Project 4: Filtering in the Frequency Domain

Contents

Course No: ECE 5256

Due Date: 2/21/2021

Q1.)Compare the difference between filtering with and without padding.

```
%Low-pass
%Take the Fourier Transform (FT) of an image and multiply its spectrum by a circle of size about ¼ of an axis of the image. The center of the circle should have a v
%Perform the same operation with padding and display an image that is the absolute value of the difference of the two results.
%High-pass
%Repeat the above experiment with a circle of size about ½ the size of the length of an axis of the image. In this case, the center of the circle should be 0, and r
```

Approach

Step 1: Input – Read an image Step 2: Saving the size of the input image in pixels Step 3: Get the Fourier Transform of the input_image Step 4: Assign the Cut-off Frequency D_{0} Step 5: Designing filter: Ideal High Pass Filter and Ideal Low Pass Filter Step 6: Convolution between the Fourier Transformed input image and the filtering mask Step 7: Take Inverse Fourier Transform of the convoluted image Step 8: Display the resultant image as output

Read an Image

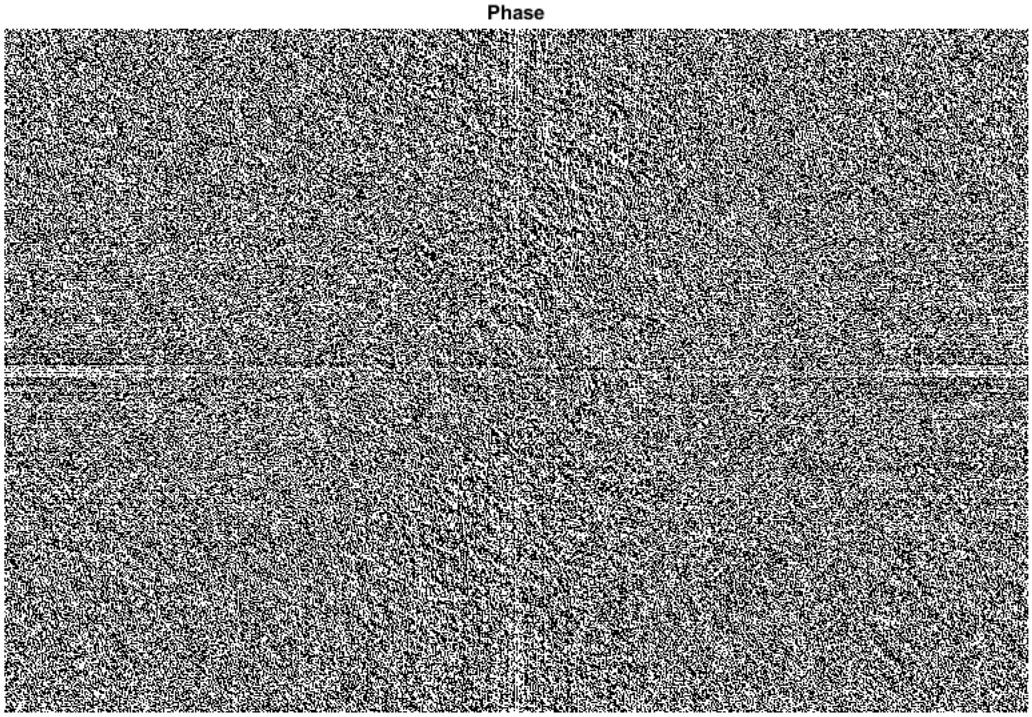
```
%FFT of image
I = rgb2gray(imread("Sea.jpg"));
imshow(I);
```

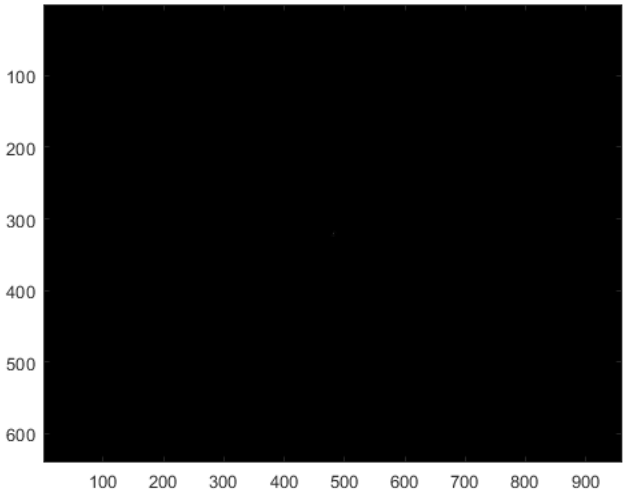
Without Padding

Take the fourier transform of image

```
fftOriginal = fft2(double(I));
% Here we use the fftshift to shift the pixel of the image. This gives us
% the spectrum.
Spectrum = fftshift(fftOriginal);
imshow(Spectrum);title("Phase");
figure;
mag = abs(Spectrum);title("Magnitude");
imagesc(mag);colormap(gray);
```

```
Warning: Displaying real part of complex input.
```



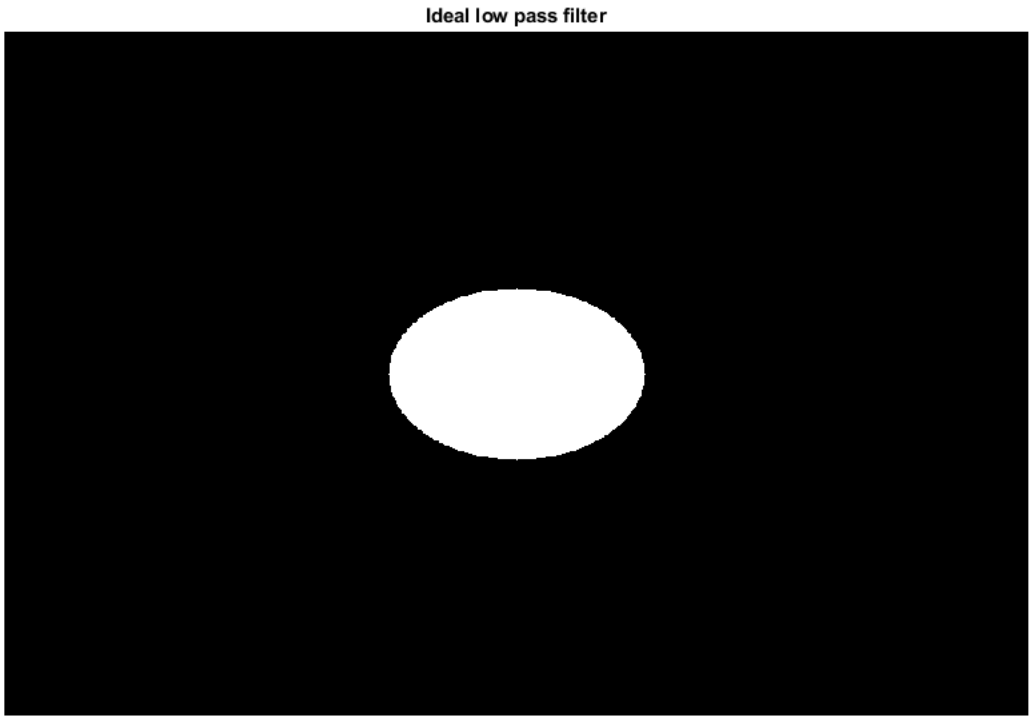


Low pass filter

Here, we create a low pass filter

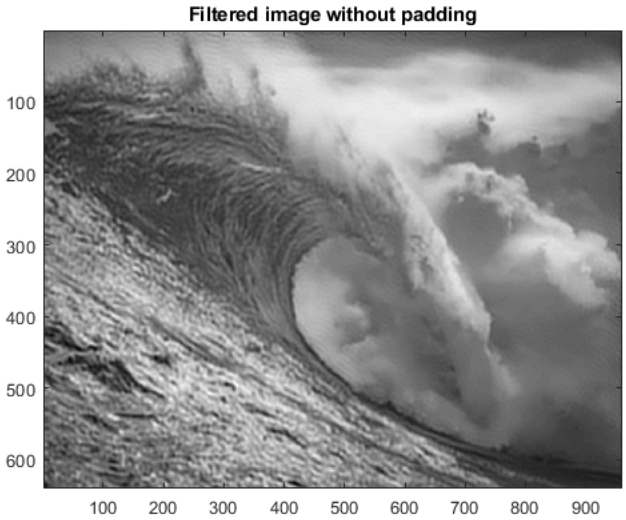
```
[a,b] = freqspace([640,960],'meshgrid');
d = zeros([640,960]);
for i = 1:640
    for j = 1:960
        d(i,j) = sqrt(a(i,j).^2 + b(i,j).^2);
    end
```

```
end
c = 0.25;
H = zeros([640,960]);
for i = 1:640
    for j = 1:960
        if abs(d(i,j)) <= 0.25;
            H(i,j) = 1;
        else
            H(i,j) = 0;
        end
    end
end
imshow(H);title("Ideal low pass filter");
% Convolution between the Fourier Transformed image and the mask
G = H.*Spectrum;
```



Take the inverse FFT of the result

```
output_image = ifft2(ifftshift((G)));
% Displaying Input Image and Output Image
imshow(I),title("Original Image");
figure;
imagesc(output_image),colormap(gray);title("Filtered image without padding");
% We use the inverse FFT to find out how our image looks without padding
```



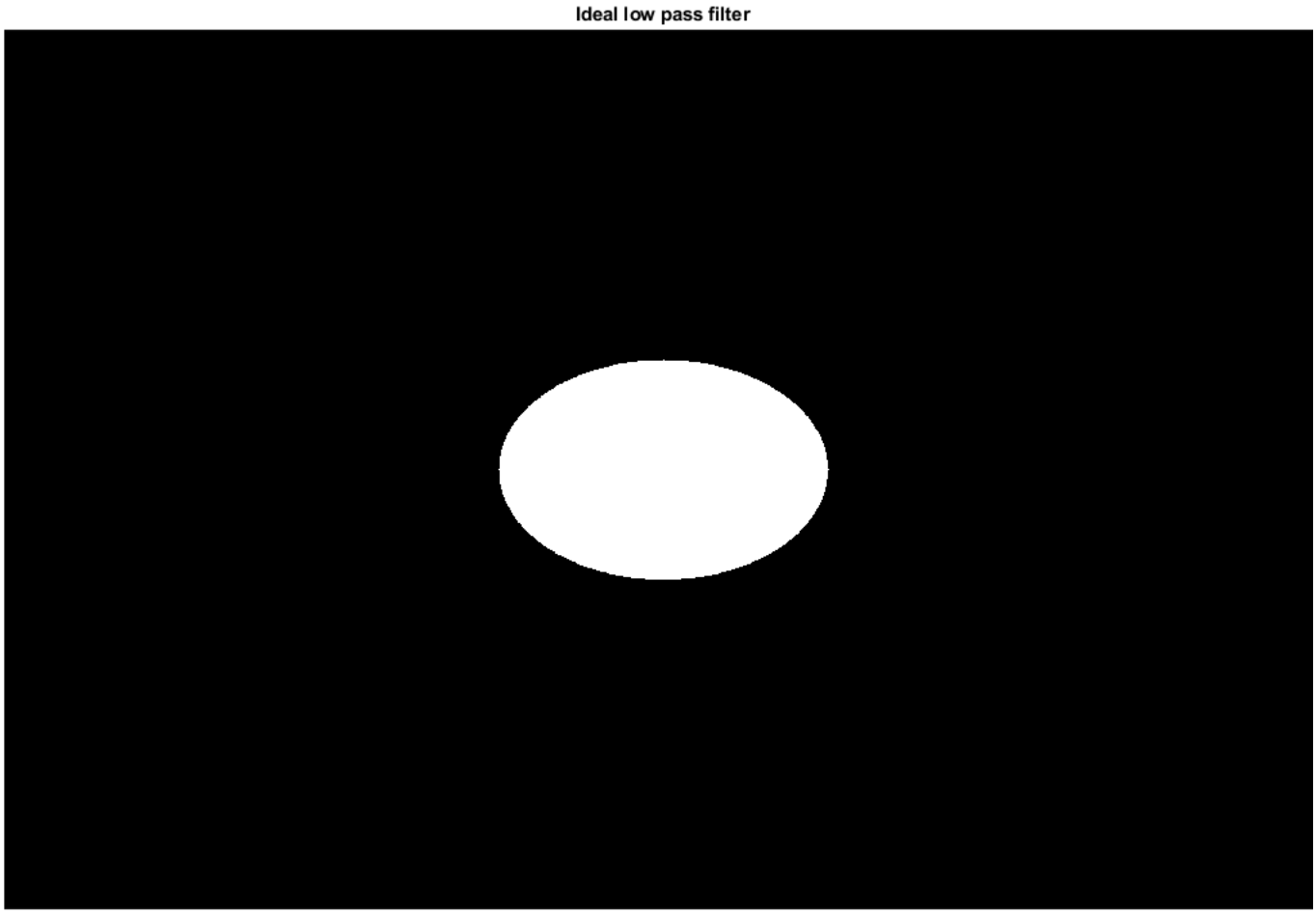
Perform operation with padding.

```
[m,n] = size(I);
% Converting the image class into "double"
b = im2double(I);
% creating a null array of size 2m X 2n
c = zeros(2*m,2*n);
% reading the size of the null array

X_axis=649;
Y_axis=960;

[X_axis,Y_axis] = size(c);
for i = 1:X_axis
    for j = 1:Y_axis
        if i <= m && j<= n
            c(i,j) = b(i,j);
        else
            c(i,j) = 0;
        end
    end
end
imshow(b);title('original image');
figure;
```

```
imshow(c);title('padded image');

Spectrum_c = fftshift(fft2(c));
```

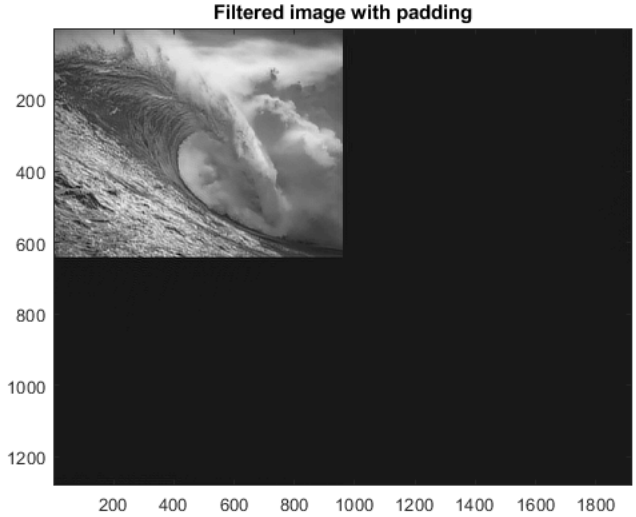
Here, we create a low pass filter

```
[a,b] = freqspace(size(c),'meshgrid');
d = zeros(size(c));
for i = 1:size(c,1)
    for j = 1:size(c,2)
        d(i,j) = sqrt(a(i,j).^2 + b(i,j).^2);
    end
end

H = zeros(size(c));
for i = 1:size(c,1)
    for j = 1:size(c,2)
        if abs(d(i,j)) <= 0.25;
            H(i,j) = 1;
        else
            H(i,j) = 0;
        end
    end
end
imshow(H);title("Ideal low pass filter");
% Convolution between the Fourier Transformed image and the mask
G = H.*Spectrum_c;
```



Take the inverse FFT of the result

```
output_image = ifft2(ifftshift((G)));
% Displaying Input Image and Output Image
imshow(I),title("Original Image");
figure;
imagesc(output_image),colormap(gray);title("Filtered image with padding");
% We use the inverse FFT to find out how our image looks without padding
```

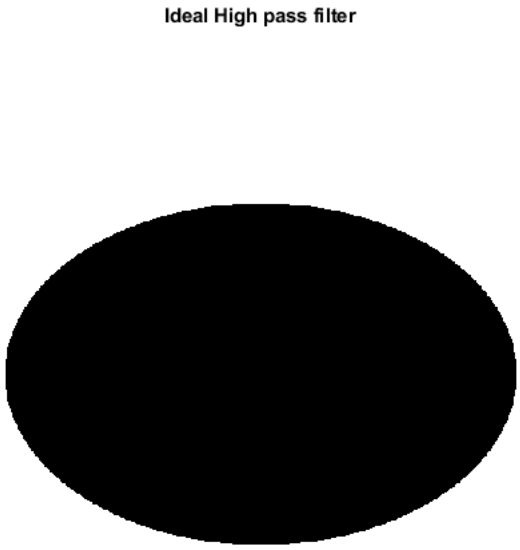


High Pass Filter

Without Padding

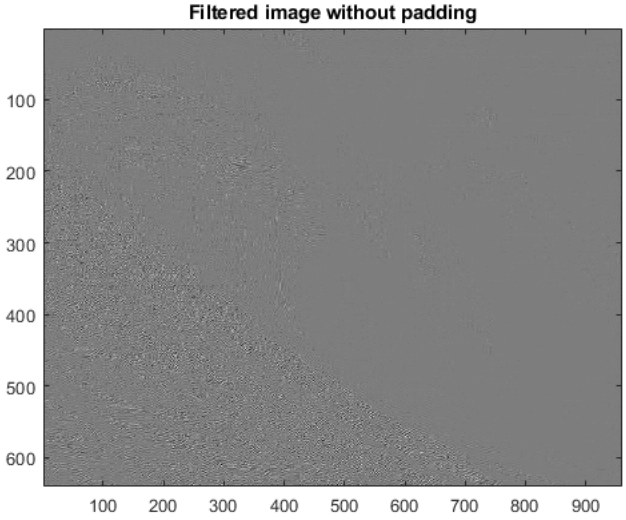
```
[a,b] =freqspace([640,960],'meshgrid');
d = zeros([640,960]);
for i = 1:640
    for j = 1:960
        d(i,j) = sqrt(a(i,j).^2 + b(i,j).^2);
    end
end

H_H = zeros([640,960]);
for i = 1:640
    for j = 1:960
        if abs(d(i,j)) >= 0.5;
            H_H(i,j) = 1;
        else
            H_H(i,j) = 0;
        end
    end
end
imshow(H_H);title("Ideal High pass filter");
```

```
% % Convolution between the Fourier Transformed image and the mask
L = H_H.*Spectrum;
```



Take the inverse FFT of the result

```
output_images = ifft2(ifftshift((L)));
% Displaying Input Image and Output Image
imshow(I),title("Original Image");
figure;
imagesc(output_images),colormap(gray);title("Filtered image without padding");
% We use the inverse FFT to find out how our image looks without padding
```

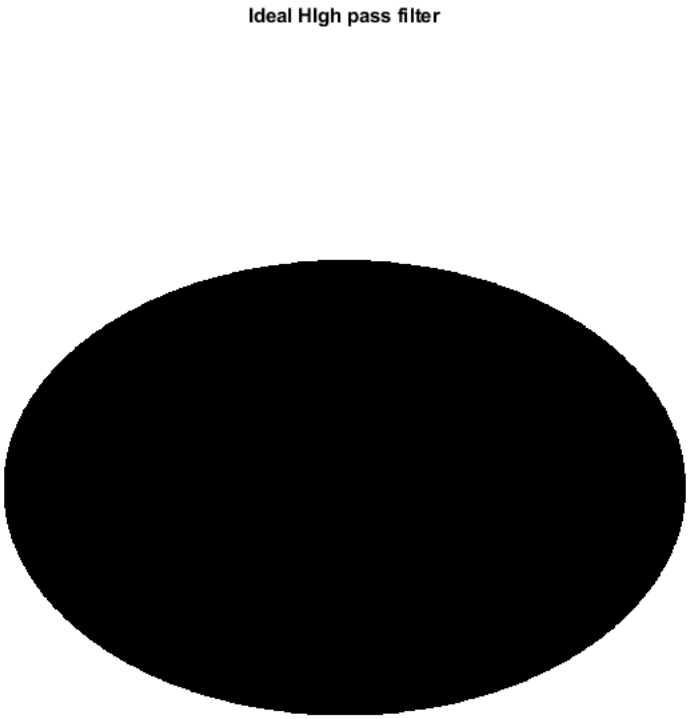


With Padding

Perform operation with padding.

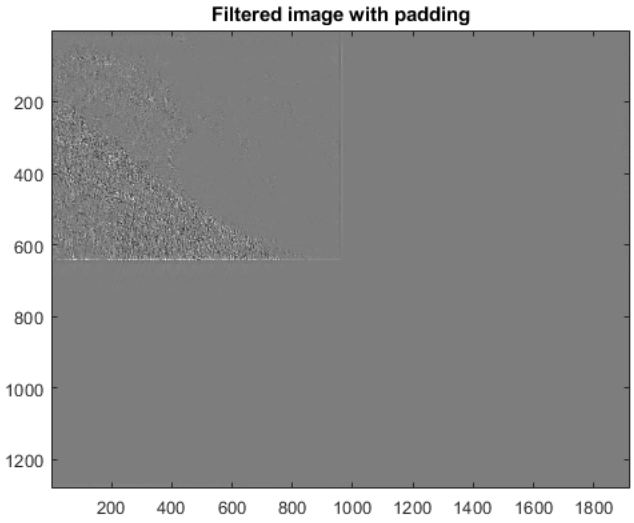
```
% Here, we create a low pass filter

[a,b] = freqspace(size(c),'meshgrid');
d = zeros(size(c));
for i = 1:size(c,1)
    for j = 1:size(c,2)
        d(i,j) = sqrt(a(i,j).^2 + b(i,j).^2);
    end
end
H = zeros(size(c));
for i = 1:size(c,1)
    for j = 1:size(c,2)
        if abs(d(i,j)) >= 0.5;
            H(i,j) = 1;
        else
            H(i,j) = 0;
        end
    end
end
```

```
imshow(H);title("Ideal HIgh pass filter");
G_1 = H.*Spectrum_c;
```



Take the inverse FFT of the result

```
output_imagess = ifft2(ifftshift((G_1)));
% Displaying Input Image and Output Image
imshow(I),title("Original Image");
figure;
imagesc(output_imagess),colormap(gray);title("Filtered image with padding");
% We use the inverse FFT to find out how our image looks without padding
```



Published with MATLAB® R2019a