1. RGB COMPONENTS

```
a = imread('baby.jpg');
b = imread('baby.jpg');
c = imread('baby.jpg');
d = imread('baby.jpg');

b(:,:,2)=0;
b(:,:,3)=0;
c(:,:,3)=0;
c(:,:,1)=0;
d(:,:,2)=0;
d(:,:,1)=0;

subplot(2,2,1),imshow(a);
subplot(2,2,2),imshow(b);
subplot(2,2,3),imshow(c);
subplot(2,2,4),imshow(d);
```

2. False Contouring

```
clc;
clear all;
close all;

a = imread('cameraman.tif');

subplot(3,2,1),imshow(a);
title('Original Image');
subplot(3,2,2),imshow(grayslice(a,128),gray(128));
title('Image with 128 gray level');
subplot(3,2,3),imshow(grayslice(a,64),gray(64));
title('Image with 64 gray level');
subplot(3,2,4),imshow(grayslice(a,32),gray(32));
title('Image with 32 gray level');
subplot(3,2,5),imshow(grayslice(a,16),gray(16));
title('Image with 16 gray level');
subplot(3,2,6),imshow(grayslice(a,8),gray(8));
title('Image with 8 gray level');
```

3. Bit Plane Slicing

```
clc;
clear all;
close all;
a = imread('cameraman.tif');
[m, n]= size(a);
for i =1:m
    for j =1:n
        b7(i,j) = bitand(a(i,j),128);
    end
end
```

```
for i =1:m
    for j =1:n
        b6(i,j) = bitand(a(i,j),64);
    end
end

for i =1:m
    for j =1:n
        b5(i,j) = bitand (a(i,j),32);
    end
end

for i =1:m
    for j =1:n
        b4(i,j) = bitand (a(i,j),16);
    end
end

for i =1:m
    for j =1:n
        b3(i,j) = bitand (a(i,j),8);
    end
end

for i =1:m
    for j =1:n
        b2(i,j) = bitand(a(i,j),4);
    end
end

for i =1:m
    for j =1:n
        b1(i,j) = bitand(a(i,j),2);
    end
end

for i =1:m
    for j =1:n
        b0(i,j) = bitand(a(i,j),1);
    end
end

subplot (3,3,1) , imshow ( a ) , title ('original image'),
subplot (3,3,2) , imshow ( b0 ) ,title ('0th bitplane image'),
subplot (3,3,3) , imshow ( b1 ) , title ( '1st bitplane image'),
subplot (3,3,4) , imshow ( b2 ) , title ( '2nd bitplane image'),
subplot (3,3,5) , imshow ( b3 ) , title ( '3rd bitplane image'),
subplot (3,3,6) , imshow ( b4 ) , title ( '4th bitplane image'),
subplot (3,3,7) , imshow ( b5 ) , title ( '5th bitplane image'),
subplot (3,3,8) , imshow ( b6 ) , title ( '6th bitplane image'),
subplot (3,3,9) , imshow ( b7 ) , title (' 7th bitplane image');
```

3. Histogram Equalization

```
clc;
clear all;
close all;

a = imread('cameraman.tif')
subplot(2,2,1);imshow(a);title('Original Image');
subplot(2,2,2);imhist(a,255);
j = histeq(a);
subplot(2,2,3);imshow(j);title('Equalized Image');
subplot(2,2,4);imhist(j,255);
```

4. High pass filter butterworth model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
[m,n] = size(i);

for u = 1:m
    for v = 1:n
        D(u,v) = ((u-(m/2))^2 + (v-(n/2))^2)^(1/2);
    end
end

for u=1:m
    for v=1:n
        H(u,v) = 1 / (1+(20/D(u,v))^4);
    end
end

F = fft2(i);
F = fftshift(F);
y = F.*H;
y1 = ifftshift(y);
y2 = ifft2(y1);
subplot(121),imshow(i),title('Original Image');
subplot(122),imshow(uint8(y2));
title('High pass filtered Image [Butterworth]');
```

4. High pass filter gaussian model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
[m,n] = size(i);
for u = 1:m
    for v = 1:n
        D(u,v) = ((u-(m/2))^2 + (v-(n/2))^2)^(1/2);
    end
end
```

```
for u=1:m
    for v=1:n
        H(u,v)=1/(1+(20/D(u,v))^2);
    end
end

F = fft2(i);
F = fftshift(F);
y = F.*H;
y1 = ifftshift(y);
y2 = ifft2(y1);
subplot(121),imshow(i),title('Original Image');
subplot(122),imshow(uint8(y2));
title('Highpass filtered Image [Gaussian]');
```

4. Low pass filter butterworth model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
[m,n] = size(i);

for u = 1:m
    for v = 1:n
        D(u,v) = ((u-(m/2))^2 + (v-(n/2))^2)^(1/2);
    end
end

for u=1:m
    for v=1:n
        H(u,v) = 1 / (1+(D(u,v)/20)^4);
    end
end

F = fft2(i);
F = fftshift(F);
y = F.*H;
y1 = ifftshift(y);
y2 = ifft2(y1);
subplot(121),imshow(i),title('Original Image');
subplot(122),imshow(uint8(y2));
title('low pass filtered Image [Butterworth]');
```

4. Low pass filter gaussian model

```
clc;
clear all;
close all;

I = imread(‘cameraman.tif’);
[m,n] = size(i);
for u = 1:m
```

```
    for v = 1:n
        D(u,v) = ((u-(m/2))^2 + (v-(n/2))^2)^(1/2);
    end
end

for u=1:m
    for v=1:n
        H(u,v)=1/(1+(D(u,v)/20)^2);
    end
end

F = fft2(i);
F = fftshift(F);
y = F.*H;
y1 = ifftshift(y);
y2 = ifft2(y1);
subplot(121),imshow(i),title(‘Original Image’);
subplot(122),imshow(uint8(y2));
title(‘lowpass filtered Image [Gaussian]’);
```

5. average filter gaussian noise model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
subplot(131),imshow(i),title('Original Image');
h = ones(5,5)/25;
i_noise = imnoise(i,'gaussian',0,.01);
j = imfilter(i_noise,h);
subplot(132);imshow(i_noise),title('Noise Image');
subplot(133);imshow(j),title('Average filtered image');
```

5. average filter salt pepper noise model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
subplot(131),imshow(i),title('Original Image');
h = ones(5,5)/25;
i_noise = imnoise(i,'salt & pepper',.02);
j = imfilter(i_noise,h);
subplot(132);imshow(i_noise),title('Noise Image');
subplot(133);imshow(j),title('Average filtered image');
```

5. median filter gaussian noise model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
subplot(131),imshow(i),title('Original Image');
h = ones(5,5)/25;
i_noise = imnoise(i,'gaussian',0,.01);
j = medfilt2(i_noise,[5,5]);
subplot(132);imshow(i_noise),title('Noise Image');
subplot(133);imshow(j),title('Median filtered image');
```

5. median filter salt and pepper noise model

```
clc;
clear all;
close all;

i = imread('cameraman.tif');
subplot(131),imshow(i),title('Original Image');
h = ones(5,5)/25;
i_noise = imnoise(i,'salt & pepper',.02);
j = medfilt2(i_noise,[5,5]);
subplot(132);imshow(i_noise),title('Noise Image');
subplot(133);imshow(j),title('Median filtered image');
```

6. Wiener filter

```
clc;
clear all;
close all;

I = checkerboard(8);
imshow(I);

n = imnoise(zeros(size(I)),"gaussian",0,0.01);
[R,psf] = degradation_model(I,n);
sn = abs(fft2(n));
na=sum(sn(:))/numel(n);
sf=abs(fft2(I));
fa=sum(sf(:))/numel(I);
nspr=na/fa;
i_restore = deconvwnr(R, psf , nspr );
subplot(133),imshow(i_restore),title('Restored Image');

function [R,psf]=degradation_model(I,n)
subplot(131), imshow(I),title('Original Image');
psf=fspecial('motion',7,45);
i_degrade=imfilter(I,psf,'circular');
R=i_degrade+n;
subplot(132),imshow(R),title('Degraded Image');
end
```

7. Edge Detection

```
clc;
clear all;
close all;

a = imread('cameraman.tif');
b = edge(a,'roberts');
c = edge(a,'sobel');
d = edge(a,'prewitt');
e = edge(a,'log');
f = edge(a,'canny');

subplot(321),imshow(a),title('Original Image');
subplot(322),imshow(b),title('roberts');
subplot(323),imshow(c),title('sobel');
subplot(324),imshow(d),title('prewitt');
subplot(325),imshow(e),title('log');
subplot(326),imshow(f),title('canny');
```