**IP&ML Practical 1**

clc

clear all

close all

a=imread('C:\Users\djsce.student\Desktop\prac11.jpg');

subplot(4,4,1);

imshow(a);

title('original');

b=imresize(a,[256 300]);

imwrite(b,'C:\Users\djsce.student\Desktop\prac111.jpg');

subplot(4,4,2);

imshow(b);

title('resized');

R=a(:,:,1) %extracting red plane

subplot(4,4,3)

imshow(R)

title('R component');

G=a(:,:,2) %extracting green plane

subplot(4,4,4)

imshow(G)

title('G component');

B=a(:,:,3)%extracting blue plane

subplot(4,4,5)

imshow(B)

title('B component');

h=cat(3,R,G,B)%combining RGB plane

subplot(4,4,6)

imshow(h)

title('recombining');

c=rgb2gray(a);

subplot(4,4,7)

imshow(c)

title('grayscale 1');

t=0.30\*R+0.59\*G+0.11\*B;

subplot(4,4,8)

imshow(t)

title('grayscale 2');

i=imrotate(a,-45,'bilinear','loose'); %try with loose and crop

subplot(4,4,9);

imshow(i)

title('rotated');

tform= maketform('affine',[1 0 0;0.5 1 0; 0 0 1]);

z=imtransform(a,tform);

subplot(4,4,10);

imshow(z);

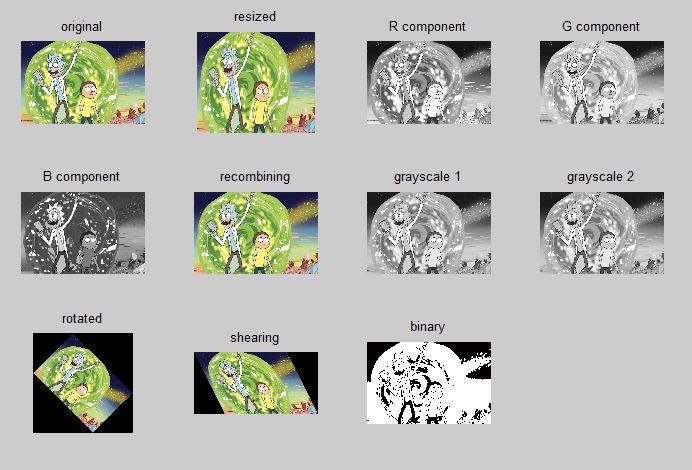
title('shearing');

d=im2bw(a,0.5)

subplot(4,4,11)

imshow(d)

title('binary')



CODE 2:

clc

clear all

close all

a=imread('C:\Users\djsce.student\Desktop\prac1.jpg');

subplot(4,4,1);

imshow(a)

title('image 1');

c=rgb2gray(a);

subplot(4,4,2);

imshow(c)

title('grayscale image 1');

b=imread('C:\Users\djsce.student\Desktop\practical1.png');

subplot(4,4,3);

imshow(b)

title('image 2');

d=rgb2gray(b);

subplot(4,4,4)

imshow(d)

title('grayscale image 2');

e=c+d;

subplot(4,4,5);

imshow(e)

title('add');

f=c-d;

subplot(4,4,6);

imshow(f)

title('subtract');

g=im2bw(a,0.5)

subplot(4,4,7)

imshow(g)

title('binary image 1');

h=im2bw(b,0.5)

subplot(4,4,8)

imshow(h)

title('binary image 2');

i=g & h;

subplot(4,4,9);

imshow(i)

title('logical AND');

j=g | h;

subplot(4,4,10);

imshow(j)

title('logical OR');

k=~g;

subplot(4,4,11);

imshow(k)

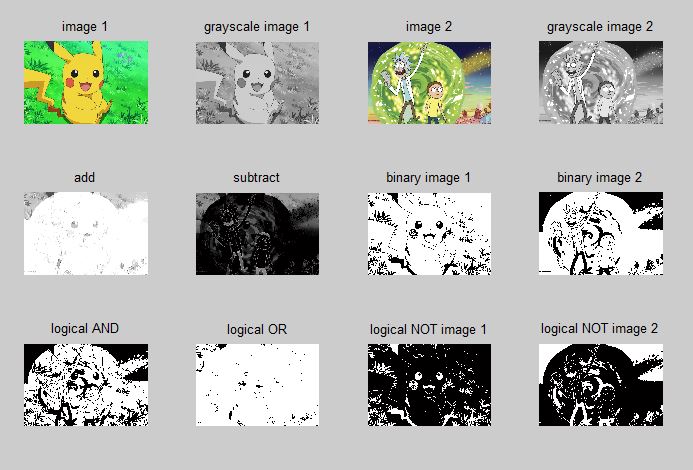
title('logical NOT image 1');

l=~h;

subplot(4,4,12);

imshow(l)

title('logical NOT image 2');



**IPML Practical 3**

**Code:**

clc;

clear all;

close all;

a=imread('C:\Users\djsce.student\Desktop\noddy.jpg');

a=rgb2gray(a);

b1=imresize(a,[256 256]);

subplot(4,4,1)

imshow(b1);

a=double(b1);

title('Original image');

m1=(1/9)\*[1 1 1;1 1 1;1 1 1];

for i=2:255

for j=2:255

b(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m1));

end

end

subplot(4,4,2)

imshow(uint8(b));

title('Low pass-1');

m2=(1/25)\*[1 1 1 1 1;1 1 1 1 1;1 1 1 1 1;1 1 1 1 1;1 1 1 1 1];

for i=3:254

for j=3:254

c(i,j)=sum(sum(a(i-2:i+2,j-2:j+2).\*m2));

end

end

subplot(4,4,3)

imshow(uint8(c));

title('Low pass-2');

m3=(1/16)\*[1 2 1;2 4 2;1 2 1];

for i=2:255

for j=2:255

d(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m3));

end

end

subplot(4,4,4)

imshow(uint8(d));

title('Low pass-3');

m4=(1/9)\*[-1 -1 -1;-1 8 -1;-1 -1 -1];

for i=2:255

for j=2:255

e(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m4));

end

end

subplot(4,4,5)

imshow(uint8(e));

title('High pass-1');

m5=[0 -1 0;-1 4 -1;0 -1 0];

for i=2:255

for j=2:255

f(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m5));

end

end

subplot(4,4,6)

imshow(uint8(f));

title('High pass-2');

m6=[-1 -2 -1;-2 12 -2;-1 -2 -1];

for i=2:255

for j=2:255

g(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m6));

end

end

subplot(4,4,7)

imshow(uint8(g));

title('High pass-3');

m7=[-1 -1 -1;0 0 0;1 1 1];

for i=2:255

for j=2:255

h(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m7));

end

end

subplot(4,4,8)

imshow(uint8(h));

title('High pass-4');

m8=[-1 0 1;-1 0 1;-1 0 1];

for i=2:255

for j=2:255

k(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m8));

end

end

subplot(4,4,9)

imshow(uint8(k));

title('High pass-5');

m9=[-1 -1 -1;-1 8.9 -1;-1 -1 -1];

for i=2:255

for j=2:255

l(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m9));

end

end

subplot(4,4,10)

imshow(uint8(l));

title('High boost-1');

m=imnoise(b1,'salt & pepper',0.05);

subplot(4,4,11)

imshow((m));

title('Noisy image');

for i=2:255

for j=2:255

p(i,j)=median(median(a(i-1:i+1,j-1:j+1)));

end

end

subplot(4,4,12)

imshow(uint8(p));

title('Median filter');

**Output:**



**IPMV Experiment- 4**

**CODE:**

clc;

close all;

clear all;

a=imread('C:\Users\djsce.student\Desktop\shin.png');

a=imresize(a,[256 256]);

a=rgb2gray(a);

a=im2double(a);

subplot(5,5,1);

imshow(a);

title('Original');

f1=fft2(a);

subplot(5,5,2);

imshow(f1);

title('Fourier transform');

for i=1:256

for j=1:256

a2(i,j)=a(i,j)\*((-1)^(i+j));

end

end

subplot(5,5,3);

imshow(a2);

title('Translated image');

f2=fft2(a2);

subplot(5,5,4);

imshow(f2);

title('Fourier transform of Translated image');

for u=1:256

for v=1:256

D(u,v)=(((u-128)^2)+((v-128)^2))^0.5;

end

end

D0=50;

for i=1:256

for j=1:256

if D(i,j)<=D0

H1(i,j)=1;

else

H1(i,j)=0;

end

end

end

subplot(5,5,5);

imshow(H1);

title('H1');

ILPF=f2.\*H1;

subplot(5,5,6);

imshow(ILPF);

title('Ideal Low Pass Filter');

ILP=ifft2(ILPF);

subplot(5,5,7);

imshow(ILP);

title('ILPF image');

for i=1:256

for j=1:256

if D(i,j)<=D0

H2(i,j)=1/1+((D(i,j)/D0)^(2\*2));

else

H2(i,j)=0;

end

end

end

subplot(5,5,8);

imshow(H2);

title('H2');

BLPF=f2.\*H2;

subplot(5,5,9);

imshow(BLPF);

title('Butterworth LPF Image');

BLP=ifft2(BLPF);

subplot(5,5,10);

imshow(BLP);

title('BLPF image');

for i=1:256

for j=1:256

if D(i,j)<=D0

H3(i,j)=exp((-D(i,j)^2)/(2\*((D0)^2)));

else

H3(i,j)=0;

end

end

end

subplot(5,5,11);

imshow(H3);

title('H3');

GLPF=f2.\*H3;

subplot(5,5,12);

imshow(GLPF);

title('Gaussian LPF Image');

GLP=ifft2(GLPF);

subplot(5,5,13);

imshow(GLP);

title('GLPF image');

for i=1:256

for j=1:256

if D(i,j)>=D0

H4(i,j)=1;

else

H4(i,j)=0;

end

end

end

subplot(5,5,14);

imshow(H4);

title('H4');

IHPF=f2.\*H4;

subplot(5,5,15);

imshow(IHPF);

title('Ideal High Pass Filter');

IHP=ifft2(IHPF);

subplot(5,5,16);

imshow(IHP);

title('IHPF image');

for i=1:256

for j=1:256

if D(i,j)>=D0

H5(i,j)=1/1+((D0/D(i,j))^(2\*2));

else

H5(i,j)=0;

end

end

end

subplot(5,5,17);

imshow(H5);

title('H5');

BHPF=f2.\*H5;

subplot(5,5,18);

imshow(BHPF);

title('Butterworth HPF Image');

BHP=ifft2(BHPF);

subplot(5,5,19);

imshow(BHP);

title('BHPF image');

for i=1:256

for j=1:256

if D(i,j)>=D0

H6(i,j)=1-exp((-D(i,j)^2)/(2\*((D0)^2)));

else

H6(i,j)=0;

end

end

end

subplot(5,5,20);

imshow(H6);

title('H6');

GHPF=f2.\*H6;

subplot(5,5,21);

imshow(GHPF);

title('Gaussian HPF Image');

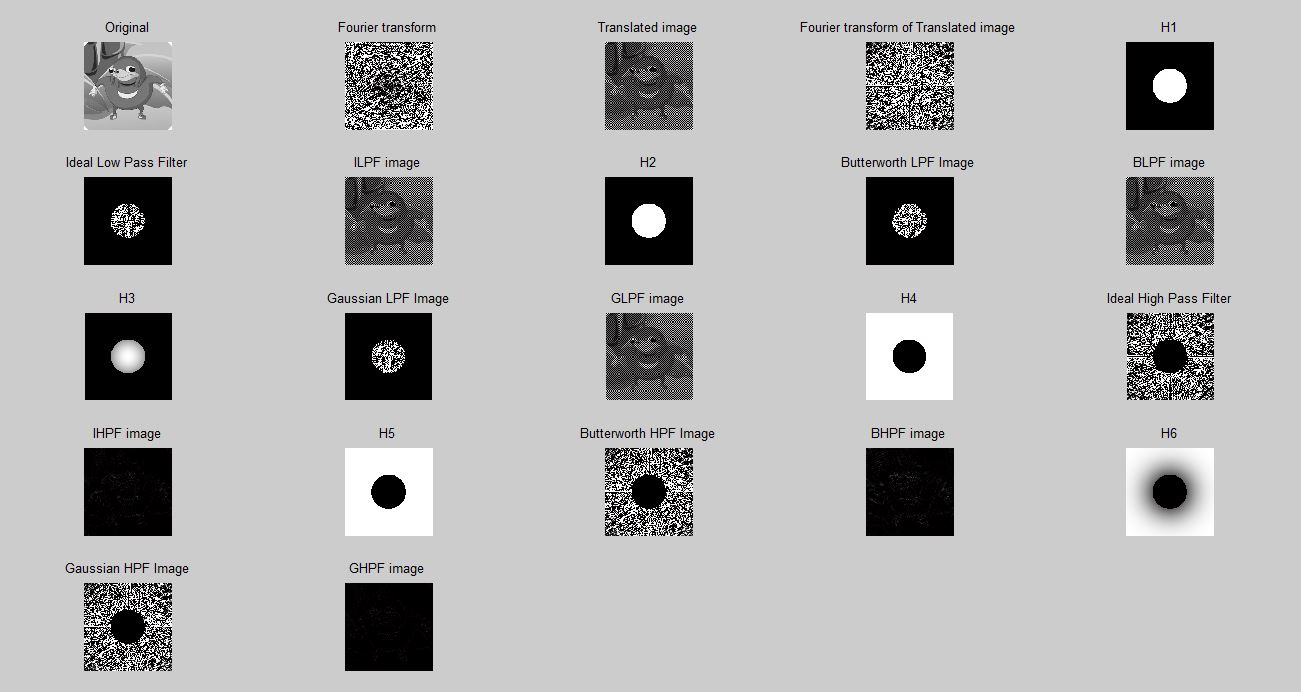
GHP=ifft2(GHPF);

subplot(5,5,22);

imshow(GHP);

title('GHPF image');

**OUTPUT:**

****

**IPMV EXPERIMENT 5**

**CODE:**

clc;

close all;

clear all;

a=imread('C:\Users\djsce.student\Desktop\str.png');

a=imresize(a,[256 256]);

b=im2bw(a,0.5);

subplot(2,3,1);

imshow(b);

title('Original');

SE = strel('diamond', 5);

c = imdilate(b,SE);

subplot(2,3,2);

imshow(c);

title('Dilation');

d = imerode(b,SE);

subplot(2,3,3);

imshow(d);

title('Erotion');

e = imopen(b,SE);

subplot(2,3,4);

imshow(e);

title('Opening');

f = imclose(b,SE);

subplot(2,3,5);

imshow(f);

title('Closing');

g=c-b;

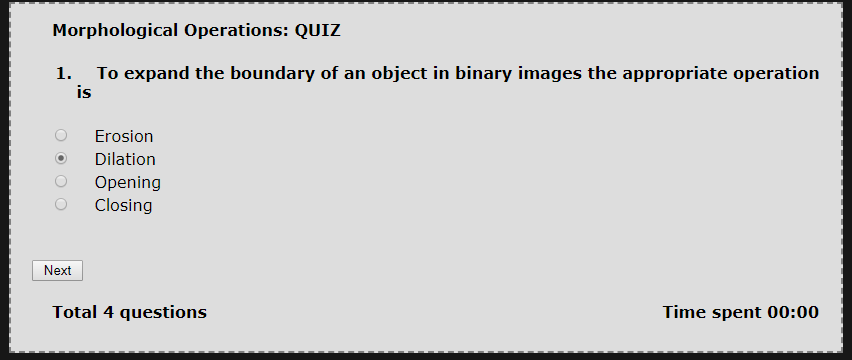
subplot(2,3,6);

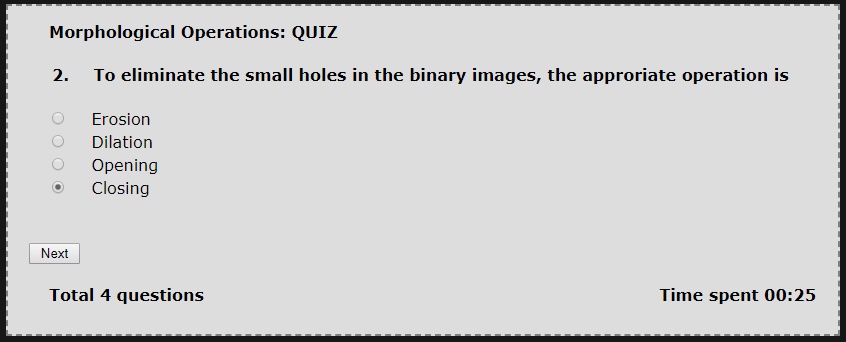
imshow(g);

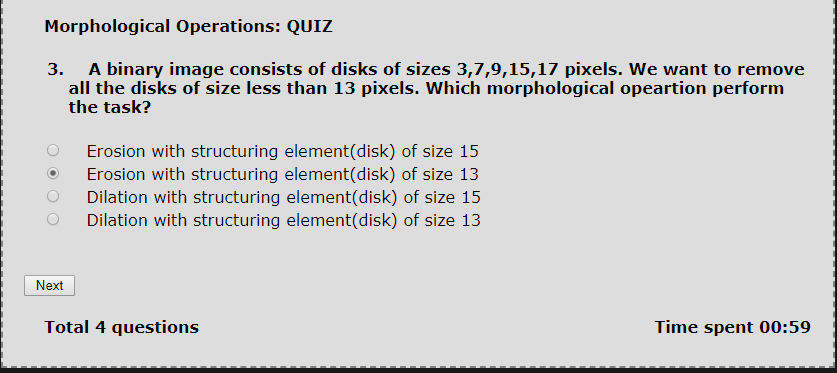
title('Boundary Extraction');

**OUTPUT:**

****







**IPML Experiment 6**

**Code:**

clc;

clear all;

close all;

a=imread('C:\Users\djsce.student\Desktop\test2.jpg');

subplot(3,3,1)

imshow(a)

title('Original image');

a=rgb2gray(a);

b1=imresize(a,[256 256]);

subplot(3,3,2)

imshow(b1);

a=double(b1);

title('GRAY SCALED IMAGE');

m1=[1 1;-1 -1]

for i=2:255

for j=2:255

c(i,j)=sum(sum(a(i-1:i,j-1:j).\*m1));

end

end

subplot(3,3,3)

imshow(uint8(c));

title('Robert H&V MASK');

m2=[-1 -1 -1;0 0 0;1 1 1]

for i=2:255

for j=2:255

x(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m2));

end

end

subplot(3,3,4)

imshow(uint8(x));

title('Prewitts H MASK');

m3=[-1 0 1;-1 0 1;-1 0 1]

for i=2:255

for j=2:255

y(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m3));

end

end

subplot(3,3,5)

imshow(uint8(y));

title('Prewitts V MASK');

m4=[-2 -1 0;-1 0 1;0 1 2]

for i=2:255

for j=2:255

xy(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m4));

end

end

subplot(3,3,6)

imshow(uint8(xy));

title('Prewitts H&V MASK');

m5=[-2 -2 0;-2 0 2;0 2 2]

for i=2:255

for j=2:255

xy1(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m5));

end

end

subplot(3,3,7)

imshow(uint8(xy1));

title('SOBEL H&V MASK');

m6=[0 1 0;1 -4 1;0 1 0]

for i=2:255

for j=2:255

xy2(i,j)=sum(sum(a(i-1:i+1,j-1:j+1).\*m6));

end

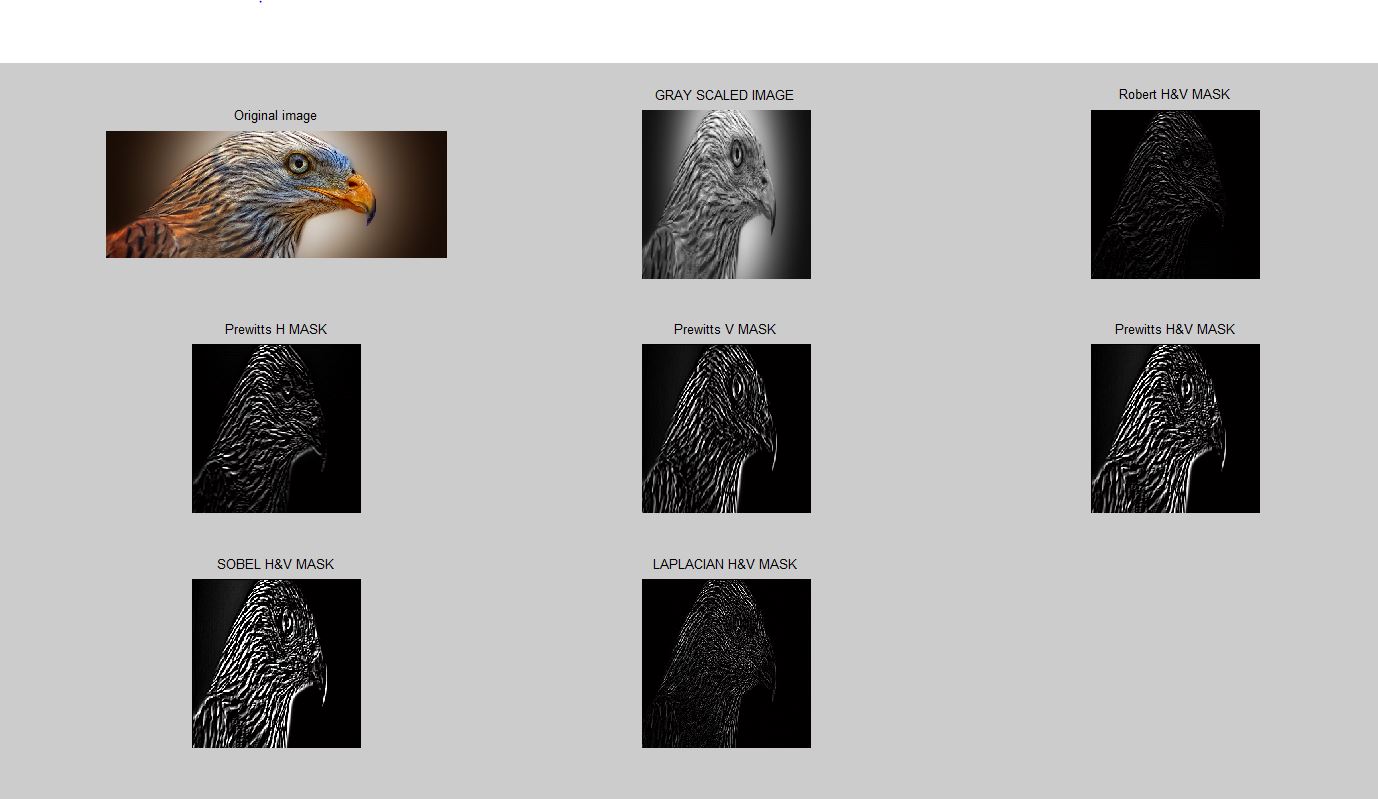
end

subplot(3,3,8)

imshow(uint8(xy2));

title('LAPLACIAN H&V MASK');

**Output:**



**Code:**

clc;

clear all;

close all;

a=imread('C:\Users\djsce.student\Desktop\test2.jpg');

subplot(3,3,1)

imshow(a)

title('Original image');

a=rgb2gray(a);

b1=imresize(a,[256 256]);

subplot(3,3,2)

imshow(b1);

title('GRAY SCALED IMAGE');

subplot(3,3,3)

BW1 = edge(b1,'roberts');

imshow(BW1);

title('Robert H&V MASK');

subplot(3,3,5)

BW2 = edge(b1,'sobel')

imshow(BW2);

title('SOBEL H&V MASK');

subplot(3,3,4)

BW3 = edge(b1,'prewitt')

imshow(BW3);

title('PREWITTS H&V MASK');

subplot(3,3,6)

BW4 = edge(b1,'log')

imshow(BW4);

title('log MASK');

subplot(3,3,7);

[BW,thresh]=edge(a,'zerocross');

BW6=edge(b1,'zerocross',thresh);

imshow(BW6);

title('Zerocross mask');

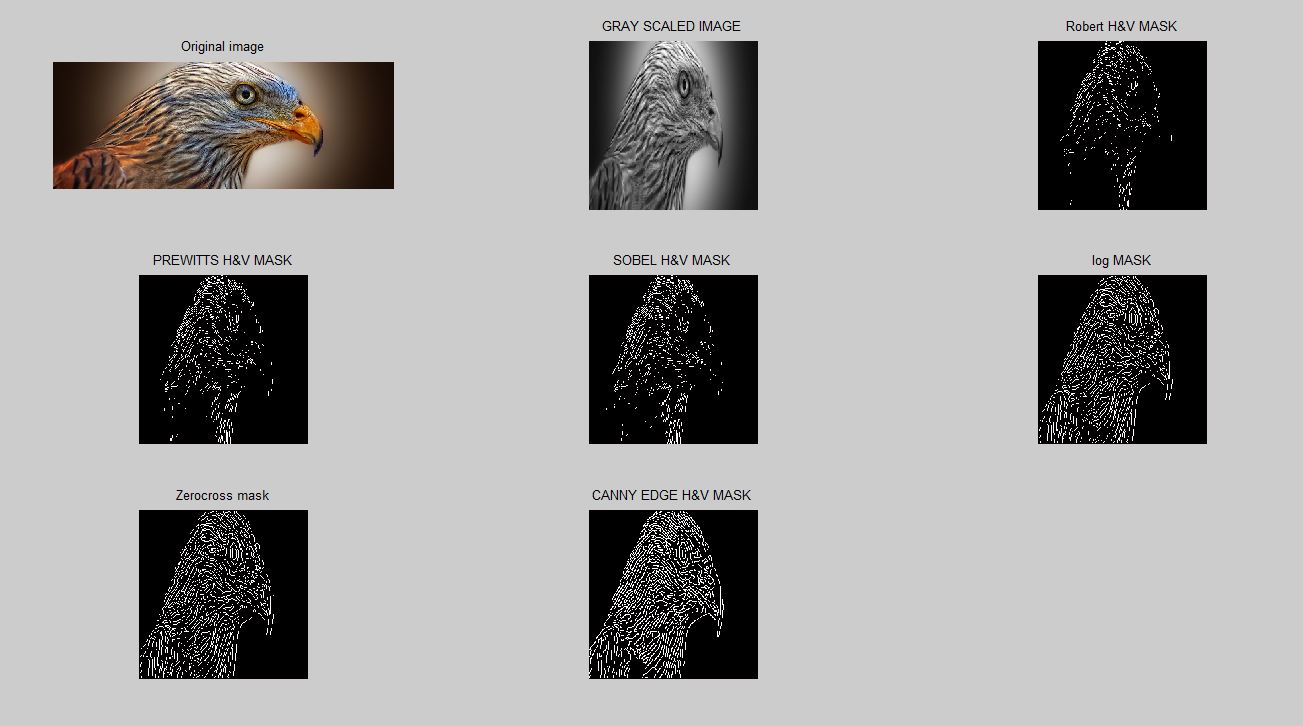
subplot(3,3,8)

BW5 = edge(b1,'canny')

imshow(BW5);

title('CANNY EDGE H&V MASK');

**Output:**

****

**IPMV Experiment 7**

**Code:**

clc

clear all

close all

A=imread('C:\Users\djsce.student\Desktop\test3.jpg');

A=rgb2gray(A);

subplot(3,3,1)

imshow(A);

title('Gray Scale Image')

psf=fspecial('motion',41,11);

B=imfilter(A,psf,'conv','circular');

subplot(3,3,2)

imshow(B);

title('Motion Blurred Image')

C = deconvwnr(B,psf,0.001);

subplot(3,3,3)

imshow(C)

title('Wiener Filtered Image')

D=imnoise(B,'salt & pepper',0.2);

subplot(3,3,4)

imshow(D)

title('Salt & Pepper Noise')

E=imnoise(B,'gaussian',0,0.01);

subplot(3,3,5)

imshow(E)

title('Gaussian Noise')

F = deconvwnr(D,psf,0.1);

subplot(3,3,6)

imshow(F)

title('Salt & Pepper Noise Wiener Filtered Image')

G = deconvwnr(E,psf,0.1);

subplot(3,3,7)

imshow(G)

title('Gaussian Noise Wiener Filtered Image')

H=medfilt2(D);

subplot(3,3,8)

imshow(H)

title('Salt & Pepper Noise Median Filtered Image')

I=medfilt2(E);

subplot(3,3,9)

imshow(I)

title('Gaussian Noise Median Filtered Image')

**Output:**



**IPML Experiment 8**

Code

clc;

clear all;

close all;

a=xlsread('C:\Users\djsce.student\Desktop\irisdataset1.xlsx');

meas=xlsread('C:\Users\djsce.student\Desktop\meas.xlsx');

species=xlsread('C:\Users\djsce.student\Desktop\species\_num.xlsx');

X=a(1:100,:)

Y=species(1:100);

rand\_num=randperm(100);

X\_train=X(rand\_num(1:80),:);

Y\_train=Y(rand\_num(1:80),:);

X\_test=X(rand\_num(81:end),:);

Y\_test=Y(rand\_num(81:end),:);

SVMModel=fitcsvm(X\_train,Y\_train);

[labels]=predict(SVMModel,X\_test);

sv=SVMModel.SupportVectors;

figure

gscatter(X(:,1),X(:,2),Y)

hold on

0

legend('setosa','versioclor','support vector')

hold off

idx=(Y\_test()==1);

idx1=(Y\_test()==2);

p=length(Y\_test(idx));

n=length(Y\_test(idx1));

N=p+n;

tp=sum(Y\_test(idx)==labels(idx));

tn=sum(Y\_test(idx1)==labels(idx1));

fp=n-tn;

fn=p-tp;

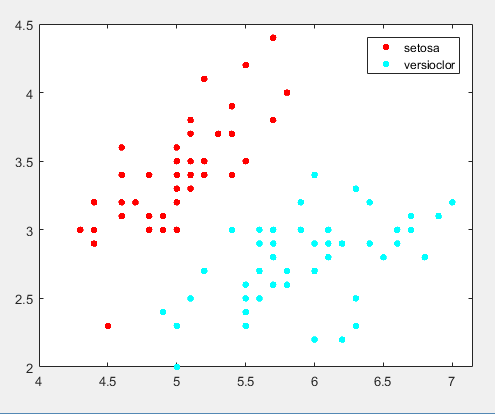
tp\_rate=tp/p;

tn\_rate=tn/n;

accuracy=(tp+tn)/N;

C=confusionmat(Y\_test,labels)

OUTPUT



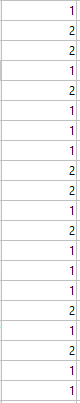
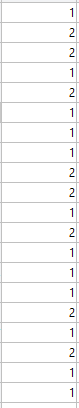
Confusion matrix

C =

12 0

1. 8

PREDICTED VALUES TESTED VALUES



**IPML Experiment 9**

Code

clc;

clear all;

close all;

%CLASS NUMERICAL WHERE X=[1,1.5;1,4.5;2,1.5;2,3.5;3,5;5,6]

k=2;

X=[1,1.5;1,4.5;2,1.5;2,3.5;3,5;5,6];

[idx,C]=kmeans(X,k,'Replicate',3,'Display','Iter');

disp(idx);

disp(C);

figure(1)

gscatter(X(:,1),X(:,2),idx,'bgm')

axis([0 8 0 8]) %to change the limits of axis=axis([xmin xmax ymin ymax])

hold on

plot(C(:,1),C(:,2),'kx','Markersize',12,'LineWidth',8)

legend('Cluster 1','Cluster 2','Cluster Centroid')

%To perform Clustering on Iris DataSet

k=3;

X=xlsread('C:\Users\djsce.student\Desktop\IPML8\iris1.xlsx');

meas=xlsread('C:\Users\djsce.student\Desktop\IPML8\irisdataset.xlsx');

species=xlsread('C:\Users\djsce.student\Desktop\IPML8\species.xlsx');

[idx,C]=kmeans(X(:,1:2),k,'Replicate',3,'Display','Iter');

disp(idx);

disp(C);

figure(2)

gscatter(X(:,1),X(:,2),idx,'bgm')

axis([3 10 1 8]) %to change the limits of axis=axis([xmin xmax ymin ymax])

hold on

plot(C(:,1),C(:,2),'kx','Markersize',12,'LineWidth',8)

legend('Cluster 1','Cluster 2','Cluster3 ','Cluster Centroid')

OUTPUT

