1. Homomorphic filtering

clc;

close all;

clear all;

r=imread('CT Brain.jpg');

b=im2double(r);

I=log(1+b);

[m,n]=size(I);

for i=1:m

for j=1:n

p(i,j)=((-1)^(i+j))\*I(i,j);

end

end

bf=fft2(p);

for u=1:m

for v=1:n

d=((u-m/2)^2+(v-n/2)^2)^(1/2);

H(u,v)=1-exp(-d^2/100);

end

end

G=bf.\*H;

g=real(ifft2(G));

for i=1:m

for j=1:n

g1(i,j)=((-1)^(i+j))\*g(i,j);

end

end

g2=(exp(g1)-1);

subplot(211);

imshow(b);

subplot(212);

imshow(g2,[]);

1. Image smoothening using Mean filter

clc; clear all; close all;

pkg load image

b=imread('C:\Users\admin\Desktop\Medical Image\MRI T2 Brain.jpg');

a1=imnoise(b,'gaussian');

a2=imnoise(b,'salt & pepper');

a3=imnoise(b,'gaussian',0.2,0.8);

m1=fspecial('average',[3,3]);

m2=fspecial('average',[5,5]);

o1=imfilter(a1,m1);

o2=imfilter(a2,m1);

o3=imfilter(a3,m1);

o4=imfilter(a1,m2);

o5=imfilter(a2,m2);

o6=imfilter(a3,m2);

figure(1)

subplot(3,3,1), imshow(a1); title("Gaussian noise");

subplot(3,3,2), imshow(a2); title("Salt and pepper noise");

subplot(3,3,3), imshow(a3); title("Gaussian noise with mean and variance");

subplot(3,3,4), imshow(o1); title("Gaussian with 3x3 average");

subplot(3,3,5), imshow(o2); title("salt and pepper with 3x3average");

subplot(3,3,6), imshow(o3); title("Gaussian with Mean & Variance with 3x3average");

subplot(3,3,7), imshow(o4); title("Gaussian with 5x5average");

subplot(3,3,8), imshow(o5); title("S&P with 5x5 average");

subplot(3,3,9), imshow(o6); title("Gaussian Mean & Var with 5x5average");

1. Image smoothening using Median filter

clc; clear all; close all;

pkg load image

b=imread('C:\Users\admin\Desktop\Medical Image\MRI T2 Brain.jpeg');

a1=imnoise(b,'gaussian');

a2=imnoise(b,'salt & pepper');

a3=imnoise(b,'gaussian',0.2,0.8);

m1=fspecial('average',[3,3]);

m2=fspecial('average',[5,5]);

o1=medfilt2(a1,[3,3]);

o2=medfilt2(a2,[3,3]);

o3=medfilt2(a3,[3,3]);

o4=medfilt2(a1,[5,5]);

o5=medfilt2(a2,[5,5]);

o6=medfilt2(a3,[5,5]);

figure(1)

subplot(2,3,1), imshow(o1); title("Gaussian with 3x3 median");

subplot(2,3,2), imshow(o2); title("Salt and pepper with 3x3 median");

subplot(2,3,3), imshow(o3); title("Gaussian with mean and variance with 3x3 median");

subplot(2,3,4), imshow(o4); title("Gaussian with 5x5 median");

subplot(2,3,5), imshow(o5); title("salt and pepper with 5x5 median ");

subplot(2,3,6), imshow(o6); title("Gaussian with Mean & Variance with 5x5 median");

1. Peak to signal noise ratio

clc;

clear all;

close all;

pkg load image

b=imread('C:\Users\admin\Desktop\image BIPL\CT Brain.jpg');

a1=imnoise(b,'gaussian');

a2=imnoise(b,'salt & pepper');

a3=imnoise(b,'gaussian',0.2,0.8);

m1=fspecial('average',[3,3]);

m2=fspecial('average',[5,5]);

o1=imfilter(a1,m1);

o2=imfilter(a2,m1);

o3=imfilter(a3,m1);

o4=imfilter(a1,m2);

o5=imfilter(a2,m2);

o6=imfilter(a3,m2);

[p1,s1]=psnr(o1,a1);

[p2,s2]=psnr(o2,a1);

[p3,s3]=psnr(o3,a2);

[p4,s4]=psnr(o4,a2);

1. Fusion technique

clc;

clear all;

close all;

pkg load image;

I1=double(imread('C:\Users\admin\Desktop\resized img\CT1.jpg'));

if size(I1,3)==3

I1=rgb2gray(I1);

end

I2=double(imread('C:\Users\admin\Desktop\resized img\mri1.png'));

if size(I2,3)==3

I2=rgb2gray(I2);

end

for i=1:size(I1,1);

for j=1:size(I1,2);

I(i,j)=max(I1(i,j),I2(i,j));

end

end

figure(1)

subplot(1,3,1);

imshow(I1,[]);

title('Image 1');

subplot(1,3,2);

imshow(I2,[]);

title('Image 2');

subplot(1,3,3);

imshow(I,[]);

title('Fused Image maximum fusion rule');

if size(I2,3)==3

I2=rgb2gray(I2);

end

for i=1:size(I1,1);

for j=1:size(I1,2);

I(i,j)=min(I1(i,j),I2(i,j));

end

end

figure(2)

subplot(1,3,1);

imshow(I1,[]);

title('Image 1');

subplot(1,3,2);

imshow(I2,[]);

title('Image 2');

subplot(1,3,3);

imshow(I,[]);

title('Fused Image minimum fusion rule');

1. Region growing

clc; clear all; close all;

pkg load image

%function(g,NR,SI,TI)=regiongrow(f,S,T)

%NR-no of regions

%SI-final seed

%TI-threshold image

f=imread('C:\Users\admin\Desktop\Medical Image\CT Brain.jpg');

f=double(f);

S=input('input value of S: ');

T=input('input value of T: ');

%if S is scalar, obtain seed image

if numel(S)==1

SI=f==S;

S1=S;

else

SI=bwmorph(S,'shrink',Inf);

J=find(SI);

S1=f(J); %array of seed value

end

TI=false(size(f));

for K=1:length(S1)

seedvalue=S1(K);

S=abs(f-seedvalue)<=T;

TI=TI|S;

end

figure;

subplot(2,2,1);

imshow(SI);

subplot(2,2,2);

imshow(TI);

[g,NR]=bwlabel(imreconstruct(SI,TI));

subplot(2,2,3);

imshow(g)

1. Gradient filters
2. Prewitt filter

clc;

clear all;

close all;

pkg load image

a=imread('C:\Users\admin\Desktop\Medical Image\ultrasound\_liver.jpg');

b1=[-1 -1 -1 :0 0 0:1 1 1];

b2=[-1 -1 0:1 0 1: 0 1 1];

b3=[-1 0 1:-1 0 1:-1 0 1];

b4=[0 1 1:-1 0 1: -1 0 1];

b5=[1 1 1:0 0 0:-1 -1 -1];

b6=[1 1 0: 1 0 -1:0 -1 -1];

b7=[1 0 -1:1 0 -1:1 0 -1];

b8=[0 -1 -1:1 0 -1: 1 1 0];

a1=imfilter(a,b1);

a2=imfilter(a,b2);

a3=imfilter(a,b3);

a4=imfilter(a,b4);

a5=imfilter(a,b5);

a6=imfilter(a,b6);

a7=imfilter(a,b7);

a8=imfilter(a,b8);

figure(1)

imshow(a);

title("original image");

figure(2);

subplot(2,2,1); imshow(a1); title("prewitt operator(H-line) o/p");

subplot(2,2,2); imshow(a2); title("prewitt operator(45 degree) o/p");

subplot(2,2,3); imshow(a3); title("prewitt operator(v-line) o/p");

subplot(2,2,4); imshow(a4); title("prewitt operator(-45 degree) o/p");

figure(3)

subplot(2,2,1); imshow(a5); title("prewitt operator(H-line) o/p");

subplot(2,2,2); imshow(a6); title("prewitt operator(45 degree) o/p");

subplot(2,2,3); imshow(a7); title("prewitt operator(v-line) o/p");

subplot(2,2,4); imshow(a8); title("prewitt operator(-45 degree) o/p");

1. Sobel filter

clc;

clear all;

close all;

pkg load image

a=imread('C:\Users\admin\Desktop\Medical Image\ultrasound\_liver.jpg');

b1=[-1 -2 -1: 0 0 0: 1 2 1];

b2=[-2 -1 0: 1 0 1: 0 1 2];

b3=[-1 0 1: -2 0 2: -1 0 1];

b4=[0 1 2 : -1 0 1: -2 -1 1];

b5=[1 2 1: 0 0 0: -1 -2 -1];

b6=[2 1 0: 1 0 -1 : 0 -1 -2];

b7=[1 0 -1 : 2 0 -2: 1 0 -1];

b8=[0 -1 -2:1 0 -1:2 1 0];

a1=imfilter(a,b1);

a2=imfilter(a,b2);

a3=imfilter(a,b3);

a4=imfilter(a,b4);

a5=imfilter(a,b5);

a6=imfilter(a,b6);

a7=imfilter(a,b7);

a8=imfilter(a,b8);

figure(1)

imshow(a);

title("original image");

figure(2);

subplot(2,2,1); imshow(a1); title("sobel operator(H-line) o/p");

subplot(2,2,2); imshow(a2); title("sobel operator(45 degree) o/p");

subplot(2,2,3); imshow(a3); title("sobel operator(v-line) o/p");

subplot(2,2,4); imshow(a4); title("sobel operator(-45 degree) o/p");

figure(3)

subplot(2,2,1); imshow(a5); title("sobel operator(H-line) o/p");

subplot(2,2,2); imshow(a6); title("sobel operator(45 degree) o/p");

subplot(2,2,3); imshow(a7); title("sobel operator(v-line) o/p");

subplot(2,2,4); imshow(a8); title("sobel operator(-45 degree) o/p");

(iii) Robert filter

Clc; close all; clear all;

a=imread(“cameraman.tiff”);

b1=[1 0: 0 -1];

b2=[0 1: -1 0];

a1=imfilter(a,b1);

a2=imfilter(a,b2);

subplot(2,2,1);

imshow(a);

title(‘original image’);

figure(2)

subplot(2,2,2);

imshow(a1);

title(‘Gx’);

subplot(2,2,3);

imshow(a2);

title(‘Gy’);

1. Segmentation using Thresholding

clc;

clear all;

close all;

pkg load image;

I1=double(imread('C:\Users\admin\Desktop\Medical Image\MRI Brain T1(image2).png'));

if size(I1,3)==3

I1=rgb2gray(I1);

end

figure(1)

imshow(I1,[]);

figure(2)

hist(I1);

[m n]=size(I1);

for i=1:m

for j=1:n

if I1(i,j)>150

I2(i,j)=I1(i,j);

else

I2(m,n)=0;

end

end

end

figure(3);

imshow(I2,[]);

1. Histogram Equalisation

Clc;close all;clear all;

a=imread(‘CT Brain.jpg’);

b=histeq(a);

subplot(2,2,1);

imshow(a);

title(‘original image’);

subplot(2,2,2);

imshow(b);

title(‘after histogram equalisation’);

subplot(2,2,3);

imshow(a);

title(‘original histogram’);

subplot(2,2,4);

imshow(b);

title(‘after histogram equalisation’);

1. 2D convolution

clc;

clear all;

close all;

f=imread(‘ultrasound\_liver.jpg’);

g1=imcomlement(f);

g2=imcomlement(f);

g3=ones(size(f));

o1=conv2(f,g1,’same’);

o2=conv2(f,g2,’same’);

o1=conv2(f,g3,’same’);

subplot(4,3,1);

imshow(f);

title(‘input image’);

subplot(4,3,2);

imshow(g1);

title(‘mask 1’);

subplot(4,3,3);

imshow(g2);

title(‘mask 2’);

subplot(4,3,4);

imshow(g3);

title(‘mask 3’);

subplot(4,3,5);

imshow(o1);

title(‘o1=f\*g1’);

subplot(4,3,6);

imshow(o2);

title(‘o2=f\*g2’);

subplot(4,3,7);

imshow(o3);

title(‘o3=f\*g3’);

subplot(4,3,8);

imhist(f);

title(‘histogram of f’);

subplot(4,3,9);

imhist(o1);

title(‘histogram of o1’);

subplot(4,3,10);

imhist(o2);

title(‘histogram of o2’);

subplot(4,3,11);

imhist(o3);

title(‘histogram of o3’);

mean of input=mean2(f);

mean of o1=mean2(o1);

mean of o2=mean2(o2);

mean of o3=mean2(o3);

1. Moving Average Filter

clc;

close all;

clear all;

pkg load image;

a2=imread('C:\Users\admin\OneDrive\Desktop\Medical Image\CT Brain.jpg');

a=imnoise(a2,'salt and pepper',0.03);

b=im2double(a)

[m,n]=size(b);

for i=2:m-1

for j=2:n-1

c=[b(i-1,j-1),b(i-1,j),b(i-1,j+1);

b(i,j-1),b(i,j),b(i,j+1);

b(i+1,j-1),b(i+1,j),b(i+1,j+1)];

o1=sum(sum(c));

o=1/9\*o1;

output(i,j)=o;

end

end

subplot(1,2,1);

imshow(a);

subplot(1,2,2);

imshow(output);

1. Hit or Miss Transformation

clc;

clear;

close all;

pkg load image;

a1=[0 0 0 0 0 0 0 0 0 0;

0 1 1 1 0 1 1 1 0 0;

0 1 1 1 0 1 1 0 0 0;

0 1 1 1 0 1 1 1 0 0;

0 0 0 0 0 0 0 0 0 0];

a2=imcomplement(a1);

b1=[0 1 0;1 1 1;0 1 0];

o2=imcomplement(b1);

o1=imerode(a1,b1);

subplot(2,2,1);

imshow(a1);

subplot(2,2,2);

imshow(o1);

o2=bwhitmiss(a1,b1,o2);

subplot(2,2,3);

imshow(o2);

1. Fourier Boundary Descriptor

Clc; clear all; close all;

a=[1+i, 2+i, 3+i, 4+i, 5+i, 5+2i, 5+3i, 4+3i, 3+3i, 2+3i,1+3i,1+2i,1+i];

subplot(2,2,1);

plot(a);

b=fft(a,12);

k1=2;

c1=b(1:k1);

o1=ifft(c1,12);

subplot(2,2,2);

plot(o1);

k2=6;

c2=b(1:k2);

o2=ifft(c2,12);

subplot(2,2,3);

plot(o2);

k2=10;

c3=b(1:k3);

o3=ifft(c3,12);

subplot(2,2,4);

plot(o3);

1. Mean and Median Filter

Clc ; clear all; close all;

i=imread(‘ultrasound fetus.jpg’);

subplot(2,2,1);

imshow(i);

title(‘original image’);

n=imnoise(i,’salt & pepper’);

subplot(2,2,2);

imshow(n);

title(‘noisy image’);

f1=fspecial(‘average’);

Ip=imfilter(n,f1);

subplot(2,2,3);

imshow(Ip);

title(‘Mean filtered image’);

m=medfilt2(n);

subplot(2,2,4);

imshow(m);

title(‘Median filtered image’);

1. Intensity Level Slicing

Clc ;clear all; close all;

a=input(‘enter the value of A: ‘);

b=input(‘enter the value of B: ‘);

figure(‘Name’, ’Intensity Level Slicing’);

i1=imread(‘ultrasound fetus.jpg’);

subplot(1,3,1);

imshow(i);

title(‘input image’);

[m,n]=size(i1);

for i=1:m

for j=1:n

if i1(i,j)>a && i1(i,j)<=b

i1(i,j)=255;

end

end

end

subplot(1,3,2);

imshow(i1);

title(‘without background’);

for i=1:m

for j=1:n

if i1(i,j)>a && i1(i,j)<=b

i1(i,j)=255;

else

i2(i,j)=i1(i,j);

end

end

end

subplot(1,3,3);

imshow(i11);

title(‘with background’);

1. High and Low Pass Filters

Clc ; clear all; close all;

i=imread(‘ultrasound fetus.jpg’);

subplot(1,3,1);

imshow(i);

title(‘original image’);

f1=fspecial(‘average’);

Ip=imfilter(i,f1);

subplot(1,3,2);

imshow(Ip);

title(‘Low pass filtered image’);

f2=fspecial(‘unsharp’);

Hp=imfilter(i,f2);

subplot(1,3,3);

imshow(Hp);

title(‘High pass filtered image’);

1. Order Filter

Clc ; clear all ; close all;

a=imread(‘CT Brain.jpg’);

o1=ordfilt2(a,1,ones(3,3));

o2=ordfilt2(a,9,ones(3,3));

o3=ordfilt2(a,5,ones(3,3));

subplot(1,4,1);

imshow(a);

title(‘original image’);

subplot(1,4,2);

imshow(o1);

title(‘Min filter image’);

subplot(1,4,3);

imshow(o2);

title(‘Max filter image’);

subplot(1,4,4);

imshow(o3);

title(‘Median filter image’);