1. Aim: - Image Processing Basics

## Program to calculate number of samples required for an image

*CODE:*

// Program to calculate the number of samples required for an image figure m = 4;

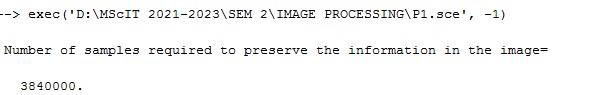
n = 6;

N = 400;

% Calculate the total number of samples required Fs = m \* N \* n \* N;

disp(['Number of samples required to preserve the information in the image = ', num2str(Fs)]);

## Output



1. Image Properties
   * Program to access image properties Dimension, height, width, number of channels, accessing and modifying any pixel

### CODE:

figure; i=imread("C:\ProgramFiles\scilab6.0.1\IPCV\images\lena.png"); s=size(i);

disp(s,"Dimensions"); disp(s(1),"height");

disp(s(2),"width");

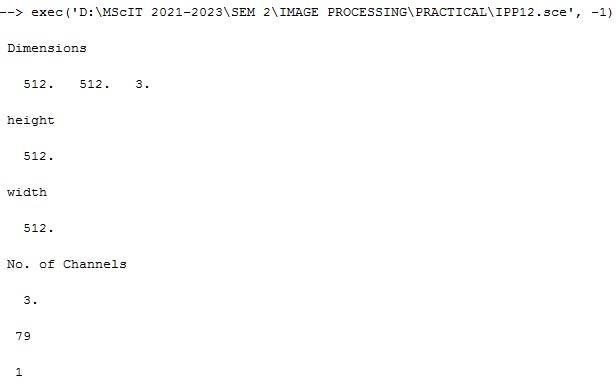
disp(s(3),"No. of Channels");

disp(i(100,100,3));

i(100,100,3)=1;

disp(i(100,100,3));

### OUTPUT:



1. Sampling
   * Program to study the effects of reducing the spatial resolution of a digital

image

Code

figure;

i=imread("C:\ProgramFiles\scilab-6.0.1\IPCV\images\Lena\_dark.png"); disp("sizeoforiginalimage",size(i));

subplot(3,3,1) imshow(i); title('OriginalImage') j1=imresize(i,0.8);

disp("sizeofresizedImage",size(j1)); subplot(3,3,2)

imshow(j1); title('resizedimage0.8') j2=imresize(i,0.5);

disp("sizeofresizedImage",size(j2)); subplot(3,3,3)

imshow(j2); title('resizedimage0.5')

## OUTPUT



1. Quantization
   * Program to study the effects of varying the number of intensity levels in a

digital image

Code

* + // Program to study the effects of varying the number of intensity levels in a

digital image

figure; i=imread('C:\ProgramFiles\scilab6.0.1\IPCV\images\Lena\_dark.png'); subplot(2,2,1);

imshow(i); title('originalimage') i=double(i); k1=(i\*255)/64;

subplot(2,2,2); k1=uint8(k1); imshow(uint8(k1)); title('Quantization64'); k2=(i\*255)/32;

subplot(2,2,3); k2=uint8(k2); imshow(uint8(k2)); title('Quantization32'); k3=(i\*255)/16;

subplot(2,2,4); k3=uint8(k3); imshow(uint8(k3)); title('Quantization16');

## OUTPUT



1. Image Addition
   * Program to perform image addition for noise reduction.

### Code

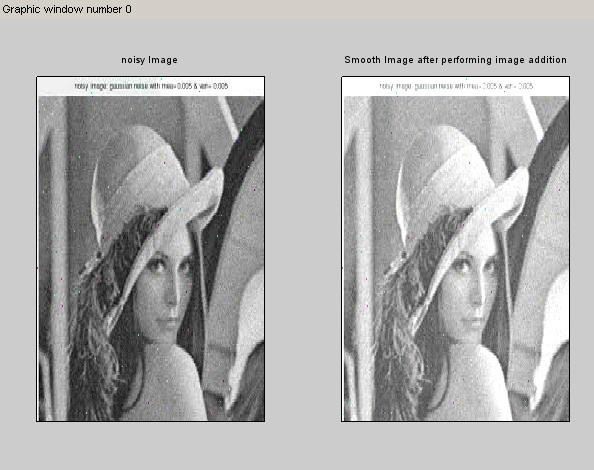
//Program to perform image addition for noise reduction. figure;

i=imread('C:\ProgramFiles\scilab-6.0.1\IPCV\images\noise.jpeg'); i=imnoise(i,'salt&pepper',0.001);

subplot(1,2,1); imshow(i); title('noisyImage'); k=imadd(i,50);//I+30 subplot(1,2,2); imshow(k);

title('Smooth Image after performing image addition');

## OUTPUT



## Image Subtraction

* + Program to compare images using subtraction for enhancing the difference

between image

Code

### // Program to compare images using subtraction for enhancing the difference between image

figure;

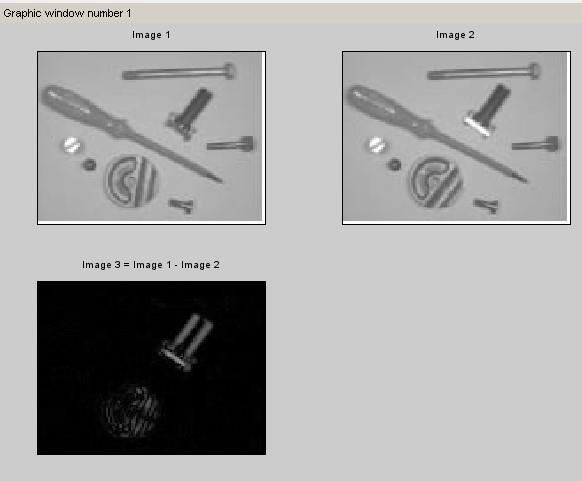
i=imread('C:\ProgramFiles\scilab-6.0.1\IPCV\images\tool1.jpeg'); j=imread('C:\ProgramFiles\scilab6.0.1\IPCV\images\tool2.jpeg'); subplot(2,2,1);

imshow(i); title('Image1');

subplot(2,2,2); imshow(j); title('Image2'); k=imabsdiff(i,j); subplot(2,2,3); imshow(k);

title('Image3=Image1-Image2');

## output



1. Aim: Histogram Processing
   1. Program to Plot Histogram of an Image

Code: -

Figure;

Img=imread(“C:\Program Files\scilab-6.0.2\IPCV\images\Lena\_dark.png”) Subplot(2,2,1) Title(‘OriginalImage’)

Imshow(img) H1=imhist(img); Subplot(2,2,2);

Plot(h1) Title(‘PredefinedHistogram,’); [r,c]=size(img);

H=zeros(1,256); S=0:255 Fori=1:r

Forj=1:c If(img(I,j)==0) Img(I,j)=1 End K=img(I,j);

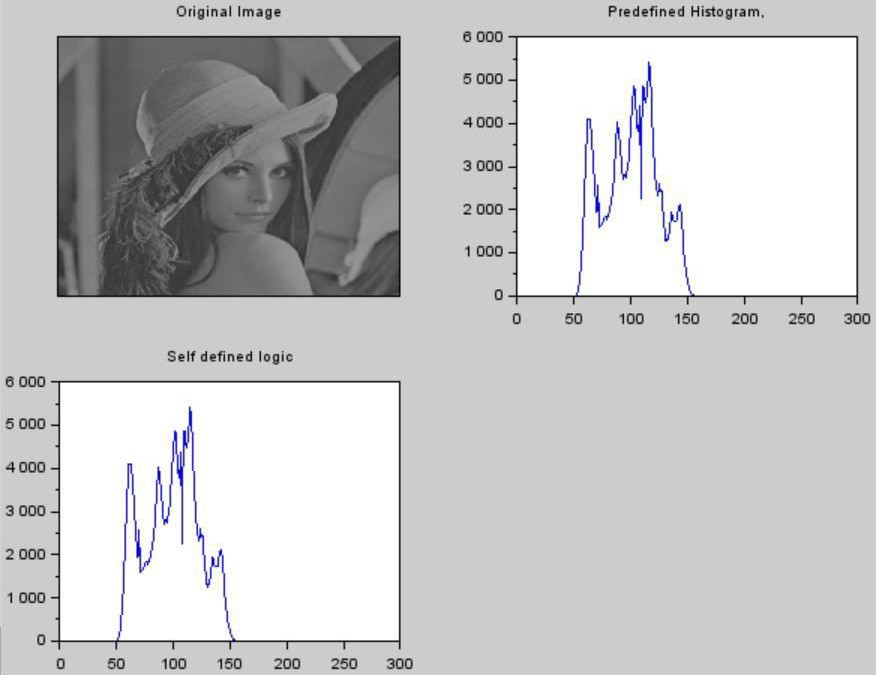
H(k)=h(k)+1;

End End

Subplot(2,2,3); Plot(h);

Title(‘Self defined logic’);

## Output



* 1. Plot Histogram of Low Contrast, Bright, dark and High Contrast Images

Code :

Clear all;

Img=imread(“C:\Program Files\scilab-6.0.2\IPCV\images\seed.tif”) Subplot(4,2,1)

Title(‘Original Image’)

Imshow(img)

H1=imhist(img);

Subplot(4,2,2); Plot(h1)

Title(‘OriginalHistogram’);

Darkimg=img

Darkimg=darkimg-80

Subplot(4,2,3)

Imshow(darkimg)

H2=imhist(darkimg)

Subplot(4,2,4) Plot(h2)

Title(“Histogram of dark image”)

Brightimg=img Brightimg=brightimg+100 Subplot(4,2,5) Imshow(brightimg) H3=imhist(brightimg) Subplot(4,2,6)

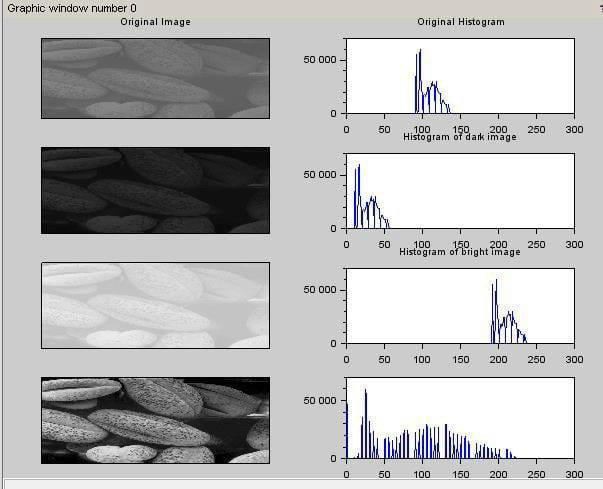
Plot(h3)

Title(“Histogram ofbrightimage”)

A=imread(“C:\Program Files\scilab-6.0.1\IPCV\images\seed.tif”); Mmin=min(a( ); Mmax=max(a( );

Lmin=0; Lmax=255;

A1=(a-mmin)\*(lmax-lmin)/(mmax-mmin)+lmin; Subplot(4,2,7)

Imshow(a1) H5=imhist(a1) Subplot(4,2,8) Plot(h5)

1. Aim: - Image Sharpening in Spatial Domain

First Order Derivative Filter

Code: figure;

p=imread('C:\Program Files\scilab-6.0.1\IPCV\images\morpex.png'); subplot(3,2,1); imshow(p);

title('OriginalImage'); d=double(p);

v=[10-1;20-2;10-1];//x-direction h=[-1-2-1;000;121];//y-direction [r1,c1]=size(p);

For i=2:1:r1-1 For j=2:1:c1-1

newv(i,j)=(v(1)\*d(i-1,j-1)+(v(2)\*d(i-1,j)+(v(3)\*d(i-1,j+1)+(v(4)\*d(i,j-

1)+(v(5)\*d(i,j)+(v(6)\*d(i,j+1)+(v(7)\*d(i+1,j- 1)+(v(8)\*d(i+1,j)+(v(9)\*d(i+1,j+1); end

end subplot(3,2,2);

imshow(uint8(newv);

title("Image after applying vertical sobel operator");

for i=2:1:r1-1 for j=2:1:c1-1

newh(i,j)=(h(1)\*d(i-1,j-1)+(h(2)\*d(i-1,j)+(h(3)\*d(i-1,j+1)+(h(4)\*d(i,j-

1)+(h(5)\*d(i,j)+(h(6)\*d(i,j+1)+(h(7)\*d(i+1,j- 1)+(h(8)\*d(i+1,j)+(h(9)\*d(i+1,j+1); end

end subplot(3,2,3);

imshow(uint8(newh);

title("Image after applying horizontal sobel operator");

v1=[-101;-202;-101];

h1=[121;000;-1-2-1];

[r1,c1]=size(p); For i=2:1:r1-1 For j=2:1:c1-1

newv1(i,j)=(v1(1)\*d(i-1,j-1)+(v1(2)\*d(i-1,j)+(v1(3)\*d(i-1,j+1)+(v1(4)\*d(i,j-

1)+(v1(5)\*d(i,j)+(v1(6)\*d(i,j+1)+(v1(7)\*d(i+1,j-1)+(v1(8)\*d(i+1,j)+(v1(9)\*d(i+1,j+1);

end end

subplot(3,2,4); imshow(uint8(newv1);

title("Image after applying vertical sobel operator rotated 180 degree"); for i=2:1:r1-1

for j=2:1:c1-1

newh1(i,j)=(h1(1)\*d(i-1,j-1)+(h1(2)\*d(i-1,j)+(h1(3)\*d(i-1,j+1)+(h1(4)\*d(i,j-

1)+(h1(5)\*d(i,j)+(h1(6)\*d(i,j+1)+(h1(7)\*d(i+1,j-1)+(h1(8)\*d(i+1,j)+(h1(9)\*d(i+1,j+1);

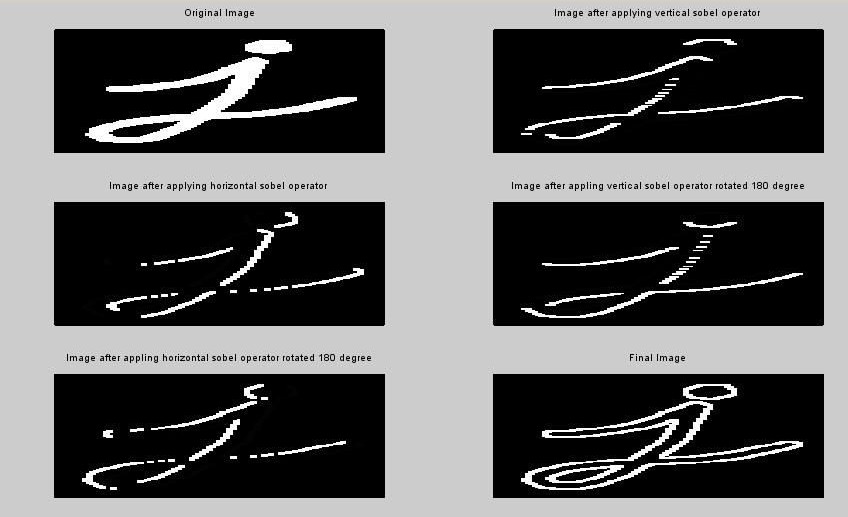
end end

subplot(3,2,5); imshow(uint8(newh1);

title("Image after applying horizontal sobel operator rotated 180 degree"); finalimg=uint8(newv)|uint8(newh)|uint8(newv1)|uint8(newh1); subplot(3,2,6);

imshow(finalimg); title("Final Image");

## output



1. Second Order Derivative – Laplacian Filter

Code: figure;

p=imread('C:\Program Files\scilab-6.0.1\IPCV\images\morpex.png'); subplot(3,2,1);

imshow(p); title('OriginalImage'); d=double(p); m=[010;1-41;010];

[r1,c1]=size(p); For i=2:1:r1-1 for j=2:1:c1-1

newl4(i,j)=(m(1)\*d(i-1,j-1)+(m(2)\*d(i-1,j)+(m(3)\*d(i-1,j+1)+(m(4)\*d(i,j-

1)+(m(5)\*d(i,j)+(m(6)\*d(i,j+1)+(m(7)\*d(i+1,j- 1)+(m(8)\*d(i+1,j)+(m(9)\*d(i+1,j+1); end end

subplot(1,2,2); imshow(uint8(newl4);

title("Image after Laplacian Filtering");

