*Editor>>*

clc;

close all;

clear all;

inp=input('1.Low Pass Filter 2.High Pass Filter \n ');

D=input('Enter the Distance of the Filter: '); %Frequency Range

im1=imread('cameraman.tif');

[r c]=size(im1); %Rows and Columns of the image

im\_d=im2double(im1); %Integer into double datatype

msk=zeros(r,c); %New Image of same rows and columns that of the original image

Cx=round(r/2); % X-component of centre of the new image

Cy=round(c/2); % Y-component of centre of the new image

for i=1:r

for j=1:c

Ed=sqrt(((Cx-i).^2) + ((Cy-j).^2)); %Euclidean Distance

Ed=round(Ed); %Rounding of the Euclidean Distance

if inp==1 %For Low Pass Filter

if Ed<=D

msk(i,j)=255; %White

else

msk(i,j)=0; %Black

end

else if inp ==2 %For High Pass Filter

if Ed<=D

msk(i,j)=0; %Black

else

msk(i,j)=255; %White

end

end

end

end

end

im\_fft= fft2(im\_d); %FFT of the Original Image

im\_shift= fftshift(im\_fft); %Shifting of Coordinates(Low Frequency At centre)

im\_filt= (im\_shift).\*(msk); %Filtering

im\_rs=ifftshift(im\_filt); %Reverse Shift(High Frequency at Center)

result=ifft2(im\_rs); %IDFT

im\_2=uint8(result); %Double into Integer datatype

subplot(1,2,1);

imshow(im\_fft);

title('Fourier Transform of Orignal Image');

subplot(1,2,2);

imshow(im\_filt);

title('Frequency Domain');

figure;

subplot(1,2,1);

imshow(im1);

title('Original Image');

if inp==1

subplot(1,2,2);

imshow(im\_2);

title('Low Pass Filtered Image');

else

subplot(1,2,2);

imshow(im\_2);

title('High Pass Filtered Image');

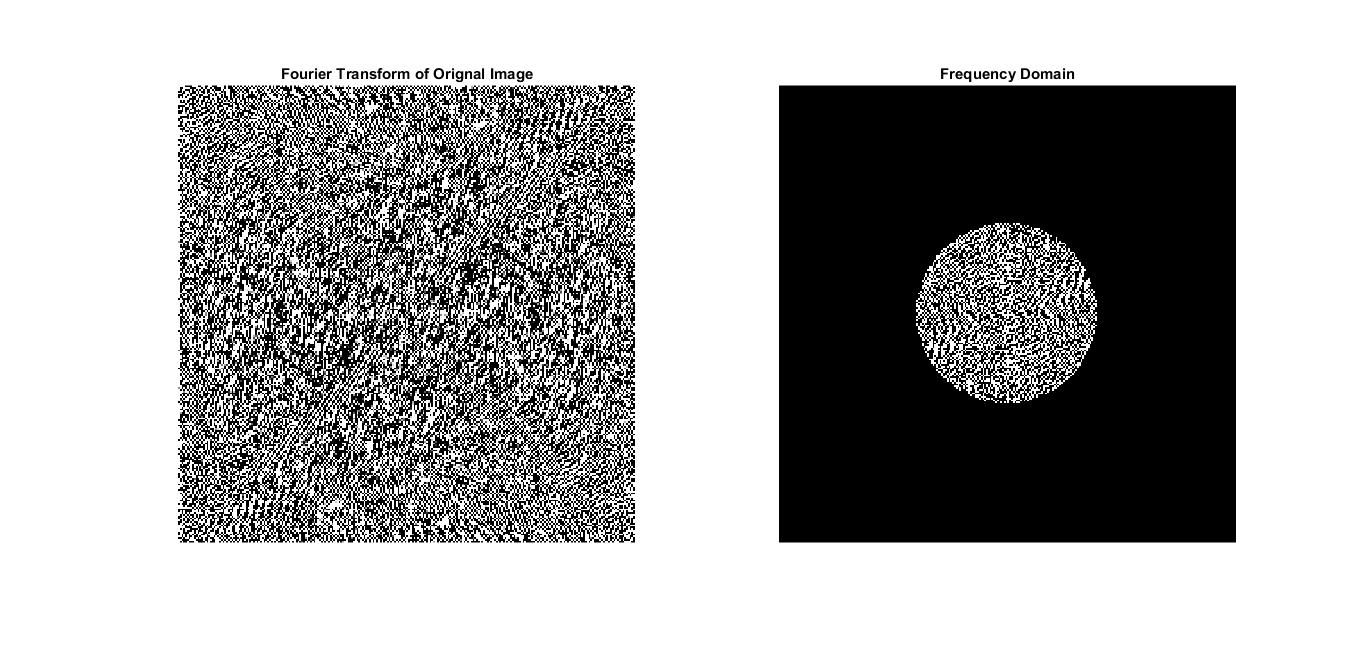
end

*Output>>*

*1.Low Pass Filter 2.High Pass Filter*

*1*

*Enter the Distance of the Filter: 50*





*1.Low Pass Filter 2.High Pass Filter*

*1*

*Enter the Distance of the Filter: 50*

