PROGRAMMING ASSIGNMENT 3
DISCRETE FOURIER TRANSFORMS

DUE: 2013 NOVEMBER 18, MONDAY

We are going to use the Discrete Fourier Transform (DFT) on data and images to get a better understand of how the Fourier Transform encodes information. Note that the DFT is just the coefficients of a trigonometric polynomial interpolation of data. Namely, it tells you the amplitude on different sines and cosines of different frequencies. This means it is very good at picking up repeating patterns.

Do the exercises below, and answer and discuss the questions in your write up. Try to give concrete examples or analytical arguments whenever possible. For this assignment, you do not need to attach your code (unless you do the bonus), but please do attach plots and images. If LaTeX gives you trouble importing images, just print them out, and refer to them in your write-up as "Fig. 1", "Fig. 2", and so on. Make sure to label them correctly!

Part I ***To be started in lab, only spend 5-10 minutes though.*** **(50 points+Bonus)** Recall that, as the earth rotates, it feels different gravitational pulls due to the presence of the sun and moon, and this creates tides in the ocean. The height of the water on a fishing dock was recorded every hour for several days. The data can be found on the course website as the file "`tide_data.txt`". Download this file, and copy/paste the data into Matlab (it is already in Matlab vector format). Call the vector "height".

Exercises

(1) Plot the data using "`plot(height)`" (Matlab assumes the x-axis is just integers unless otherwise stated.)

(2) To take the DFT of Matlab using the FFT algorithm, just use "`fft(height)`". Plot this using `plot(fft(height),'.')`. (Remember to type "`help plot`" if you need information about the plot command.) Recall that the FFT gives you complex numbers. What symmetries do you see, if any?

(3) Since it hard to see what is really going on (since the FFT gives complex numbers), take the absolute value (`abs`) of the FFT and plot it. You should see some spikes. What are the spikes telling you? Suppose you were from another planet, and knew very little about earth, but got this data back about the tides. What could you conclude? Also, why are some of the spikes seemingly reflected about the middle point? What does the very first (non-reflected) spike represent in terms of the data?

(4) Try compressing the data: Set all the other data to zero except for some data near the spikes. Reconstruct the signal approximately by using the inverse Fourier Transform "`ifft`". If we don't could the data that has been "zeroed out" as information, how much compression can we gain while still maintaining a "good" signal?

(5) **Bonus** (50 points): Code your own FFT and compare it with the results for the tide data. Also, test your speed (e.g., using `tic` and `toc` in Matlab, or recorded times in C/C++/Fortran/Python, etc.), and compare your speed with Matlab's. If you do this, then you *should* submit your code in both paper and email format.

Part II ***To be started in lab. Spend most of your time on Part II.*** **(50 points)** We are going to use the 2D FFT on images. Grey-scale images can be thought of as functions on a 2D rectangle which take values between 0 and 1 (or, in digital images, between 0 and

255), with 0 meaning "black" and 1 meaning "white". Color images are just 3 channels of monochrome images (red, blue and green), so they can be handled similarly.
Matlab can work with images, but it is hard to directly edit them in Matlab, so we are going to use "Gimp", which is similar to Photoshop, but free.

Getting set-up with the Gimp FFT plugin

Step 1. Follow the instructions on the course webpage to install the FFT plug-in for Gimp.
Step 2. Download all the images under "Pictures for transforming" by right-clicking on them and using "Save as".
Step 3. Open Gimp, and the open the "`v_stripes.jpg`" image from within Gimp.
Step 4. Go to the Menu and do:

`Filters` → `Generic` → `FFT Forward`

Step 5. You are now looking at the FFT of the image, with the x-y-axes having their origin at the center of the screen. Can you find the tiny single black point? Which axis (x or y) is it on? Why?

Exercises

(1) Repeat the above steps with "`h_stripes.jpg`". Then try "`d_stripes.jpg`" and "`box.jpg`". Record any observations you may have in your report. It may be fun to make your own variations on these and play with them (not necessary for the report).
(2) Next, open "`moire_family.jpg`". Note the speckled newsprint pattern. Such patterns are sometimes called Moiré patterns, and they have a very periodic nature. Take the FFT of the image. Can you find the spots where the pattern lies? Zero-out these spots by coloring them black with the paintbrush tool. Then do:

`Filters` → `Generic` → `FFT Inverse`

What do you see? Can you clean up the photo and get rid of the Moiré pattern while still maintaining decent image quality?
(3) Open "`lion.jpg`". Can you "free the lion" from the cage by "removing" the bars? Where is the information about the bars stored in Fourier space?
(4) Zero-out (i.e., paint black) as much of the FFT of `lion.jpg` as you can while still maintaining a good image. What compression rate can you achieve? What does this mean about where the major image details are stored? Why does this make sense?

The last images, `parrots.jpg` and `stripe_family.jpg` are not required for you to include in your report. They are just there for you to play around with if you want.