**Assignment-3**

**Kunal Saini (2014053)**

**Q1 (b)**

****

**Frequency Domain**

****

**Time Domain**

**Type:-'laplacian'**

**M:-256**

**N:-256**

**Do:-20**

**Function Call:-lpfilter('laplacian',256,256,20)**

**OBSERVATION: We can see that in the frequency domain the center portion have pixel values 0 and as we move away they keep on increasing and goes closer to or equal to 255.**

**Q2 (b)**

**Horizontal + Vertical**

**CODE:**

clear all

close all

clc

f=imread('barbara.tif');

figure,imshow(f,[])

f=double(f);

% f = (f - min(min(f)))./(max(max(f))-min(min(f)));

M=size(f,1); N=size(f,2); % nr of rows/columns of image f

C=3; D=3; % nr of rows/columns of kernel h

P=M+C-1; Q=N+D-1; % nr of rows/columns after padding

fp=zeros(P,Q); % zero padding: start with zeroes

fp(1:M,1:N)=f; % insert f into image fp

hp=zeros(P,Q); % Construct filter matrix hp, same size as fp.

hp(1,1)=-4; hp(2,1)=1; hp(1,2)=1;hp(2,2)=0; % Center is at (1,1)

hp(P,Q)=0;hp(P,1)=1; hp(P,2)=0;hp(2,Q)=0;hp(1,Q)=1; % Indices modulo P or Q

Fp=fft2(double(fp), P, Q); % FFT of image fp

Hp=fft2(double(hp), P, Q); % FFT of kernel hp

H = fftshift(Hp);

F1 = abs(H); % Get the magnitude

F1 = log(F1+1); % Use log, for perceptual scaling, and +1 since log(0) is undefined

F1 = mat2gray(F1,[0 1]);

figure,imshow(F1) % Filter

Gp=Fp .\* Hp; % Product of FFTs

gp=ifft2(Gp); % Inverse FFT

gp=real(gp); % Take real part

g=gp(1:M, 1:N);

figure,imshow(g,[])

gnorm = g;

gshar = double(f) - gnorm;

figure,imshow(uint8(gshar))

****

**Filter in Frequency Domain**

****

**Gmask in Time Domain**

****

**Sharp Image in Time Domain**

**Horizontal + Vertical + Diagonal**

**CODE:**

clear all

close all

clc

f=imread('barbara.tif');

figure,imshow(f,[])

f=double(f);

% f = (f - min(min(f)))./(max(max(f))-min(min(f)));

M=size(f,1); N=size(f,2); % nr of rows/columns of image f

C=3; D=3; % nr of rows/columns of kernel h

P=M+C-1; Q=N+D-1; % nr of rows/columns after padding

fp=zeros(P,Q); % zero padding: start with zeroes

fp(1:M,1:N)=f; % insert f into image fp

hp=zeros(P,Q); % Construct filter matrix hp, same size as fp.

hp(1,1)=-8; hp(2,1)=1; hp(1,2)=1;hp(2,2)=1; % Center is at (1,1)

hp(P,Q)=1;hp(P,1)=1; hp(P,2)=1;hp(2,Q)=1;hp(1,Q)=1; % Indices modulo P or Q

Fp=fft2(double(fp), P, Q); % FFT of image fp

Hp=fft2(double(hp), P, Q); % FFT of kernel hp

H = fftshift(Hp);

F1 = abs(H); % Get the magnitude

F1 = log(F1+1); % Use log, for perceptual scaling, and +1 since log(0) is undefined

F1 = mat2gray(F1,[0 1]);

figure,imshow(F1) % Filter

Gp=Fp .\* Hp; % Product of FFTs

gp=ifft2(Gp); % Inverse FFT

gp=real(gp); % Take real part

g=gp(1:M, 1:N);

figure,imshow(g,[])

gnorm = g;

gshar = double(f) - gnorm;

figure,imshow(uint8(gshar))

****

**Filter in Frequency Domain**

****

**Gmask in Time Domain**

****

**Sharp Image in Time Domain**

**OBSERVATION:**

**As we can see when we use H+V+D laplacian filter the output image is much sharper that the image obtained using H+V laplacian filter though due to sharpening in all H,V and D axis the image become so sharp that that the image doesn’t seem much pleasant but the image that is sharpened in only H and V axis seems more effective.**

**As observer from the image of the two filters in the frequency domain, the H+V filter have higher black portion that the H+V+D filter and H+V+D filter id quite circular as we have taken all the 3 axis while the other one is oval because only the H+V axis.**

**Q3 CODE:**

clear all

close all

w=imread('barbara.tif');

w=im2double(w);

b=fspecial('motion',7,0);

k = imfilter(w,b);

figure,imshow(k,[])

[M,N] = size(w);

hp = zeros(1132,804);

P = 1132; Q = 804;

hp(1,1)=1/7;hp(1,2)=1/7;hp(1,3)=1/7;hp(1,4)=1/7;

hp(1,5)=1/7;hp(1,6)=1/7;hp(1,7)=1/7;

g = fft2(w,1132,804).\*fft2(hp,1132,804);

j=real(ifft2(g));

j=j(1:M, 1:N);

figure,imshow(j,[]);

D=abs(k-j);

disp(max(D(:)));

disp(min(D(:)));

****

**Using Command (fspecial)**

****

**Without Using Any Command**

**Max Difference:-0.3597**

**Min Difference:-0**

**OBSERVATION: We can see that the difference is less than 0.5 and both the images appear similar.**

**NOTE:-Solution to rest all questions and parts is in hard copy.**