4 AIM:

Write and execute the MATLAB Code to learn the image

filtering operations of smoothing and sharpening in spatial

domain.

Refer the Lenna.tif color image and perform the following:

(i) Convert the given image to grey level image and visualize

the obtained image.

(ii) Perform the following filtering operations on the image

obtained in (i).

(A)Smoothing using the following spatial domain filters

(1) Average Filter (Box Filter)

(2) Weighted Average Filter

(3) Min Filter

(4) Median Filter

(5) Max Filter

(B) Sharpening using the following spatial domain filters

(1) All Four Versions of Laplacian Filter

(2) Roberts Cross Gradient Operators

(3) Sobel Operators

(iii) Plot the results obtained in (ii)(A)(1), and (ii)(A)(2)

(Single plot consists of 3 images, including original grey

Ii=imread('C:\Users\jadha\Desktop\dipimage\Lenna.tif');

I=rgb2gray(Ii);

double=im2double(I);

noice=double+randn(512,512)\*0.10;

noice1=imnoise(I,'gaussian',0.01);

%noice2=imnoise(I,'salt & pepper',0.02);

w=[1 1 1;1 1 1;1 1 1]/9;%Average filter

w1=[1 2 1;2 4 2;1 2 1]/16;%weighted filter

I1=imfilter(noice,w);

I2=imfilter(noice,w1);

subplot(1,3,1);

imshow(noice);

title("Original Image");

subplot(1,3,2);

imshow(I1);

title("Average filter");

subplot(1,3,3);

imshow(I2);

title("weighted filter");

a = imnoise(double,'salt & pepper',0.02);

med = medfilt2(a);

figure;

subplot(321);

imshow(a);

title(" salt & pepper Image");

subplot(322);

imshow(med);

title("Medium Filter");

x=rand(size(double));

double(x(:)<0.05)=0;

max\_Img = ordfilt2(double,9,ones(3,3));

subplot(323);

imshow(double);

title(" salt & pepper Image");

subplot(324);

imshow(max\_Img);

title("Maximum Filter");

a=rgb2gray(Ii);

a = im2double(a);

x=rand(size(a));

a(x(:)>0.95)=255;

min\_Img = ordfilt2(a,1,ones(3,3));

subplot(325);

imshow(a);

title(" salt & pepper Image");

subplot(326);

imshow(min\_Img);

title("Minimum Filter");

gray=rgb2gray(Ii);

dd = im2double(gray);

q3=[0 1 0;1 -4 1;0 1 0];

q4=[ 1 1 1;1 -8 1;1 1 1];

q5=[0 -1 0;-1 4 -1;0 -1 0];

q6=[-1 -1 -1;-1 8 -1;-1 -1 -1];

Q1=imfilter(dd,q3);

Q2=imfilter(dd,q4);

Q3=imfilter(dd,q5);

Q4=imfilter(dd,q6);

figure;

subplot(321);

imshow(dd);

title("Original Image");

subplot(322);

imshow(Q1);

title(" 1st version for Laplacian filter");

subplot(323);

imshow(Q2);

title(" 2st version for Laplacian filter");

subplot(324);

imshow(Q3);

title(" 3st version for Laplacian filter");

subplot(325);

imshow(Q4);

title(" 4st version for Laplacian filter");

dd1 = im2double(Ii);

gray1=rgb2gray(dd1);

hx=[+1 0;0 -1];

hy=[0 +1;-1 0];

gx=imfilter(gray1,hx);

gy=imfilter(gray1,hy);

gm=sqrt(gx.^2+gy.^2);

figure;

subplot(121);

imshow(gray1);

subplot(122);

imshow(gm);

title("Roberts cross Gradient Operator");

gray2=rgb2gray(Ii);

c = im2double(gray2);

for i=1:size(c,1)-2

for j=1:size(c,2)-2

gx1=((2\*c(i+2,j+1)+c(i+2,j)+c(i+2,j+2))-(2\*c(i,j+1)+c(i,j)+c(i,j+2)));

gy1=((2\*c(i+1,j+2)+c(i,j+2)+c(i+2,j+2))-(2\*c(i+1,j)+c(i,j)+c(i+2,j)));

b1(i,j)=sqrt(gx1.^2+gy1.^2);

end

end

figure;

subplot(121);

imshow(c);

subplot(122);

imshow(b1);

title("Sobel gradient");