

CS474/674 Image Processing and Interpretation

Fall 2018 – Dr. George Bebis

Programming Assignment 3

Due Date: 10/29/2018

This assignment will help you to better understand the Discrete Fourier Transform (DFT) and its properties. An efficient 1-D Fast Fourier Transform (FFT) implementation in C can be downloaded from the course's webpage. The code is from the book *"Numerical Recipes in C"*. The function prototype is shown below:

```
void fft(float data[], unsigned long nn, int isign)
```

The real part is stored in the odd locations of the array (data[1], data[3], data[5]. etc) and the imaginary part in the even locations (data[2], data[4], data[6], etc.)

The elements in the array data must be stored from data[1] to data[2*nn]; data[0] is not used. If you want, you can use data[0] (i.e., so that you are not wasting memory), call fft as follows: **fft(data-1,N,isign);**

nn is the length of the input which should be power of 2. Warning: the program does not check this condition).

isign: -1 Forward FFT, isign: 1 Inverse FFT (based on our definition)

Warning: the FFT routine provided does not multiply by the normalization factor 1/N that appears in the forward DFT equation; you should do this yourself (see page 506 in the fft handout).

Experiment 1

(1.a) Compute the DFT of the signal $f = [2, 3, 4, 4]$ using the fft routine above and plot the real, imaginary, and magnitude parts of the result. Note that we did this example in class; so you should be able to verify the results. Verify that the inverse DFT works correctly; if you do not obtain the original signal values back, then something is wrong!

(1.b) Generate and display a one-dimensional sine wave with 128 samples that makes 8 cycles over this length:

$$f(x) = \cos(2 \pi u x / N) \text{ where } u = 8 \text{ and } N = 128$$

Provide a graph of $f(x)$ to verify that you have obtained the samples correctly. Compute the DFT $f(x)$ and plot the real, imaginary, magnitude, and phase parts of the result. Make sure that you shift the magnitude to the center of the frequency domain using the property $f(x)(-1)^x \leftrightarrow F(u - N/2)$ (i.e., see page 287). Report and justify your findings in your report.

(1.c) Do the same again but this time for the Rectangular Pulse (file: "Rect_128.dat").

For the rest of the experiments, you would need to implement the 2-D DFT as well as its inverse DFT. Your implementation should be based on the separability property that we discussed in class (i.e., compute the 2-D DFT by computing the 1-D DFT on the rows, followed by the 1-D DFT on the columns of the result). Use the following function prototype:

fft2D(N, M, real_Fuv, imag_Fuv, isign)

Before doing the experiments below, make sure that your 2-D DFT works correctly. For example, take the 2-D DFT of an image, followed by the inverse 2-D DFT; if you don't get the original image back, then something is wrong!

Experiment 2

(2.a) Generate a 512 x 512 image which contains a 32 x 32 square placed at the center of the image. Set the background to black (i.e., 0) and the interior of the square (and its boundary) to white (i.e. 255). The image should look like the one shown in Fig. 4.23 on page 296. Take the DFT of the image and show its magnitude without shifting it to the center of the frequency domain. Then, shift the magnitude to the center of the frequency domain (i.e., using property 4 from Table 4.3 on page 304) and show the centered magnitude. Your results should look similar to those shown in Fig. 4.23(d) on page 296. Note: to properly show the magnitude of the DFT, you should use the $\log(1+F(u,v))$ transformation as discussed in class. Report and justify your findings in your report.

(2.b) Generate a 512 x 512 image which contains a 64 x 64 square placed at the center of the image. Repeat the steps given in (2.a). How do your results compare with (2.a)?

(2.c) Generate a 512 x 512 image which contains a 128 x 128 square placed at the center of the image. Repeat the steps given in (2.a). How do your results compare with (2.a) and (2.b)?

Graduate Students Only: Experiment 3

In this experiment, you are going to examine the importance of magnitude and phase (see my slides as well as pages 298-299). First, compute the DFT of the *Lenna* image. Then, do the following:

(3.a) Set the phase equal to zero, and take the inverse DFT (**hint:** set the real part to the magnitude of the image and the imaginary part to zero). The resulting image should look nothing like the original. Explain your results.

(3.b) Let the phase be the original one and set the magnitude equal to one and take the inverse DFT (**hint:** to set the magnitude equal to one, set the real part to $\cos(\theta)$ and the imaginary part to $\sin(\theta)$ where $\theta = \tan^{-1}(\text{imag}/\text{real})$ – **prove it!**). Since the magnitude is set to such a small value in the frequency domain, all the values in the spatial domain will be very small when you take the inverse DFT. To alleviate this problem, rescale the pixel values after the inverse DFT has been taken (i.e., values should be in $[0, 255]$). Explain your results.

Note: to compute \tan^{-1} , use the function **atan2()**.

Laboratory Write-up

For each programming assignment, you are to turn in a report (please, follow closely the instructions posted on the course's website). **The report is very important in determining your grade for the programming assignment.** Be well organized, type your reports, and include figure captions with a brief description for all the figures included in your report. Motivation and initiative are greatly encouraged and will earn extra points.