Image Processing-2

Subhajit Sahu

110EC0181

**Code**

clc;

close all;

im = (imread('tire.tif'));

[m n]=size(im);

im2=zeros(2\*m,2\*n);

for i=1:m

for j=1:n

im2(i,j)=im2(i,j)+im(i,j);

end

end

[r c]=size(im2);

[X,Y] = meshgrid(-n:n-1,-m:m-1);

Z = 100\*sin(2\*pi\*50\*X/(2\*m)+2\*pi\*40\*Y/(2\*n) )+ 100\*sin(2\*pi\*150\*X/(2\*m)+2\*pi\*120\*Y/(2\*n) );

imn = im2 + Z;

figure,

subplot(2,4,1);

imshow(im,[]);

title('original image');

subplot(2,4,2);

imshow(Z);

title('periodic sinusoidal noise');

subplot(2,4,3);

imshow(im2,[]);

title('padded image');

subplot(2,4,4);

imshow(log(1+abs(fftshift(fft2(im2)))),[]);

title('fft of padded image');

subplot(2,4,5);

imshow(imn,[]);

title('padded noisy image');

Fimn = abs(fftshift(fft2(imn))); %----------fftshift for centralizing the frequency and log for contrast stretching------%

logFimn = log(1+Fimn);

subplot(2,4,6);

imshow(logFimn,[]);

title('fft of the noisy image');

maxval = max(logFimn(:));

[yy,xx] = find(logFimn >= floor(maxval));

% ideal notch filter which allows all frequencies except the selected frequencies

H = ones(2\*m,2\*n);

for i=1:length(yy)

for j=1:length(xx)

H(yy(i)-1:yy(i)+1,xx(i)-1:xx(i)+1) = 0; %Notch Reject%

H1=1-H; %Notch pass%

end;

end

Fim\_restored = H.\*fftshift(fft2(imn));

im\_restored = abs(ifft2(ifftshift(Fim\_restored)));

subplot(2,4,7);

imshow(im\_restored,[]);

title('notchpass filter restored image');

subplot(2,4,8);

I=fftshift(fft2(im\_restored));

imshow(log(1+abs(I)),[]);

title('fft of the restored image');

m1(:,:)=mean(Z(:));

m2(:,:)=mean(imn(:));

m3(:,:)=mean(imn(:).\*Z(:));

m4(:,:)=mean(imn(:).\*imn(:));

m5(:,:)=m1.^2;

w(:,:)=(m3-m1\*m2)./(m4-m5)

f=imn(:,:)-w.\*Z(:,:);

figure;

subplot(2,2,1);

imshow(im2,[])

title('padded image');

subplot(2,2,2);

imshow(Z,[])

title('periodic sinusoidal noise');

subplot(2,2,3);

imshow(imn,[]);

title('noisy image');

subplot(2,2,4);

imshow(f,[]);

title('noisy image after optimum filtering');

figure;

surf(H);

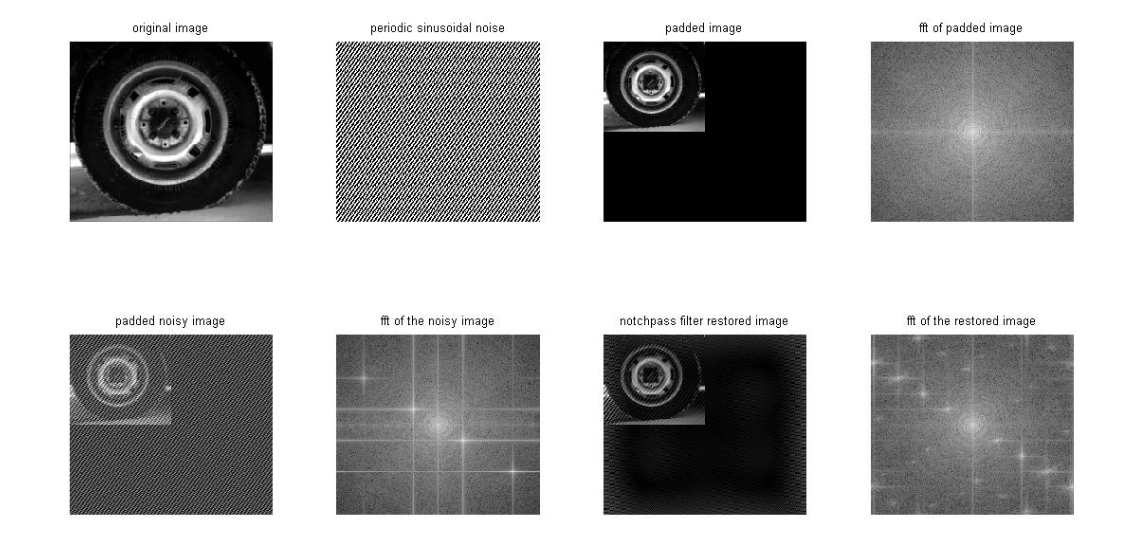
title('notch reject filter in 3D');

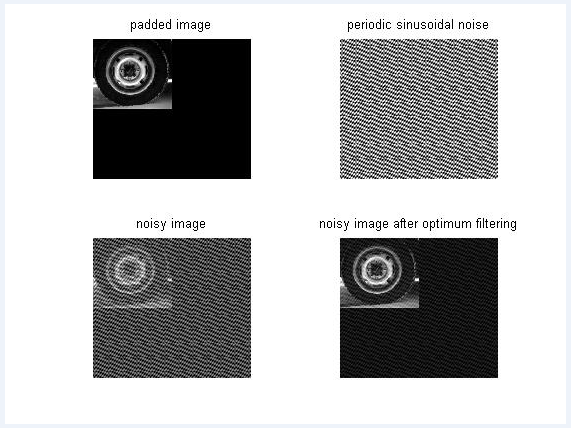
figure;

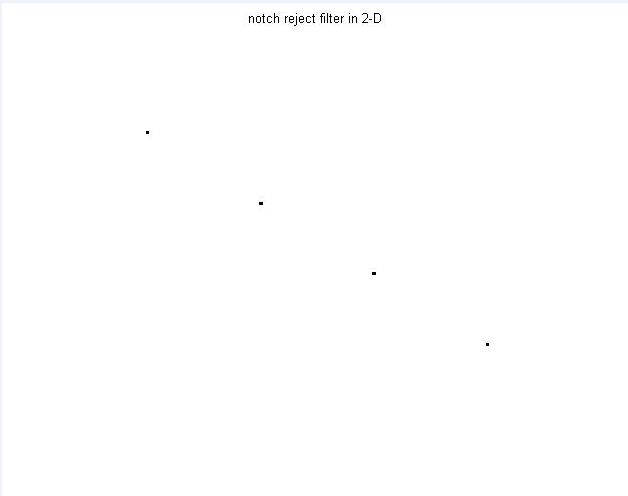
imshow(H);

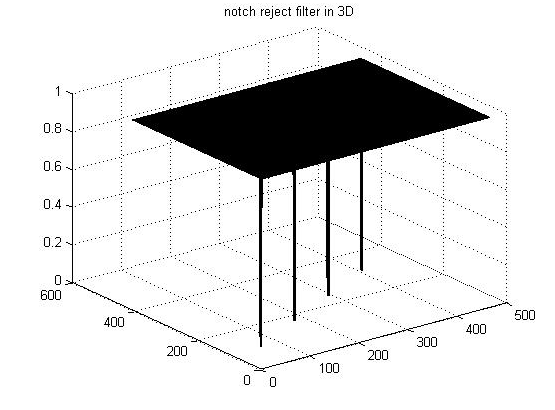
title('notch reject filter in 2-D');

Output-------‘tire.tif’









Z = 100\*sin(2\*pi\*50\*X/512+2\*pi\*40\*Y/512 )+ 100\*sin(2\*pi\*150\*X/512 + 2\*pi\*120\*Y/512)

W=0.8165-------(Optimum Filter)

Discussion

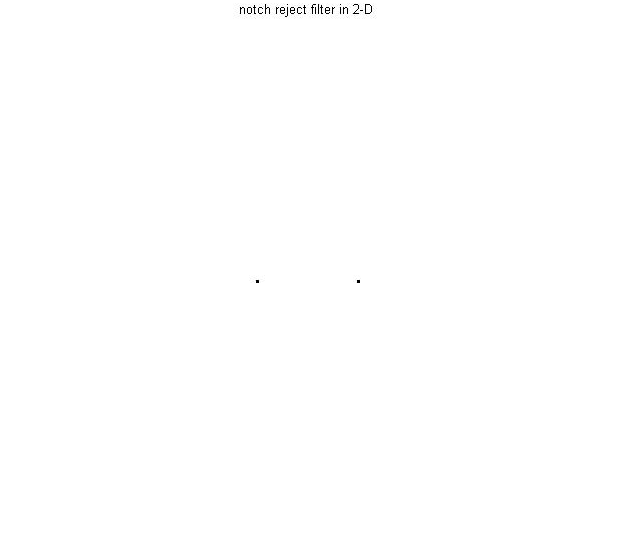
Lets Analyze Different Sinusoids, Optimum Weight and noise expression

Z = 50\*sin(2\*pi\*50\*X/512 )

w =0.2181

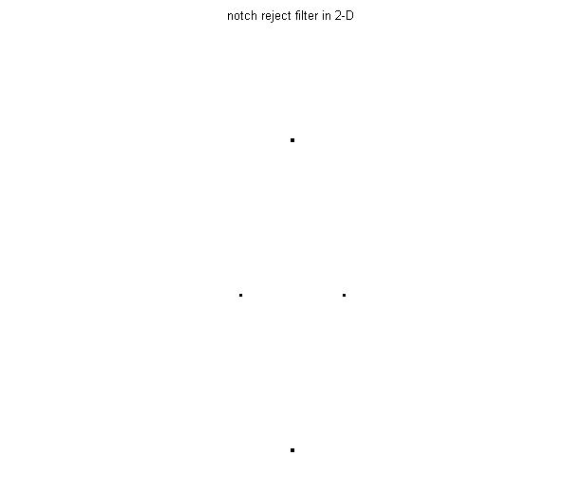
Z = 100\*sin(2\*pi\*50\*X/512 )+100\*sin (2\*pi\*150\*Y/512);

w = 0.6894



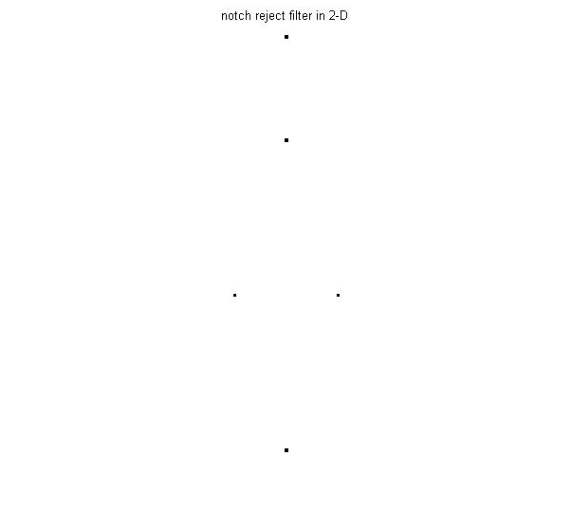
Z= 100\*sin(2\*pi\*50\*X/512 )+100\*sin (2\*pi\*150\*Y/512)+100\*sin(2\*pi\*250\*Y/512);

w =0.7691



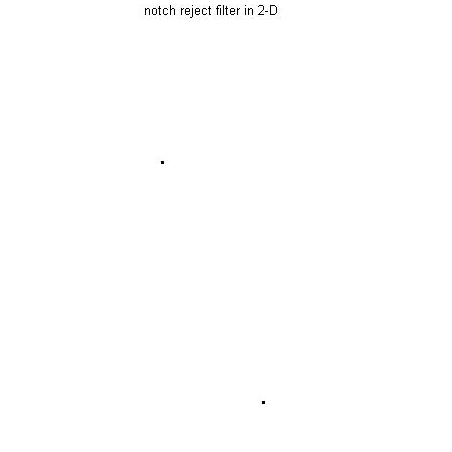
Z = 100\*sin(2\*pi\*50\*X/512+2\*pi\*120\*Y/512 );

w = 0.5266



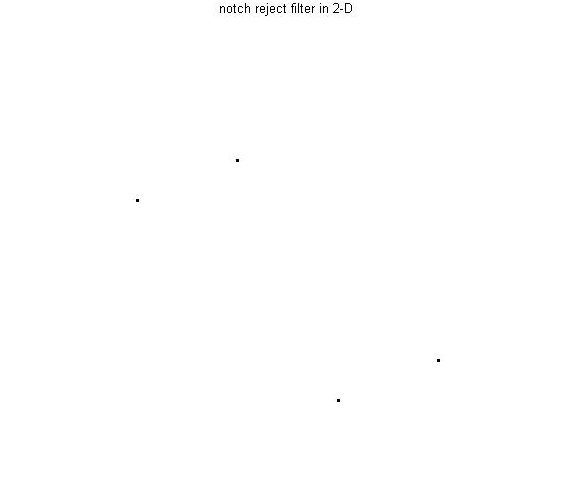
Z = 100\*sin(2\*pi\*50\*X/512+2\*pi\*120\*Y/512 )+ 100\*sin(2\*pi\*150\*X/512 + 2\*pi\*80\*Y/512);

w = 0.6899



Z = 100\*sin(2\*pi\*50\*X/512+2\*pi\*120\*Y/512 )+ 100\*sin(2\*pi\*150\*X/512 + 2\*pi\*80\*Y/512)+100\*sin(2\*pi\*250\*X/512 + 2\*pi\*40\*Y/512

w =0.7694



Thank you