

*Image Processing*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cour se:*** | *Image Processing* | | |
| ***Topic*** | *Image Processing Basics* | | |

|  |
| --- |
| ***Pixel*** |
| *Program to calculate number of samples required for an image*  ***/****Programtocalculatenumberofsamplesrequiredforanimage*  *figure; m=4; n=6; N=400;*  *Fs=m\*N\*n\*N;*  *disp(Fs,'Number of samples required to preserve the information in the image=');* |

|  |
| --- |
|  |
| ***Image Properties*** |
| *Program to accessimageproperties– Dimension, height, width, number of channels, accessing and modifying anypixel*  */Programtoaccessimageproperties–Dimension,height,width,numberof channels,accessingandmodifyinganypixel*  *figure;*  *i=imread("C:\Program Files\scilab 6.0.2\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));* |

|  |
| --- |
|  |
| ***Sampling*** |
| *Program to study the effects of reducing the spatial resolution of a digital image*  ***/****Programtostudythee****f****ectsofreducingthespatialresolutionofadigitalimage*  *figure;*  *i=imread('C:\Program Files\scilab-6.0.2\IPCV\images\lena\_dark.png'); disp("size of original image",size(i));*  *subplot(3,3,1); imshow(i); title('original image'); j1=imresize(i,0.8);*  *disp("size of resized Image",size(j1)); subplot(3,3,2);*  *imshow(j1);*  *title('resized image 0.8');* |

|  |
| --- |
| *j2=imresize(i,0.5);*  *disp("size of resized Image",size(j2)); subplot(3,3,3);imshow(j2); title('resized image 0.5')* |
| ***Quantization*** |
| *Program to study the effects of varying the number of intensity levels in a digital image*  ***/****Programtostudythee****f****ectsofvaryingthenumberofintensitylevelsinadigital imagefigure;*  *i=imread('C:\Program Files\scilab-6.0.2\IPCV\images\lena\_dark.png'); subplot(2,2,1);*  *imshow(i); title('original image') i=double(i); k1=(i\*255)/64;*  *subplot(2,2,2); ––k1=uint8(k1); imshow(uint8(k1)); title('Quantization 64');*  *k2=(i\*255)/32;* |

|  |
| --- |
| *subplot(2,2,3); k2=uint8(k2); imshow(uint8(k2)); title('Quantization 32'); k3=(i\*255)/16;*  *subplot(2,2,4); k3=uint8(k3); imshow(uint8(k3)); title('Quantization 16');* |
| ***Image Addition*** |
| *Program to perform image addition for noise reduction.*  ***/****Programtoperformimageadditionfornoisereduction.*  *figure;*  *i=imread('C:\Program Files\scilab-6.0.2\IPCV\images\noisy.png'); i=imnoise(i,'salt & pepper',0.001);* |

|  |
| --- |
| *subplot(1,2,1); imshow(i); title('noisy Image'); k=imadd(i,50); –––subplot(1,2,2); imshow(k);*  *title('Smooth Image after performing image addition');* |
| ***Image Subtraction*** |

|  |
| --- |
| *Program to compare images using subtraction for enhancing the difference between images*  ***/****Programtocompareimagesusingsubtractionforenhancingthedi****f****erence betweenimages*  *figure;*  *i=imread('C:\Program Files\scilab-6.0.2\IPCV\images\tool1.png'); j=imread('C:\Program Files\scilab-6.0.2\IPCV\images\tool2.png'); subplot(2,2,1);*  *imshow(i); title('Image 1');*  *subplot(2,2,2); imshow(j); title('Image 2'); k=imabsdiff(i,j); subplot(2,2,3); imshow(k);*  *title('Image 3 = Image 1 - Image 2');* |

|  |
| --- |
|  |



*Image Processing Practical #2*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Enhancement* | | |

|  |
| --- |
| ***Image Negative*** |
| *Code:*  *//ImageNegatvie figure;*  *r=imread('C:\Program Files\scilab-6.0.2\IPCV\images\checkerbox.png'); subplot(2,2,1);*  *imshow(r); title('original Image'); L=2^8;*  *neg=(L-1)-r;*  *subplot(2,2,2); imshow(neg); title('Negative Image');*  *Output: -* |

*Vidyalankar School of Information Technology*

|  |
| --- |
|  |
| ***Thresholding*** |
| *Code:*  *figure;*  *r=imread('C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif'); a=r;*  *[row col]=size(a);*  *T=input('Enter value of threshold:');*  *// valueofthreshold= 70 for i=1:1:row*  *for j=1:1:col if(r(i,j)>=T)*  *a(i,j)=r(i,j); else a(i,j)=0; end*  *end end*  *subplot(1,2,1); imshow(r); title('Original Image'); subplot(1,2,2); imshow(a);*  *title('Image obtained using threshold');* |

|  |
| --- |
| *Output:-* |
| ***Log Transformation*** |

|  |
| --- |
| *Code:- figure;*  *r=imread('C:\Program Files\scilab-6.0.2\IPCV\images\log.tif'); c=15;*  *S=c\*log(1+double(r)); O=uint8(S); subplot(2,2,1); imshow(r) title("Original Image"); subplot(2,2,2); imshow(O);*  *title("Log Transformation") Output:-* |

|  |
| --- |
|  |
| ***Piece-Wise Linear Transformation*** |
| *Code:-*  ***/****powerlawtransformation figure;*  *a = imread('C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif'); subplot(2,2,1);*  *imshow(a); title("Original Image") r=double(a)/255; c=1;*  *gamma=0.3; s=c\*(r).^gamma; subplot(2,2,2); imshow(s); title("Gamma Image")*  *Output:-* |

|  |
| --- |
|  |
| ***Gray Level Slicing*** |
| *With Background Code:-*  ***/****graylevelslicing figure;*  *p=imread('C:\Program Files\scilab-6.0.2\IPCV\images\log.tif'); z=double(p);*  *[row col]=size(p); for i=1:1:row*  *for j=1:1:col if(z(i,j)>50)&&(z(i,j)<200) z(i,j)=150;*  *else z(i,j)=p(i,j); end*  *end end*  *subplot(1,2,1); imshow(p); title('Original Image');*  *subplot(1,2,2); imshow(z);*  *title('Grey level slicing with background'); Output:-* |

|  |
| --- |
| *Without Background Code:-*  ***/****graylevelslicing figure;*  *p=imread('C:\Program Files\scilab-6.0.2\IPCV\images\log.tif'); z=double(p);*  *[row col]=size(p); for i=1:1:row*  *for j=1:1:col if(z(i,j)>50)&&(z(i,j)<100) z(i,j)=200;*  *else z(i,j)=20 end end end*  *subplot(1,2,1); imshow(p); title('Original Image');*  *subplot(1,2,2); imshow(z);*  *title('Grey level slicing without background'); Output:-* |

|  |
| --- |
|  |

|  |
| --- |
|  |
| ***Bit Plane Slicing*** |
| *Code:-*  *figure;*  *B=imread('C:\Program Files\scilab-6.0.2\IPCV\images\coins.png'); [r,c]=size(B);*  *for i=1:r for j=1:c*  *MSB(i,j)=bitand(B(i,j),bin2dec('10000000'));*  *LSB(i,j)=bitand(B(i,j),bin2dec('00000001')); Second(i,j)=bitand(B(i,j),bin2dec('01000000')); Third(i,j)=bitand(B(i,j),bin2dec('00100000')); Fourth(i,j)=bitand(B(i,j),bin2dec('00010000')); Fifth(i,j)=bitand(B(i,j),bin2dec('00001000'));*  *Sixth(i,j)=bitand(B(i,j),bin2dec('00000100')); Seventh(i,j)=bitand(B(i,j),bin2dec('00000010')); end*  *end*  *subplot(4,4,1); imshow(MSB); title("Bit Plane 7") subplot(4,4,2); imshow(Second); title("Bit Plane 6") subplot(4,4,3); imshow(Third); title("Bit Plane 5") subplot(4,4,4); imshow(Fourth); title("Bit Plane 4") subplot(4,4,5); imshow(Fifth); title("Bit Plane 3") subplot(4,4,6); imshow(Sixth); title("Bit Plane 2") subplot(4,4,7); imshow(Seventh); title("Bit Plane 1") subplot(4,4,8); imshow(LSB); title("Bit Plane 0") subplot(4,4,9); imshow(B); title("Original Image")*  *Output:-* |

|  |
| --- |
|  |



*Image Processing*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cour se:*** | *Image Processing* | | |
| ***Topic*** | *Histogram Processing* | | |

|  |
| --- |
| ***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('Original Image') imshow(img)*  *h1=imhist(img); subplot(2,2,2); plot(h1)*  *title('Predefined Histogram,');*  *[r,c]=size(img); h=zeros(1,256); s=0:255*  *for i=1:r for j=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:-* |
| ***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('Original Histogram');* |

|  |
| --- |
| *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 of bright image")*  *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)*  *Output:-* |

|  |
| --- |
|  |
| ***Histogram Equalization*** |

|  |
| --- |
|  |



*Image Processing Practical #4*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Smoothing in Spatial Domain* | | |

|  |
| --- |
| ***Averaging*** |
| *Code:-*  *clc; figure;*  *a=imread("C:\Program Files\scilab-6.0.1\IPCV-4.1.2-win64- bin\IPCV\images\blurry\_moon.tif")*  *subplot(1,3,1) title("Original image") imshow(a)*  *//Adding noise c=imnoise(a,'gaussian') subplot(1,3,2) title("Noisy image") imshow(c)*  *d=double(c); b=d;*  *m=(1/9)\*(ones(3,3));*  *[r1,c1]=size(a); for i=2;r1-1*  *for j=2:c1-1*  *a1=d(i-1,j-1)+d(i-1,j)+d(i-1,j+1)+d(i,j-1)+d(i,j)+d(i,j+1)+d(i+1,j- 1)+d(i+1,j)+d(i+1,j+1);* |

|  |
| --- |
| *b(i,j)=a1\*(1/9); end*  *end subplot(1,3,3)*  *title('Filtered Image') imshow(uint8(b))*  *Output:-* |
| ***Min Filter*** |
| *Code:-*  ***/****min clc; figure;*  *p1=imread("C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif"); subplot(1,3,1)*  *title("Original image") imshow(p1)*  *pmin=imnoise(p1,'salt & pepper',0.2); subplot(1,3,2);* |

|  |
| --- |
| *imshow(pmin); title('Noisy Image');*  *[r1,c1]=size(pmin); for i=2:1:r1-1*  *for j=2:1:c1-1*  *a1=[pmin(i-1,j-1) pmin(i-1,j) pmin(i-1,j+1) pmin(i,j-1) pmin(i,j) pmin(i,j+1) pmin(i+1,j-1) pmin(i+1,j) pmin(i+1,j+1)];*  *a2=min(min(a1)); newmin(i,j)=a2;*  *end end*  *subplot(1,3,3); imshow(newmin);*  *title('Image aftermin filtering'); Output:-* |

|  |
| --- |
|  |
| ***Max Filter*** |

*Code:-*

***/****max clc; figure;*

p1=imread("C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif"); subplot(1,3,1)

title("Original image") imshow(p1)

pmax=imnoise(p1,'salt & pepper',0.03); subplot(1,3,2);

imshow(pmax); title('Noisy Image');

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

for j=2:1:c1-1

a1=[pmax(i-1,j-1) pmax(i-1,j) pmax(i-1,j+1) pmax(i,j-1) pmax(i,j) pmax(i,j+1) pmax(i+1,j-1) pmax(i+1,j) pmax(i+1,j+1)];

a2=max(max(a1)); newmax(i,j)=a2;

end end

subplot(1,3,3); imshow(newmax);

title('Imageaftermaxfiltering'); Output:-

|  |
| --- |
|  |

|  |
| --- |
|  |
| ***Median Filter*** |
| *Code:-*  */med*  *figure;*  *p1=imread("C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif"); pmed=imnoise(p1,'salt & pepper');*  *subplot(1,2,1);*  *imshow(pmed); title('Noisy Image');* |

|  |
| --- |
| *[r1,c1]=size(pmed); for i=2:1:r1-1*  *for j=2:1:c1-1*  *a1=[pmed(i-1,j-1) pmed(i-1,j) pmed(i-1,j+1) pmed(i,j-1) pmed(i,j) pmed(i,j+1) pmed(i+1,j-1) pmed(i+1,j) pmed(i+1,j+1)];*  *a2=gsort(a1); med=a2(5); newmed(i,j)=med; end*  *end subplot(1,2,2); imshow(newmed);*  *title('Imageaftermedfiltering'); Output:-* |



*Image Processing Practical #5*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Sharpening in Spatial Domain* | | |

|  |
| --- |
| ***First Order Derivative Filter*** |
| *Code:-*  *//5-a figure;*  *p=imread("C:\Program Files\scilab-6.0.1\IPCV\images\morpex.png"); subplot(3,2,1);*  *imshow(p); title("Original Image"); d=double(p);*  *v=[1 0 -1;2 0 -2;1 0 -1];*  *h=[-1 -2 -1; 0 0 0 ; 1 2 1];*  *[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=[-1 0 1; -2 0 2; -1 0 1];*  *h1=[1 2 1; 0 0 0; -1 -2 -1];*  *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:-* |

|  |
| --- |
|  |
| ***Second Order Derivative – Laplacian Filter*** |
| *Code:-*  *figure;*  *p=imread('C:\Program Files\scilab-6.0.1\IPCV\images\morpex.png') subplot(1,2,1)*  *title('Original image') imshow(p) d=double(p)*  *v=[0 1 0;1 -4 1;0 1 0];*  *[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(1,2,2) imshow(newv)*  *Output:-* |



*Image Processing Practical #6*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Smoothing in Frequency Domain* | | |

|  |
| --- |
| ***Ideal Low Pass Filter*** |
| *Code:-*  ***/****IdealLowPassFilter figure;*  *p=imread("C:\Program Files\scilab-6.0.1\IPCV\images\cameraman.tif");*  *a1=double(p); r=size(a1,1);*  *c=size(a1,2);*  *d0=input('Enter the cut-off frequency-(Radius):-'); for u=1:1:r*  *for v=1:1:c*  *d=(((u-(r/2))^2)+((v-(c/2))^2))^0.5;*  *if d<=d0 h(u,v)=1;*  *else*  *h(u,v)=0; end*  *end end*  *b=fft2(a1); c=fftshift(b); new=h.\*c;*  *new1=abs(ifft(new)); subplot(1,2,1);* |

|  |
| --- |
| *imshow(uint8(p)); title('Original Image');*  *subplot(1,2,2); imshow(uint8(new1));*  *title(['Filtered Image with radius = ', string(d0)]);*  *Output:-* |
| ***Butterworth Low Pass Filter*** |

|  |
| --- |
| *Code:-*  ***/****Bu****t****erworthlowpassfilter figure;*  *a=imread("C:\Program Files\scilab-6.0.1\IPCV\images\cameraman.tif"); a1=double(a);*  *r=size(a1,1);*  *c=size(a1,2);*  *d0=input('enter the cutt-of frequency-(Radius):-'); n=input("enter exponential")*  *foru=1:1:r forv=1:1:c*  *d=(((u-(r/2))^2)+ ((v-(c/2))^2))^0.5;*  *h(u,v)=1/(1+((d/d0)^(2\*n))); end*  *end b=fft2(a1); c=fftshift(b); new=h.\*c;*  *new1=abs(ifft(new)); subplot(1,2,1); imshow(uint8(a1)); title('original Image'); subplot(1,2,2); imshow(uint8(new1));*  *title(['filtered image with radius=',string(d0)]);* |

|  |
| --- |
|  |
| ***Gaussian Low Pass Filter*** |
| *Code:-*  *a=imread('C:\Program Files\scilab-6.0.1\IPCV\images\cameraman.tif'); a1=double(a);*  *r=size(a1,1);*  *c=size(a1,2);*  *d0=input('Enter the cut-off frequency - (Radius):-'); for u=1:1:r*  *for v=1:1:c*  *d=(((u-(r/2))^2)+((v-(c/2))^2))^0.5*  *h(u,v)=exp(-d\*d/(2\*d0\*d0))*  *end end*  *b=fft2(a1); c=fftshift(b); new=h.\*c;*  *new1=abs(ifft(new)); subplot(1,2,1);*  *imshow(a);* |

|  |
| --- |
| *title('Original Image'); subplot(1,2,2); imshow(uint8(new1));*  *title(['Filtered Image with radius = ',string(d0)]);*  *new1=abs(ifft(new)); subplot(1,2,1); imshow(a); title('Original Image'); subplot(1,2,2); imshow(uint8(new1));*  *title(['Filtered Image with radius = ',string(d0)]);* |



*Image Processing Practical #7*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Sharpening in Frequency Domain* | | |

|  |
| --- |
| ***Ideal High Pass Filter*** |
| *Code:-*  ***/****IdealhighPassFilter figure;*  *p=imread("C:\Program Files\scilab-6.0.1\IPCV\images\cameraman.tif");*  *a1=double(p); r=size(a1,1);*  *c=size(a1,2);*  *d0=input('Enter the cut-off frequency-(Radius):-'); for u=1:1:r*  *for v=1:1:c*  *d=(((u-(r/2))^2)+((v-(c/2))^2))^0.5;*  *if d<=d0 h(u,v)=0;*  *else*  *h(u,v)=1; end*  *end end*  *b=fft2(a1); c=fftshift(b); new=h.\*c;*  *new1=abs(ifft(new)); subplot(1,2,1);* |

|  |
| --- |
| *imshow(uint8(p)); title('Original Image');*  *subplot(1,2,2); imshow(uint8(new1));*  *title(['Filtered Image with radius = ', string(d0)]);*  *Output:-* |
| ***Butterworth High Pass Filter*** |
| *Code:-*  ***/****Bu****t****erworthhighpassfilter* |

|  |
| --- |
| *figure;*  *a=imread("C:\Program Files\scilab-6.0.2\IPCV\images\cameraman.tif"); a1=double(a);*  *r=size(a1,1);*  *c=size(a1,2);*  *d0=input('enter the cutt-of frequency-(Radius):-'); n=input("enter the order of the filter")*  *foru=1:1:r forv=1:1:c*  *d=(((u-(r/2))^2)+ ((v-(c/2))^2))^0.5;*  *h(u,v)=1/(1+((d0/d)^(2^n))); end*  *end b=fft2(a1); c=fftshift(b); new=h.\*c;*  *new1=abs(ifft(new)); subplot(1,2,1); imshow(uint8(a1)); title('original Image'); subplot(1,2,2); imshow(uint8(new1));*  *title(['filtered image with radius=',string(d0)]);*  *Output:-* |

|  |
| --- |
|  |
| ***Gaussian High Pass Filter*** |
| *Code:-*  *figure;*  *a=imread('C:\Program Files\scilab-6.0.2\IPCV\images\cameraman.tif'); a1=double(a);*  *r=size(a1,1);*  *c=size(a1,2);*  *d0=input('Enter the cut-off frequency - (Radius):-'); for u=1:1:r*  *for v=1:1:c*  *d=(((u-(r/2))^2)+((v-(c/2))^2))^0.5*  *h(u,v)=1-exp(-d\*d/(2\*d0\*d0))*  *end end*  *b=fft2(a1); c=fftshift(b); new=h.\*c;*  *new1=abs(ifft(new)); subplot(1,2,1); imshow(a); title('Original Image'); subplot(1,2,2); imshow(uint8(new1));*  *title(['Filtered Image with radius = ',string(d0)]);*  *Output:-* |

|  |
| --- |
|  |



*Image Processing Practical #8*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Restoration* | | |

|  |
| --- |
| ***Arithmetic Mean Filter to Remove Gaussian Noise*** |
| *Code:-*  ***/****arithmetic mean figure;*  *a=imread('C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif');*  *subplot(1,3,1) title("Original image") imshow(a)*  *//Adding noise*  *c=imnoise(a,'gaussian') / c=imnosie(a,g'aussain,'0.5)--Noisewithdenstiy subplot(1,3,2)*  *title("Noisy image") imshow(c) d=double(c);*  *b=d; [r1,c1]=size(a); for i=2;r1-1 for j=2:c1-1*  *a1=d(i-1,j-1)+d(i-1,j)+d(i-1,j+1)+d(i,j-1)+d(i,j)+d(i,j+1)+d(i+1,j-1)+d(i+1,j)+d(i+1,j+1); b(i,j)=a1\*(1/9);*  *//b(i,j)=a1\*(1/9); end*  *end subplot(1,3,3)*  *title('Arithmetic filtered Image')* |

|  |
| --- |
| *imshow(uint8(b))*  *Output:-* |
| ***Geometric Mean Filter to Remove Gaussian Noise*** |
| *Code:-*  ***/****GeometricMean figure;*  *a=imread("C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif"); c=imnoise(a,'gaussian');*  *d=double(c); b=d; //backup*  *[r1,c1]=size(a); for i=2:r1-1*  *for j=2:c1-1*  *a1=d(i-1,j-1)\*d(i-1,j)\*d(i-1,j+1)\*d(i,j-1)\*d(i,j)\*d(i,j+1)\*d(i+1,j)\*d(i+1,j)\*d(i+1,j+1); p=a1^(1/9);*  *b(i,j)=p; end end*  *subplot(1,3,1); imshow(a); title('Original Image');* |

|  |
| --- |
| *subplot(1,3,2); imshow(c); title('noisy image');*  *subplot(1,3,3); imshow(uint8(b)); title('geometric mean filter');*  *Output:-* |
| ***Harmonic Mean Filter to remove Gaussian / Salt and Pepper Noise*** |
| *Code:-*  ***/****harmonic meanwithGaussianNoise figure;*  *a=imread('C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif');*  *subplot(1,3,1) title("Original image") imshow(a)*  *//Adding noise*  *c=imnoise(a,'gaussian') / c=imnoise(a,g'aussian',0.5)-Noisewtihdensity subplot(1,3,2)*  *title("Noisy image") imshow(c) d=double(c);*  *b=d;*  *[r1,c1]=size(a);* |

|  |
| --- |
| *for i=2;r1-1 for j=2:c1-1*  *a1=(1/d(i-1,j-1))+(1/d(i-1,j))+(1/d(i-1,j+1))+(1/d(i,j-1))+(1/d(i,j))+(1/d(i,j+1))+(1/d(i+1,j- 1))+(1/d(i+1,j))+(1/d(i+1,j+1));*  *p=9/a1; b(i,j)=p;*  *end end*  *subplot(1,3,3)*  *title('Filtered Image harmonic Mean Gaussian Noise') imshow(uint8(b))*  *Output:-*    ***/****harmonic meanwithSaltandPepperNoise figure;*  *a=imread('C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.tif');*  *subplot(1,3,1) title("Original image") imshow(a)*  *//Adding noise*  *c=imnoise(a,'salt & pepper')/ c=imnoise(a',gaussain',0.5)-Noisewithdensity subplot(1,3,2)*  *title("Noisy image") imshow(c) d=double(c);*  *b=d; [r1,c1]=size(a);* |

|  |
| --- |
| *for i=2;r1-1 for j=2:c1-1*  *a1=(1/d(i-1,j-1))+(1/d(i-1,j))+(1/d(i-1,j+1))+(1/d(i,j-1))+(1/d(i,j))+(1/d(i,j+1))+(1/d(i+1,j- 1))+(1/d(i+1,j))+(1/d(i+1,j+1));*  *p=9/a1; b(i,j)=p;*  *end end*  *subplot(1,3,3)*  *title('Filtered Imageharmonic Meansaltand Pepper Noise') imshow(uint8(b))*  *Output:-* |
| ***Contra Harmonic Mean Filter to Remove Salt & Pepper Noise*** |
| *Code:-*  ***/****contraharmonicmeantoremovesaltandpeppernoise*  *figure;*  *a=imread("C:\Program Files\scilab-6.0.2\IPCV\images\blurry\_moon.png");*  *Q=input('Q:order of filter'); c=imnoise(a,'salt & pepper',0.002); d=double(c);*  *Q=double(Q);*  *[r1,c1]=size(a); for i=2:r1-1* |

|  |
| --- |
| *for j=2:c1-1*  *N=(d(i-1,j-1)^(Q+1))+(d(i-1,j)^(Q+1))+(d(i-1,j+1)^(Q+1))+(d(i,j-*  *1)^(Q+1))+(d(i,j)^(Q+1))+(d(i,j+1)^(Q+1))+(d(i+1,j-*  *1)^(Q+1))+(d(i+1,j)^(Q+1))+(d(i+1,j+1)^(Q+1));*  *D=(d(i-1,j-1)^(Q))+(d(i-1,j)^(Q))+(d(i-1,j+1)^(Q))+(d(i,j-*  *1)^(Q))+(d(i,j)^(Q))+(d(i,j+1)^(Q))+(d(i+1,j-1)^(Q))+(d(i+1,j)^(Q))+(d(i+1,j+1)^(Q));*  *b(i,j)=N/D; end*  *end subplot(1,3,1); imshow(a);*  *title('Original Image'); subplot(1,3,2); imshow(c); title('Noisy Image');*  *subplot(1,3,3); imshow(uint8(b));*  *title('Contra Harmonic Mean Filtered Image');*  *Output:- Q=2*    *Q=-2* |

|  |
| --- |
| *n*  *(if Q is positive then black dots remove and if Q is negative then white dots remove)* |
| ***Mid Point Filter to Remove Salt & Pepper Noise*** |
| *Code:-*  *figure;*  *img=imread('C:\Program Files\scilab-6.0.1\IPCV\images\moon.tif'); c=imnoise(img,"gaussian",0.002)*  *d=double(c);*  *//b=d;*  *//m=(1/9)\*(ones(3,3));*  *[r1,c1]=size(c); for i=2:1:r1-1 for j=2:1:c1-1*  *a1=[d(i-1,j-1) d(i-1,j) d(i-1,j+1) d(i,j-1) d(i,j) d(i,j+1) d(i+1,j-1) d(i+1,j) d(i+1,j+1)];* |



|  |
| --- |
| *amin=min(min(a1)); amax=max(max(a1)); newmin(i,j)=double((amin+amax)/2); end*  *end subplot(2,2,1); imshow(img);*  *title("Original Image"); subplot(2,2,2); imshow(c)*  *title("After adding Salt and Pepper Noise in Original Image") subplot(2,2,3);*  *imshow(uint8(newmin)); title("FilteredImageusing Min Filter")*  *Output:-* |

|  |
| --- |
|  |



*Image Processing Practical #9*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Color Image Processing* | | |

|  |
| --- |
| ***Display different color planes and plot histogram of color image*** |
| *Code:-*  *clc; clear all;*  *a=imread("C:\Program Files\scilab-6.0.2\IPCV\images\peppers.png"); R=a(:,:,1);*  *G=a(:,:,2);*  *B=a(:,:,3);*  *subplot(2,2,1); imshow(a); title("Original Image"); subplot(2,2,2); imshow(R);*  *title("Red Component Image"); subplot(2,2,3);*  *imshow(G);*  *title("Green ComponentImage"); subplot(2,2,4);*  *imshow(B);*  *title("Blue Component Image");*  *figure;*  *//GethistValues for eachchannel [yR,x]= imhist(R);* |

|  |
| --- |
| *[yG,x]= imhist(G);*  *[yB,x]= imhist(B);*  *//Plot them together in one plotx plot(x,yR,x,yG,x,yB,"LineWidth",2); xlabel("RGB Intensity"); ylabel("No of Pixels"); set(gca(),"grid",[1,1])*  *Output:-* |

|  |
| --- |
|  |
| ***Convert RGB image to HSV and CMY model*** |
| *Code:-*  *figure;*  *a=imread("C:\Program Files\scilab-6.0.1\IPCV\images\peppers.png"); subplot(2,2,1);*  *title("Original Image") imshow(a); HSV=rgb2hsv(a); subplot(2,2,2); title("RGB2HSV");*  *imshow(HSV); CMY=imcomplement(a); subplot(2,2,3); title("RGB2CMY") imshow(CMY);*  *Output:-* |

|  |
| --- |
|  |
| ***Color Image Smoothing*** |
| *Code:-*  *//colorimagesmoothing*  *clc;*  *clear all; figure;* |

|  |
| --- |
| *I=imread("C:\Program Files\scilab-6.0.1\IPCV\images\peppers.png"); F = fspecial('average',11);*  *pic(:,:,1) = imfilter(I(:,:,1),F);*  *pic(:,:,2) = imfilter(I(:,:,2),F);*  *pic(:,:,3) = imfilter(I(:,:,3),F); Average\_Image = uint8(pic); subplot(2,2,1);*  *imshow(I); title('Original Image'); subplot(2,2,2);*  *imshow(Average\_Image); title('Average Filtered Image'); I=imnoise(I,'gaussian',0.07); F1 = fspecial('gaussian',3); pic1(:,:,1) =imfilter(I(:,:,1),F1);*  *pic1(:,:,2) = imfilter(I(:,:,2),F1);*  *pic1(:,:,3) = imfilter(I(:,:,3),F1); GLPF\_Image = uint8(pic1);*  *subplot(2,2,3);imshow(I);title('Noisy Image(Gaussian Noise)'); subplot(2,2,4);imshow(GLPF\_Image);title('GLPF Filtered Image')*  *Output:-* |

|  |
| --- |
|  |
| ***Color Image Sharpening*** |
| *Code:-*  ***/****colorimagesharpening clc;*  *clear all; figure;*  *I=imread("C:\Program Files\scilab-6.0.1\IPCV\images\peppers.png"); F=fspecial('prewitt');*  *pic(:,:,1) = imfilter(I(:,:,1),F);*  *pic(:,:,2) = imfilter(I(:,:,2),F);*  *pic(:,:,3) = imfilter(I(:,:,3),F); Sharp\_Image = uint8(pic);*  *subplot(1,2,1);imshow(I);title('Original Image'); subplot(1,2,2);*  *imshow(Sharp\_Image); title('Sharp Filtered Image')*  *Output:-* |

|  |
| --- |
|  |



*Image Processing Practical #10*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Morphological Image Processing* | | |

|  |
| --- |
| ***Erosion*** |
| *Code:-*  ***/****Programtoapplyerosion clc;*  *clear all; figure;*  *a=imread('C:\Program Files\scilab-6.0.1\IPCV\images\morpex.png'); d=a;*  *[r,c]=size(d); m=ones(3,3);*  *for i=2:1:r-1 for j=2:1:c-1*  *new=[(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))]; A1(i,j)=min(new);*  *end end*  *subplot(1,2,1); imshow(a); title('Original Image');*  *subplot(1,2,2);* |

|  |
| --- |
| *imshow(A1); title('Erosion Image');*  *Output:-* |
| ***dilation*** |
| *Code:-*  ***/****Programtoapplydilation clc;*  *clear all; figure;*  *a=imread('C:\Program Files\scilab-6.0.1\IPCV\images\morpex.png'); d=a;*  *[r,c]=size(d); m=ones(3,3);*  *for i=2:1:r-1 for j=2:1:c-1*  *new=[(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))];* |

|  |
| --- |
| *A1(i,j)=max(new); end*  *end subplot(1,2,1); imshow(a);*  *title('Original Image');*  *subplot(1,2,2); imshow(A1); title('Dilation Image');*  *Output:-* |
| ***Opening*** |
| *Code:-*  *figure;*  *a=imread('C:\Program Files\scilab-6.0.1\IPCV\images\line.tif'); d=a;*  *[r,c]=size(d); m=ones(3,3);*  *fori=2:1:r-1 for j=2:1:c-1*  *new=[(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))]; A1(i,j)=min(new);*  *end end d=A1;*  *[r,c]=size(d);*  *fori=2:1:r-1 for j=2:1:c-1*  *new=[(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))]; A2(i,j)=max(new);*  *end end*  *subplot(1,2,1); imshow(a); title('Original Image'); subplot(1,2,2); imshow(A2);*  *title('processed image - o'); Output:-* |
| ***Closing*** |
| *Code:-* |

|  |
| --- |
| ***/****ProgramtoapplyProcessedImage-Closing clc;*  *clear all; figure;*  *a=imread('C:\Program Files\scilab-6.0.1\IPCV\images\circbw.png'); d=a;*  *[r,c]=size(d); m=ones(3,3);*  *fori=2:1:r-1 for j=2:1:c-1*  *new=[(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))]; A1(i,j)=max(new);*  *end end d=A1;*  *[r,c]=size(d);*  *fori=2:1:r-1 for j=2:1:c-1*  *new=[(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))]; A2(i,j)=min(new);*  *end end*  *subplot(1,2,1); imshow(a); title('Original Image'); subplot(1,2,2); imshow(A2);*  *title('processed image - o');*  *Output:-* |

|  |
| --- |
|  |
| ***Boundary Extraction*** |
| *Code:-*  *Boundary*  *clc; clear*  *a= imread('C:\Program Files\scilab-6.0.1\IPCV\images\boundary.tif'); d=a;*  *[r,c]=size(d);*  *m= ones(3,3)* |

*Vidyalankar School of Information Technology*

|  |
| --- |
| *fori=2:1:r-1 for j=2:1:c-1*  *new =[(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))]; A1(i,j)=min(new);*  *aa(i,j)= d(i,j) - A1(i,j) end*  *end*  *subplot(1,2,1) imshow(a); title("Original Image")*  *subplot(1,2,2) imshow(aa)*  *title("Processed Image Boundary Extraction")* |

|  |
| --- |
|  |
| ***Morphological Gradient*** |
| *Code:-*  *clc;*  *Clear*  *a= imread('C:\Program Files\scilab-6.0.1\IPCV\images\boundary.tif'); d=a;*  *[r,c]=size(d); m= ones(3,3) for i=2:1:r-1 for j=2:1:c-1*  *new =[(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))] A1(i,j)=max(new);*  *End end d=A1;*  *[r,c]= size(d)*  *for i=2:1:r-1* |

|  |
| --- |
| *for j=2:1:c-1*  *new =[(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))] A2(i,j)=min(new);*  *aa(i,j)= A1(i,j)-A2(i,j) end*  *end subplot(1,2,1) imshow(a);*  *title("Original Image") subplot(1,2,2) imshow(aa)*  *title("Processed Image Gradient")* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Image Processing Practical #10* | | | | |
|  | ***Name*** | *GORIMA SAYED* | ***Roll Number*** | *21306A1054* |
| ***Subject/Cours e:*** | *Image Processing* | | |
| ***Topic*** | *Image Segmentation* | | |
|  | | | | |
|  | ***Edge Detection –Sobel operator*** | | | |





# **/**Firstorderderivativefilter-SobelOperator

***/****prewi****t*** *clc; clear all;*

*p=imread('C:\Program Files\scilab-6.0.1\IPCV\images\edge2.tif') figure;*

*subplot(2,2,1); imshow(p); title('Original Image');*

*;*

*v1=[1,0,-1;2, 0, -2;1, 0, -1];* ***/****x-direction*

*h1=[-1, -2, -1; 0, 0 ,0; 1, 2, 1 ]; //y-direction*

*v2=[-1, 0, 1; -2, 0, 2; -1, 0, 1];*

*h2=[1, 2, 1; 0, 0, 0; -1, -2 ,-1];*

*gv1=abs(imfilter(double(p),v1));*

# /subplot(3,2,2);

***/****imshow(gv1);*

***/****title(V' ericalEdges)'; gh2=abs(imfilter(double(p),h2)); gh1=abs(imfilter(double(p),h1)); subplot(2,2,2);*

*imshow(gh1);*

*Vidyalankar School of Information Technology*

*title('horizontal Edges'); gv2=abs(imfilter(double(p),v2)); subplot(2,2,3);*

*imshow(gv2); title('vertical2 Edges');*

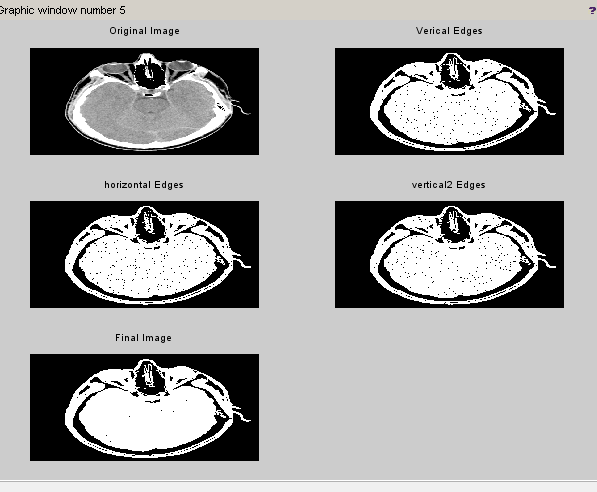
*finaledge=gh1+gv2+gh2+gv1; subplot(2,2,4); imshow(finaledge);*

*title('Final Image');*

|  |  |
| --- | --- |
|  |  |
| ***Edge Detection –Canny Algorithm*** |
| *clc; clear all;*  *p=imread('C:\Program Files\scilab-6.0.1\IPCV\images\edge2.tif') figure;*  *subplot(2,2,1); imshow(p); title('Original Image'); d=double(p); thresh=0.02;* |

|  |  |
| --- | --- |
|  | *sigma=3; / 3,5,7*  *E=edge(d, 'canny', thresh, sigma); subplot(2,2,2);*  *imshow(E);*  *title('Canny Edge Image');*    ***/****Firstorderderivativefilter-SobelOperator*  ***/****prewi****t*** *clc; clear all;*  *p=imread('C:\Program Files\scilab-6.0.1\IPCV\images\edge2.tif'); figure;*  *subplot(3,2,1); imshow(p); title('Original Image');*  *;*  *v1=[1,0,-1;2, 0, -2;1, 0, -1]; //x-direction*  *h1=[-1, -2, -1; 0, 0 ,0; 1, 2, 1 ]; //y-direction* |

v2=[-1, 0, 1; -2, 0, 2; -1, 0, 1];



h2=[1, 2, 1; 0, 0, 0; -1, -2 ,-1];

gv1=abs(imfilter(double(p),v1)); subplot(3,2,2);

imshow(gv1); title('Verical Edges');

gh2=abs(imfilter(double(p),h2)); gh1=abs(imfilter(double(p),h1)); subplot(3,2,3);

imshow(gh1); title('horizontal Edges');

gv2=abs(imfilter(double(p),v2)); subplot(3,2,4);

imshow(gv2); title('vertical2 Edges');

finaledge=gh1+gv2+gh2+gv1; subplot(3,2,5); imshow(finaledge);

title('Final Image');

*Vidyalankar School of Information Technology*

*Vidyalankar School of Information Technology*