



Assignment 2

DUE: OCTOBER 30TH

Visual Computing

Deliverables

One person from each group should upload a **PDF report** on **Gradescope** and a zipfile of your code on Canvas. Add any explanations and the result for each section of the assignment in your PDF report to be graded.

Part 1: Discrete Fourier Transform

- a) Step1: Create a 512×512 pixel image.
Step2: Assume that X and Y are the coordinates of pixels in the above image. Calculate the value of each pixel using the following equation:

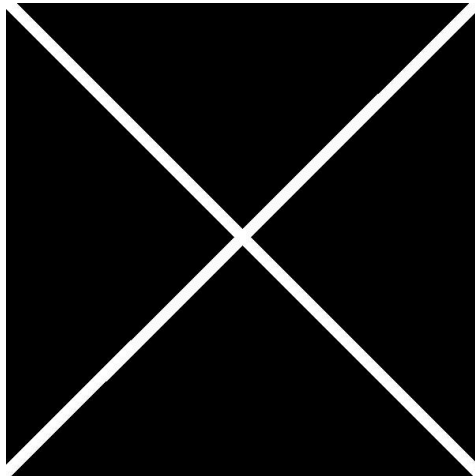
$$I(x, y) = \sin(0.2x) + \sin(0.3x) + \cos(0.4x) + \sin\left(\sqrt{x^2 + y^2} \times 0.15\right) \\ + \sin\left(\sqrt{x^2 + y^2} \times 0.35\right)$$

Show your image in the pdf file.

Step3: Compute the Discrete Fourier Transform of this image. And show the magnitude and phase of DFT for this image. You can use Matlab functions to calculate the DFT. You should rearrange the final result to show the zero frequency in the center of the image. For this purpose you can use “fftshift” function in Matlab.

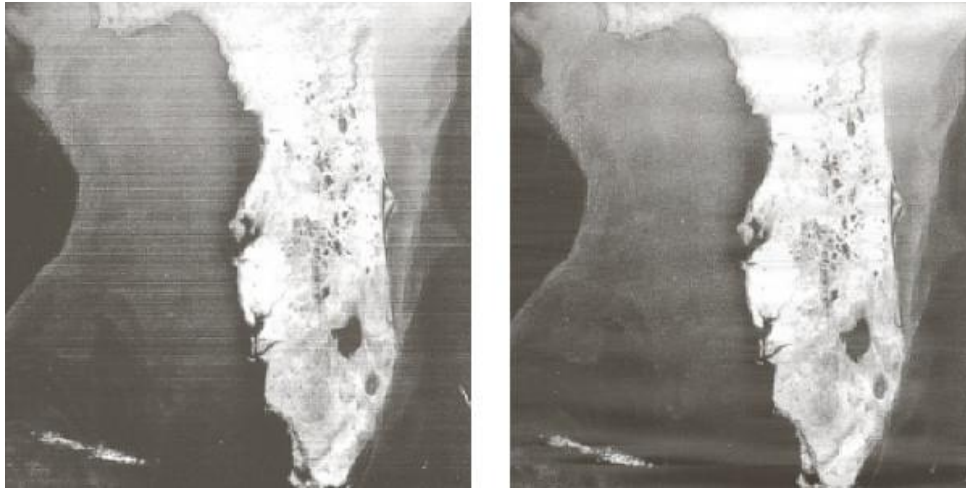
Step4: Multiply the magnitude of DFT with 2 and calculate the inverse Discrete Fourier Transform of it and show the result and explain the difference between the first image that you created in step 2 with this image.

- b) Show the Discrete Fourier Transform of the following image and explain the pattern in the result in your pdf file.



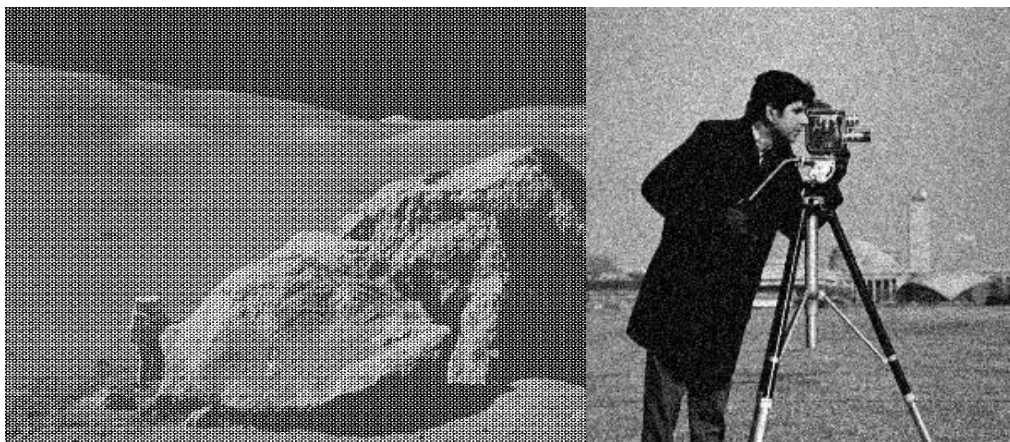
Part 2: Notch Filter

A band-stop filter is a filter that passes most frequencies, but attenuates those in a specific range to very low levels. A notch filter is a band-stop filter with a narrow stopband. These filters are used to remove periodic noise that can be approximated as two-dimensional sinusoidal functions from the image.



For example in the above figure at left you can see the satellite image of Florida showing horizontal scan lines, and at right you can see the result of notch reject filtering.

Use the following images from the image gallery and try to develop your own band-stop filter to remove the noise. For this purpose you should calculate the DFT of the image and remove a specific range of frequencies and then calculate the inverse DFT. You need to test different frequency intervals to find the best filter. (the final results might not be perfect but you should clearly see the noise reduce). Show the final result in the PDF file.



Part 3: Analyzing DFT

Choose a 64x64 image and find the Discrete Fourier Transform for the image. Now Add 64 columns and rows of zeros to the right and bottom side of the original image. Now find the DFT of this new image again. Repeat this process 2 more times each time doubling the image size and padding the pixels on the right and bottom by zeroes. You will therefore have 4 images – first one 64 x 64 with no zero padding and then 128x128, 256x256 and 512 x 512 after padding. Find the DFTs of all these images. Show all your results in the pdf file and explain the relationship between the four DFTs you get. Justify the relationship you discover.