LAB 12:Image Analysis in Frequency Domain

Objective:

The objective of this lab is to understand Fourier Transform, apply it on images and understand the results.

Lab Description: Fourier Series tells us that any function can be represented as a sum of sines/cosines of different frequencies multiplied by a different coefficient. Similarly, non periodic functions can also be represented as the integral of sines/cosines multiplied by weighing function**.**

The Discrete Fourier Transform of f(x, y), for x = 0, 1, 2…M-1 and y = 0,1,2…N-1, denoted by F(u, v)

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)}$$

for u = 0, 1, 2…M-1 and v = 0, 1, 2…N-1.

It is really important to note that the Fourier transform is completely reversible .The inverse DFT is given by:

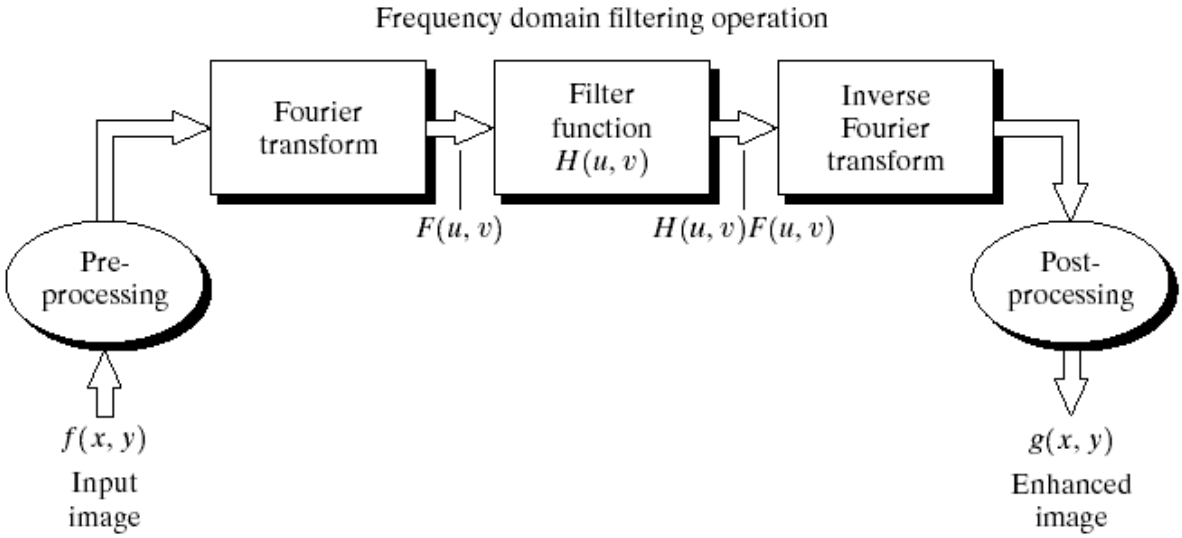
$$f(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(ux/M+vy/N)}$$

for x = 0, 1, 2…M-1 and y = 0, 1, 2…N-1

The filtering in frequency domain consists of following steps:

1. Compute F (u, v) the DFT of the image
2. Multiply F (u, v) by a filter function H (u, v)

Compute the inverse DFT of the result



Some Useful Commands:

1. To obtain the Fourier Transform of an image: **my_transformed_image = numpy.fft.fft2(my_image)**
2. To obtain the Inverse Fourier Transform of an image: **my_inverse_image = numpy.fft.ifft2(my_image)**
3. To shift the DC component of a Fourier Transformed Image to center: **my_shifted_image = numpy.fft.fftshift(my_transformed_image)**
4. To shift the DC component back to the top left corner: **my_inverse_shifted_image = np.fft.ifftshift(my_shifted_image)**
5. To calculate absolute of a value**: my_absolute = numpy.abs(my_image)**
6. To calculate the exponential of an element: **my_exponential = numpy.exp(my_input)**
7. To use the value of pi: **numpy.pi**
8. To denote a complex number: simply put j after it e.g. -1j
9. To multiply two matrices point by point: **my_result = numpy.multiply(my_image, my_filter)**
10. To take log transformation of the image: **my_result = numpy.log(my_image)**
11. To use math function: **import math, x = math.sqrt(25)**

Steps:

1. Take Fourier Transform of the image
2. Shift its Dc component to center
3. Obtain absolute values of the output
4. Brighten up the image using log transformation
5. Normalize the image to obtain High Contrast (min, max) ->(0, 255)

Lab Tasks:

Lab Task 1:

Use the following figures to compute the FFT and display the results.

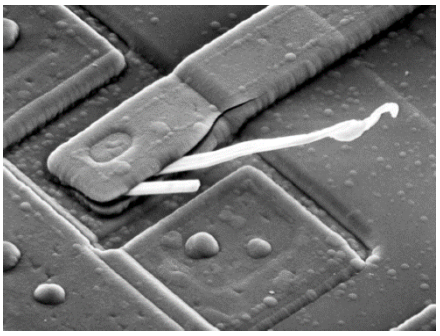


Fig 1

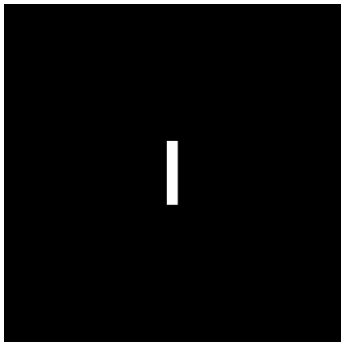


Fig 2

Lab Task 2:

Apply smoothing effect on the given image (Fig 3) using Fourier transform technique. You will apply rectangular shape low pass filter with cut off frequency 30 on given image. Display input image, its magnitude spectrum and output image.

Fig 3

Lab Task 3:

Apply high pass filter on given image below .Create high pass filter of rectangular shape with cut off frequency 30.

Display input image its Fourier spectrum and after applying filter display image in spatial domain.

Fig 04