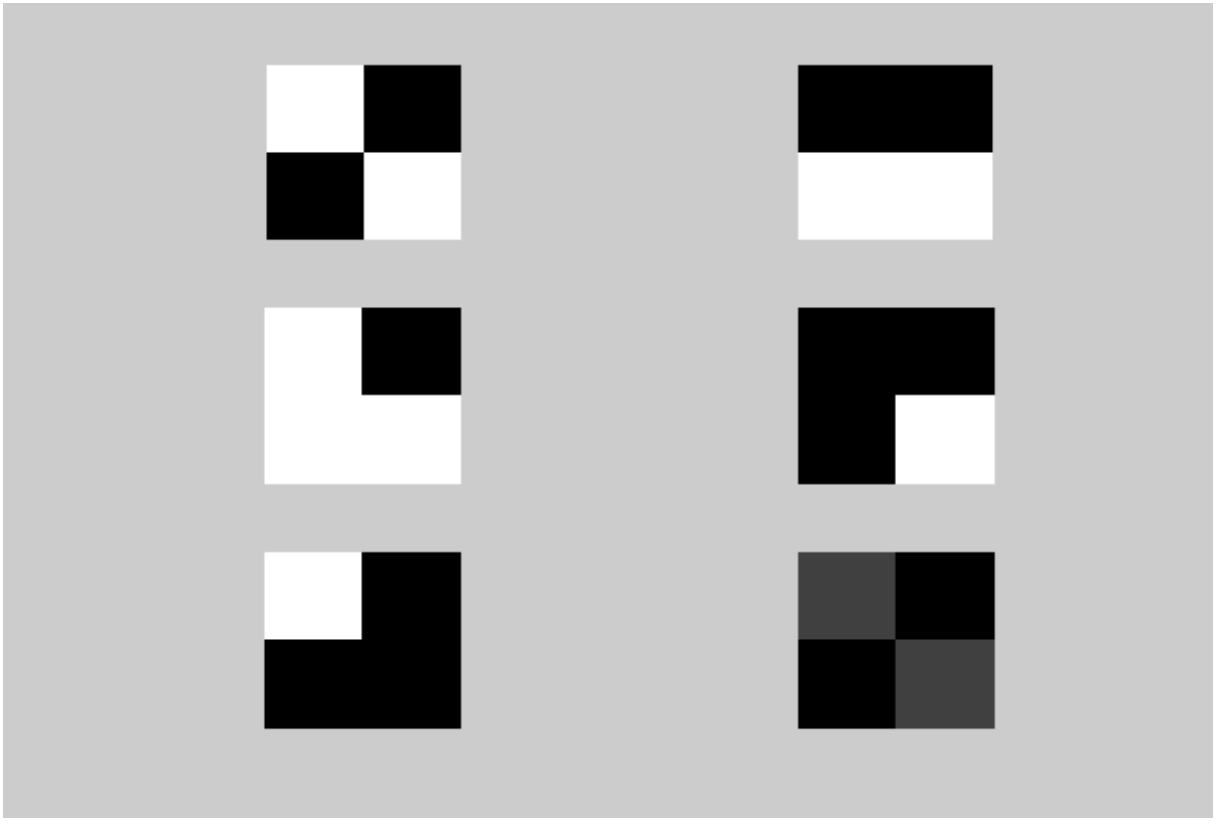
```
% Red Blue and Green and Gray Components
i=imread('cancercell.jpg');
subplot(3,2,1); imshow(i); title('Original Image');

%Red Component
r=i(:,:,1);
subplot(3,2,2); imshow(r);title('Red Component');

%Green Component
g=i(:,:,2);
subplot(3,2,3); imshow(g); title('Green Component');

%Blue Component
b=i(:,:,3);
subplot(3,2,4); imshow(b); title('Blue Component');

%Color to Gray Image
rg=rgb2gray(i);
subplot(3,2,5); imshow(rg); title('Gray Image');
```



Complement, Converting and Simulation of an Image

```
% Display color Image, find its complement and convert to gray scale
I=imread('cancercell.jpg');
subplot(2,2,1); imshow(I); subimage(I); title('Color Image');

c=imcomplement(I);
subplot(2,2,2); imshow(c); subimage(c); title('Complement of color Image');

r=rgb2gray(I);
subplot(2,2,3); imshow(r); subimage(r); title('Gray scale of color Image');

%Complement of Gray Image
b=imcomplement(r);
subplot(2,2,4); imshow(b); subimage(b); title('Complement of Gray Image');

%Simulation of an Image( Arithmetic & Logic Operation)
a=ones(40); b=zeros(40);
c=[a b;b a]; d=[b b;a a];
A=10*(c+d);
M=c.*d;
S=c-d;
D=c/4;
figure;
subplot(3,2,1);  imshow(c);
subplot(3,2,2);  imshow(d);
subplot(3,2,3);  imshow(A);
subplot(3,2,4);  imshow(M);
subplot(3,2,5);  imshow(S);
subplot(3,2,6); imshow(D);
```


Color Image
Complement of color Image
Gray scale of color Image
Complement of Gray Image
100
200
300
400
200
400
600



2. Implementation of Relationships between Pixels

Neighbour of 4,8 and Diagonal point

```
% To find Neighbour of a given Pixel

a=magic(5);
disp('a='); disp(a);
b=input('Enter the row < size of the Matrix');
c=input(' Enter the Column < size of matrix');
disp('Element'); disp(a(b,c));

% 4 Point Neighbour
N4=[a(b+1,c), a(b-1,c), a(b,c+1), a(b,c-1)];
disp('N4='); disp(N4);

%8 Point Neighbour
N8=[a(b+1,c), a(b-1,c), a(b,c+1), a(b,c-1), a(b+1,c+1), a(b+1,c-1), a(b-1,c-1), a(b-1,c+1)];
disp('N8='); disp(N8);

%Diagonal Neighbour
ND=[ a(b+1,c+1), a(b+1,c-1), a(b-1,c-1), a(b-1,c+1)];
disp('ND='); disp(ND);
```

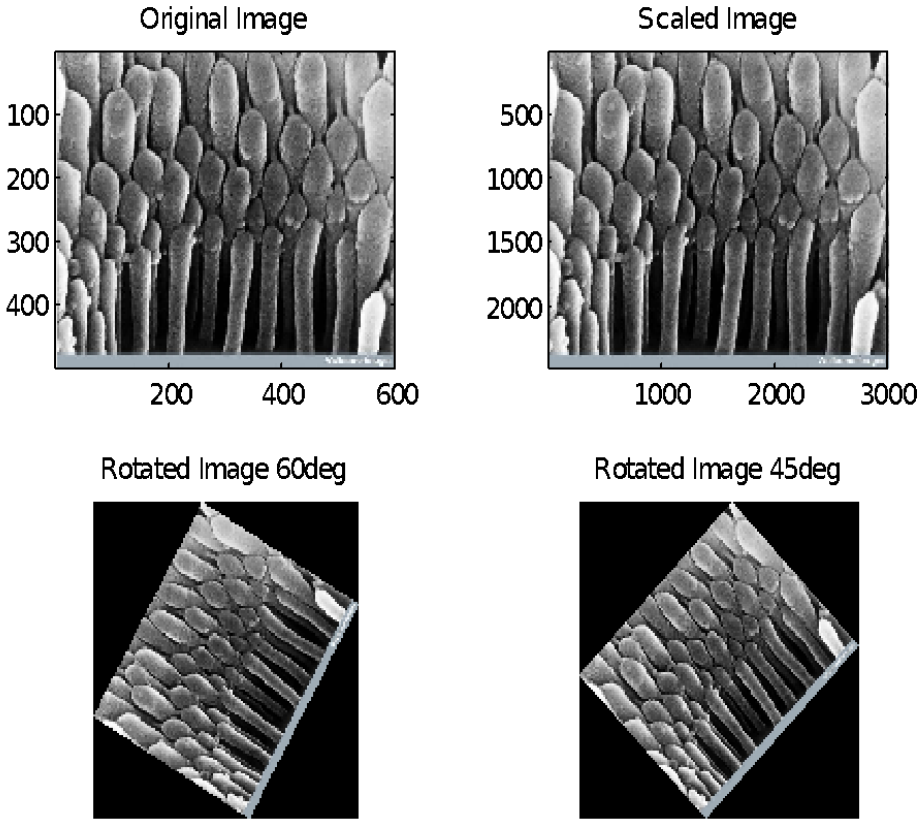
Output

```
a=
   17   24    1    8   15
   23    5    7   14   16
    4    6   13   20   22
   10   12   19   21    3
   11   18   25    2    9

Enter the row < size of the Matrix    3
 Enter the Column < size of matrix    3
  Element =
         13
N4=
   19    7   20    6
N8=
   19    7   20    6   21   12    5   14
ND=
   21   12    5   14
```

3. Implementation of Transformations of an Image

%Scaling & Rotation

```
% Scaling (Resize)
I=imread('earcell.jpg');
subplot(2,2,1); subimage(I); title('Original Image');

s=input('Enter Scaling Factor');
j=imresize(I,s);
subplot(2,2,2); subimage(j); title('Scaled Image');

% Rotation
K=imrotate(j,60);
subplot(2,2,3); imshow(K); title('Rotated Image 60deg');

R=imrotate(j,45);
subplot(2,2,4); imshow(R); title('Rotated Image 45deg');
```



%Display the color image and its Resized images by different methods

%Display the color image

```
I=imread('embryo.jpg');
figure,
subplot(2,2,1);
subimage(I);
title('Original Image');
```

%Display Resized image by Bilinear method

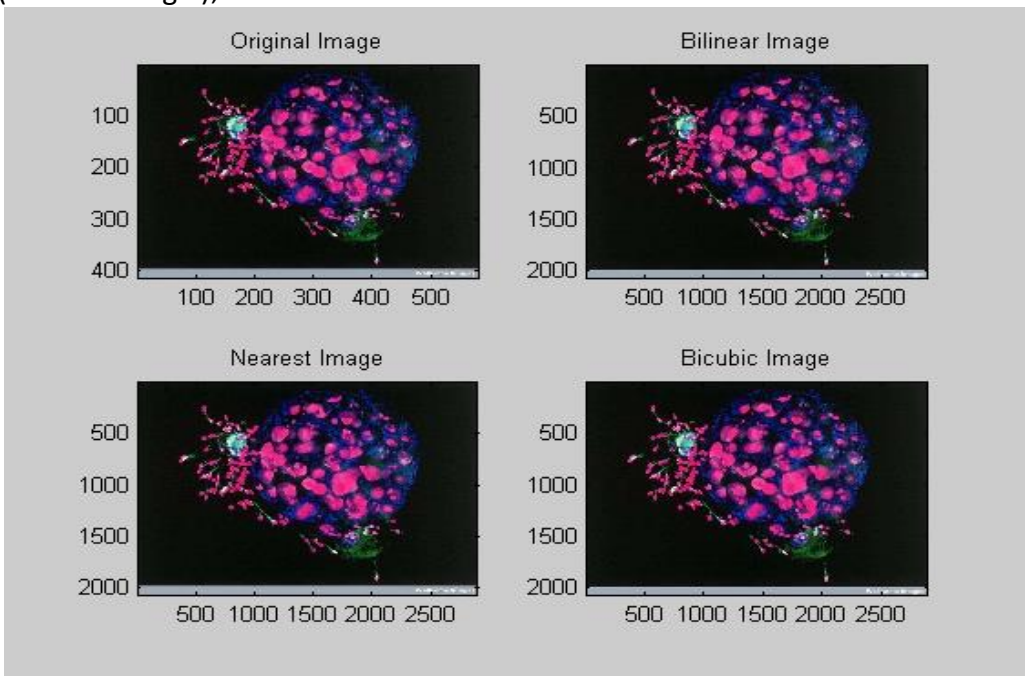
```
B=imresize(I,5);
subplot(2,2,2);
subimage(B);
title('Bilinear Image');
```

%Display Resized image by Nearest method

```
C=imresize(I,5,'nearest');
subplot(2,2,3);
subimage(C);
title('Nearest Image');
```

%Display Resized image by Bicubic method

```
D=imresize(I,5,'Bicubic');
subplot(2,2,4);
subimage(D);
title('Bicubic Image');
```



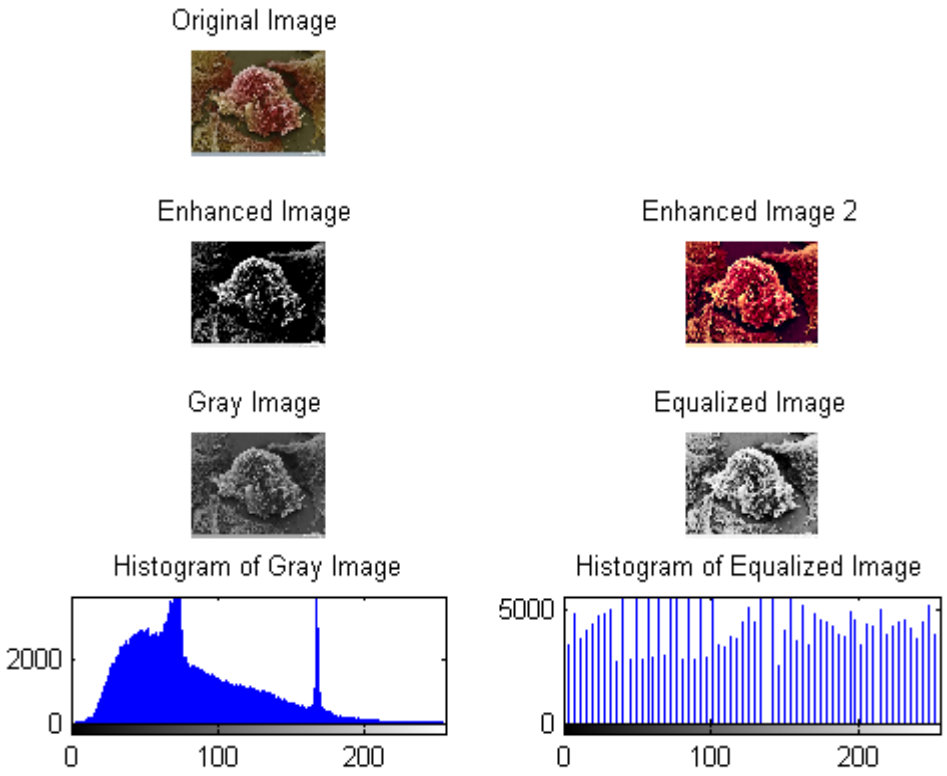
4. Contrast stretching of a low contrast image, Histogram, and Histogram Equalization

% Image Enhancement

```
I=imread('cancercell.jpg');
subplot(4,2,1); imshow(I); title('Original Image');

g=rgb2gray(I);
subplot(4,2,5); imshow(g); title('Gray Image');

J=imadjust(g,[0.3 0.7],[]);
subplot(4,2,3); imshow(J); title('Enhanced Image');

D= imadjust(I,[0.2 0.3 0; 0.6 0.7 1],[]);
subplot(4,2,4);imshow(D);title('Enhanced Image 2');
```

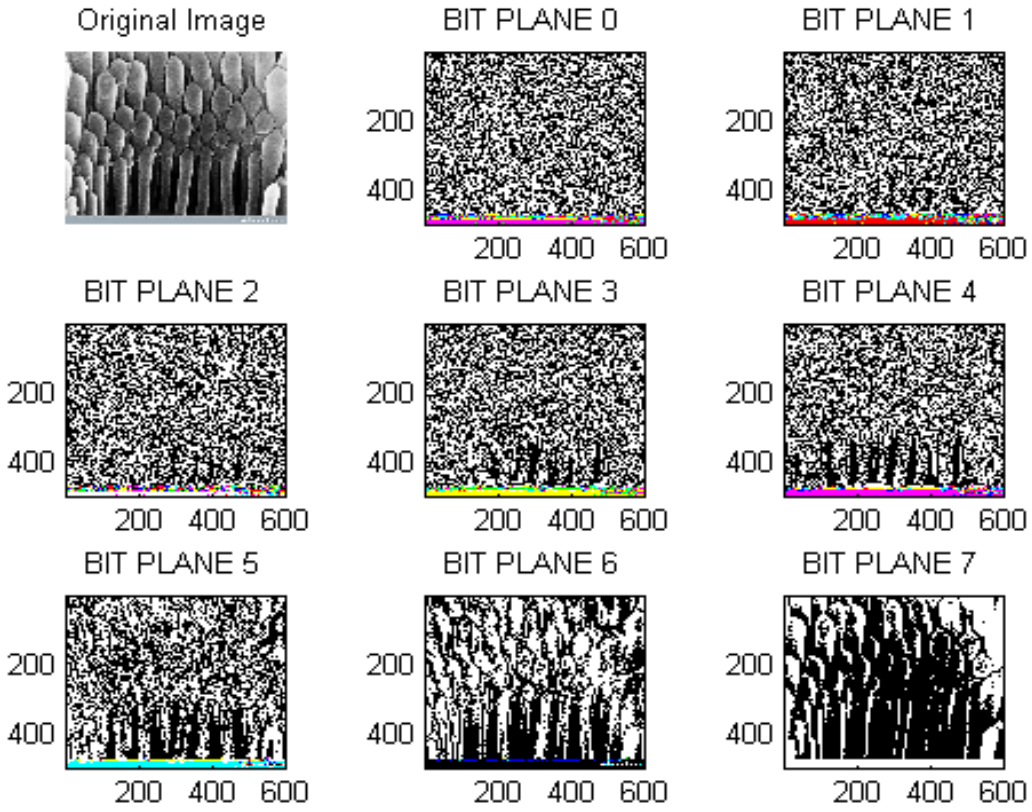
% Histogram and Histogram Equalization

```
subplot(4,2,7); imhist(g); title('Histogram of Gray Image');

m=histeq(g);
subplot(4,2,6); imshow(m); title('Equalized Image');
subplot(4,2,8); imhist(m); title('Histogram of Equalized Image');
```



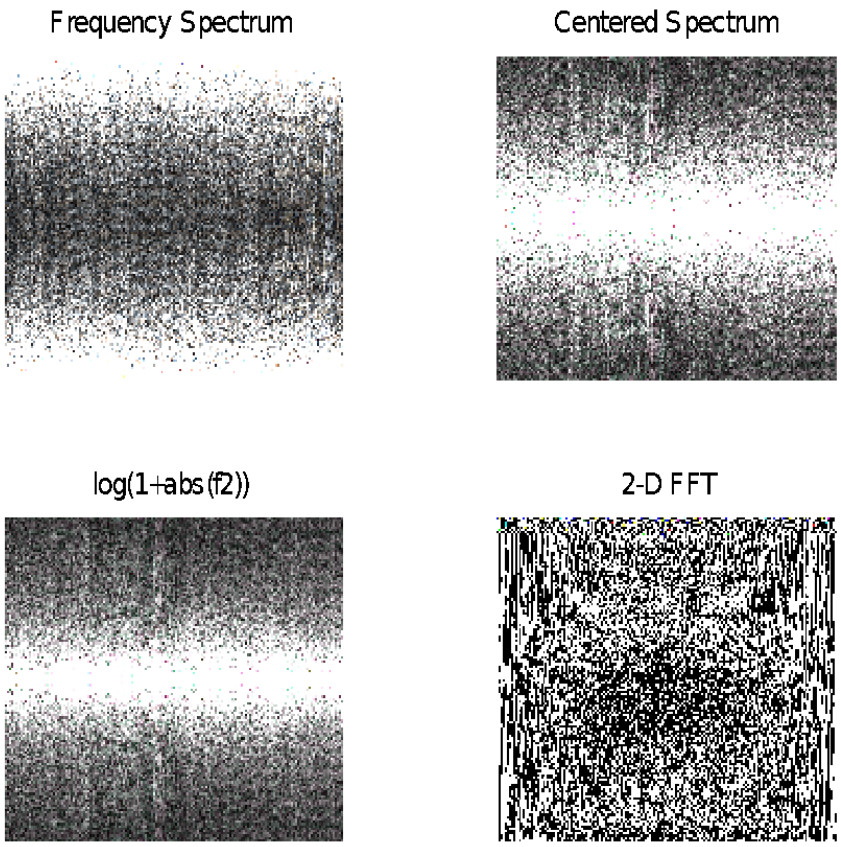
5. Display of bit planes of an Image

```
i=imread('earcell.jpg');
b0=double(bitget(i,1));
b1=double(bitget(i,2));
b2=double(bitget(i,3));
b3=double(bitget(i,4));
b4=double(bitget(i,5));
b5=double(bitget(i,6));
b6=double(bitget(i,7));
b7=double(bitget(i,8));

subplot(3,3,1);imshow(i);title('Original Image');
subplot(3,3,2);subimage(b0);title('BIT PLANE 0');
subplot(3,3,3);subimage(b1);title('BIT PLANE 1');
subplot(3,3,4);subimage(b2);title('BIT PLANE 2');
subplot(3,3,5);subimage(b3);title('BIT PLANE 3');
subplot(3,3,6);subimage(b4);title('BIT PLANE 4');
subplot(3,3,7);subimage(b5);title('BIT PLANE 5');
subplot(3,3,8);subimage(b6);title('BIT PLANE 6');
subplot(3,3,9);subimage(b7);title('BIT PLANE 7');
```



6. Display of FFT(1-D & 2-D) of an image

```
l=im2double(imread('cancercell.jpg'));
f1=fft(l);
f2=fftshift(f1);
subplot(2,2,1); imshow(abs(f1)); title('Frequency Spectrum');
subplot(2,2,2); imshow(abs(f2)); title('Centered Spectrum');

f3=log(1+abs(f2));
subplot(2,2,3); imshow(f3); title('log(1+abs(f2))');
l=fft2(f1);
l1=real(l);
subplot(2,2,4); imshow(l1);title(' 2-D FFT');
```

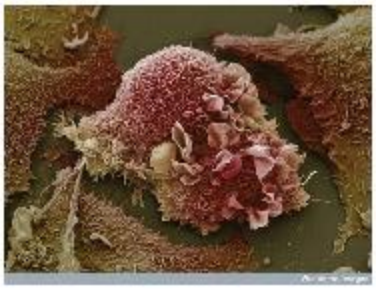


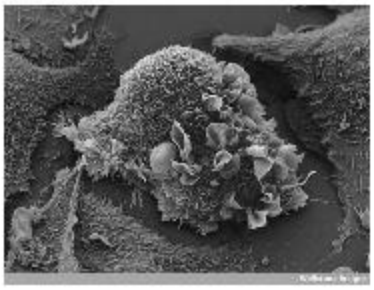
7. Computation of mean, Standard Deviation, Correlation coefficient of the given Image

```
 i=imread('cancercell.jpg');
subplot(2,2,1); imshow(i);title('Original Image');

g=rgb2gray(i);
subplot(2,2,2); imshow(g);title('Gray Image');

c=imcrop(g);
subplot(2,2,3); imshow(c);title('Cropped Image');

m=mean2(c);disp('m'); disp(m);
s=std2(c); disp('s'); disp(s);

figure,
k=(checkerboard>0.8);
subplot(2,1,1); imshow(k); title('Image1');

k1=(checkerboard>0.5);
subplot(2,1,2); imshow(k1); title('Image2');

r=corr2(k,k1);
disp('r');disp(r);
```

```
m
  74.5173

s
  44.2327

r
   0.5774
```

Original Image





Cropped Image

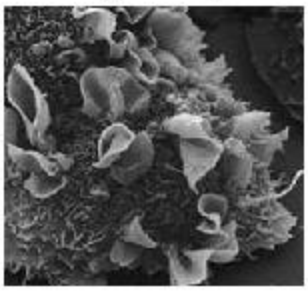


Image1

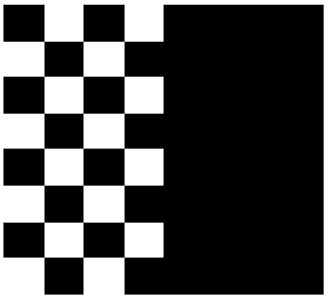
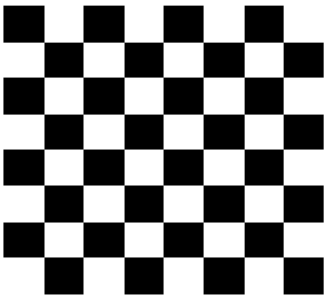


Image2



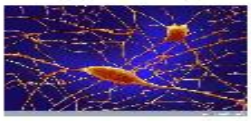
8. Implementation of Image Smoothening Filters(Mean and Median filtering of an Image)

```
% Median Filters
I=imread('nuron.jpg');
K = rgb2gray(I);
J= imnoise(K ,'salt & pepper',0.05);
f= medfilt2(J,[3,3]);
f1=medfilt2(J,[10,10]);

subplot(3,2,1); imshow(I); title('Original Image');
subplot(3,2,2); imshow(K); title('Gray Image');
subplot(3,2,3); imshow(J); title('Noise added Image');
subplot(3,2,4); imshow(f); title('3x3 Image');
subplot(3,2,5); imshow(f1); title('10x10 Image');
```

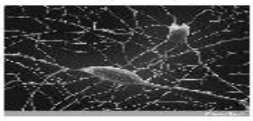
```
%Mean Filter and Average Filter
figure;
i=imread('nuron.jpg');
g=rgb2gray(i);
g1=fspecial('average',[3 3]);
b1 = imfilter(g,g1);
subplot(2,2,1); imshow(i); title('Original Image');
subplot(2,2,2); imshow(g); title('Gray Image');
subplot(2,2,3); imshow(b1); title('3x3 Image');
g2= fspecial('average',[10 10]);
b2=imfilter(g,g2);
subplot(2,2,4); imshow(b2); title('10x10 Image');
```

```
%Implementation of filter using Convolution
figure;
I= imread('earcell.jpg');
I=I(:,:,1); subplot(2,2,1); imshow(I); title('Original Image');

a=[0.001 0.001 0.001; 0.001 0.001 0.001; 0.001 0.001 0.001];
R=conv2(a,I);
subplot(2,2,2); imshow(R); title('Filtered Image');

b=[0.005 0.005 0.005; 0.005 0.005 0.005; 0.005 0.005 0.005];
R1=conv2(b,I);
subplot(2,2,3); imshow(R1); title('Filtered Image 2');
```

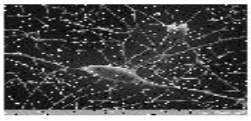
Original Image



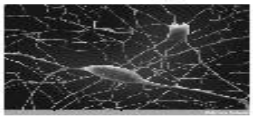
Gray Image



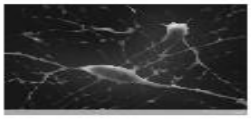
Noise added Image



3x3 Image

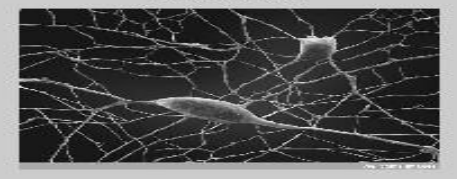


10x10 Image

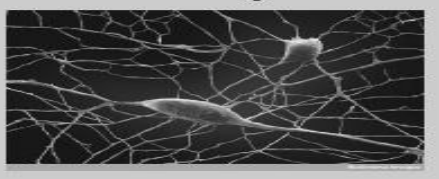


Original Image

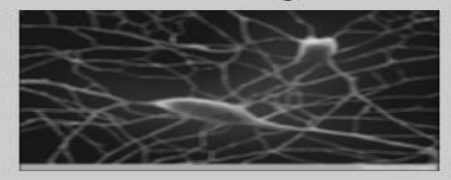
Gray Image



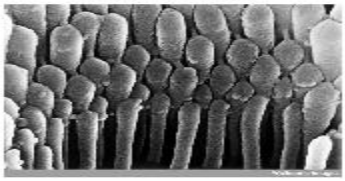
3x3 Image



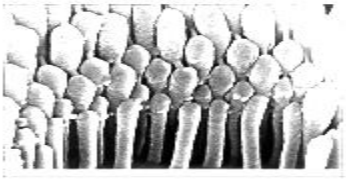
10x10 Image



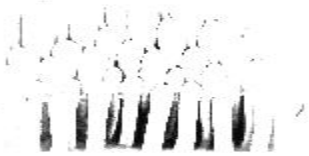
Original Image



Filtered Image



Filtered Image 2



9. Implementation of image sharpening filters and Edge Detection using Gradient Filters

```
i=imread('cancercell.jpg');
subplot(4,2,1); imshow(i);
title('Original Image');

g=rgb2gray(i);
subplot(4,2,2); imshow(g); title('Gray Image');

f=fspecial('laplacian',0.05);
im=imfilter(g,f);
subplot(4,2,3); imshow(im); title('Laplacian ');

s=edge(g, 'sobel');
subplot(4,2,4); imshow(s); title('Sobel');

p=edge(g, 'prewitt');
subplot(4,2,5); imshow(p); title('Prewitt');

r=edge(g, 'roberts');
subplot(4,2,6); imshow(r); title('Roberts');

[BW,thresh,gv,gh]=edge(g,'sobel',[],'horizontal');
[BW1,thresh1,gv1,gh1]=edge(g,'sobel',[],'vertical');

subplot(4,2,7); imshow(BW); title('Sobel Horizontal');
subplot(4,2,8);
imshow(BW); title('Sobel Vertical');
```

Original Image

Gray Image

Laplacian

Sobel

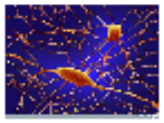
Prewitt

Roberts

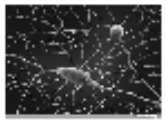
Sobel Horizontal

Sobel Vertical

Original Image



Gray Image



Laplacian

Sobel

Prewitt

Roberts

Sobel Horizontal

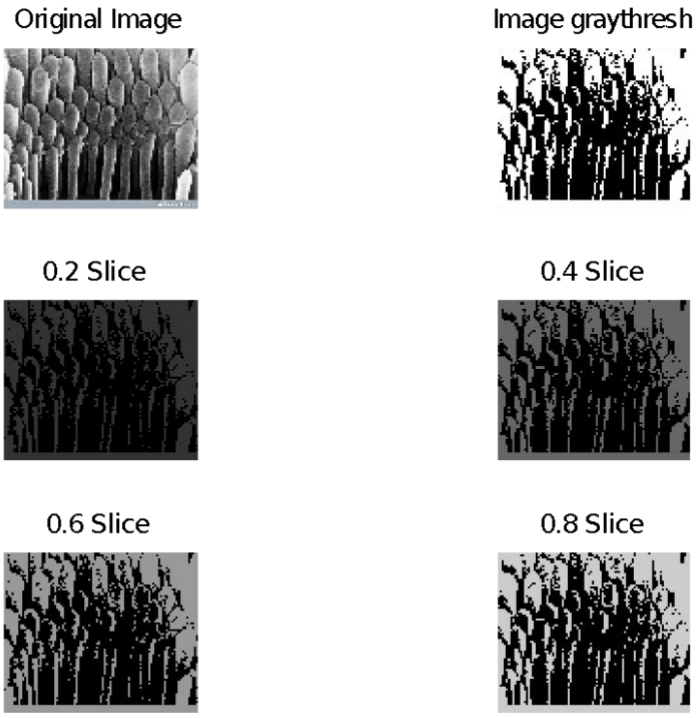
Sobel Vertical

10. Image Compression by DCT,DPCM, HUFFMAN coding

11. Implementation of image restoring techniques

12. Implementation of Image Intensity slicing technique for image enhancement

```
i=imread('earcell.jpg');
subplot(3,2,1);imshow(i); title('Original Image');
l=im2double(i);

level=graythresh(l);
BW = im2bw(l,level);
subplot(3,2,2); imshow(BW); title('Image graythresh');

level1=0.2*BW;
subplot(3,2,3); imshow(level1); title('0.2 Slice');

level2=0.4*BW;
subplot(3,2,4); imshow(level2);title('0.4 Slice');

level3=0.6*BW;
subplot(3,2,5); imshow(level3);title('0.6 Slice');

level4=0.8*BW;
subplot(3,2,6); imshow(level4); title('0.8 Slice');
```



13. Canny edge detection Algorithm

```
i= imread('cancercell.jpg');
g=rgb2gray(i);
subplot(2,2,1); imshow(i); title('Original Image');
subplot(2,2,2); imshow(g); title('Gray Image');
c=edge(g,'canny');
subplot(2,2,3); imshow(c); title('Canny output');
```