CS 370 Winter 2013: Assignment 3

Instructor: Yuying Li	Office: DC3623	e-mail: *yuying@uwaterloo.ca*
Lec 001 :	MWF 8:30-9:20 MC2054	OH (Li):Tues 2-3pm
TAs:		
Nian Ke	knian@uwaterloo.ca	DC3594
Mark Prosser	rmprosse@uwaterloo.ca	DC2302B
Haofan Zhang	h267zhan@uwaterloo.ca	DC3594

Web Site: www.student.cs.uwaterloo.ca/~cs370/

Due Mar 15, 2013, 5:00 PM, in the Assignment Boxes, 3rd Floor MC. Please attach a cover page, which you can download at the course website, to your submitted assignment.

1. **(10 marks)** Consider a second order Runge-Kutta method given by

$$y_{n+1} = y_n + \frac{h}{3}\left(f(t_n, y_n) + 2f(t_n + \frac{3}{4}h, y_n + \frac{3}{4}hf(t_n, y_n))\right)$$

   Carry out a stability analysis of this method using the test equation

$$y'(t) = -\lambda y(t), \qquad \lambda > 0$$

   and determine a condition for stability.

2. **(10 marks)**

   (a) Compute by hand the DFT of

$$\begin{bmatrix} -1/4 \\ 1/4 \\ 3/4 \\ 1/4 \end{bmatrix}$$

   (b) Compute by hand the FFT of

$$\begin{bmatrix} -1/2 \\ 0 \\ 1 \\ 0 \\ -1/2 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

   using the butterfly diagram.

   Show your work.

3. **(5 marks)** Suppose the vector $\{f_n\}_{n=0..N-1}$ has DFT given by $\{F_k\}_{k=0..N-1}$. Consider the array of $2N$ numbers

$$\tilde{f} = [f_0, f_1, \ldots, f_{N-1}, f_0, f_1, \ldots, f_{N-1}].$$

   Give the DFT of the sequence $\{\tilde{f}_n\}_{n=0,\ldots,2N-1}$ in terms of $\{F_k\}_{k=0..N-1}$ using the Fast Fourier algorithm.

4. **(10 marks)** Let f_n, $n = 0, 1, \cdots, N-1$, be real periodic input data with an even N and $f_{N+n} = f_n$. Let F_k, $k = 0, 1, \cdots, N-1$, be the DFT of $\{f_n\}$. Show that if $\{f_n\}$ values satisfy the symmetry $f_n = f_{N-n}$ then the Fourier coefficients $\{F_k\}$ are all real numbers.

5. **(Image Compression, 15 marks)** In this problem, we study the compression of gray-scale images. In Appendix F: Image Processing in Matlab of the course notes, you have the information needed to convert such images into a two dimensional array. Compression is obtained by dropping (relatively) small Fourier coefficients on 8×8 pixel subblocks. By this we mean that if $\{f_{i,j}\}$ are the original pixel values in a given subblock and $\{F_{k,\ell}\}$ are the corresponding DFT, then we drop any $F_{k,\ell}$ such that

$$|F_{k,\ell}| \leq F_{max} \cdot tol.$$

Here F_{max} is the maximum of $\{|F_{k,\ell}|\}$ in each block and tol is our drop tolerance.

The file `dogBW.jpg` on the course web page contains an image which we will use in this compression question.

a) **Compression**

Create a MATLAB function, `Compress.m`, that has the following prototype:

```
[Y, drop] = Compress(X, tol)
```

It takes as inputs the original image, `X`, and the drop tolerance parameter, `tol`, and outputs a compressed image `Y`. It also returns the drop ratio, `drop`, which is defined to be:

$$\text{drop ratio} = \frac{\text{Total number of nonzero Fourier coefficients dropped}}{\text{Number of nonzeros in original Fourier Coefficients}}.$$

If drop ratio = 0, then no Fourier coefficient is dropped; if drop ratio = 1, then all Fourier coefficients are dropped. In general, it should be between 0 and 1.

Specifically your MATLAB function should:

- compute the 2D Fourier coefficients (`fft2`) for every 8×8 subblock.
- for each subblock, set those Fourier coefficients having modulus less than $F_{max}\cdot$ `tol` to 0.
- record the number of coefficients dropped
- reconstruct the compressed 8×8 image array by using the inverse 2D Fourier transform (`ifft2`). **Note**: the reconstructed image array must be set to the real part of the inverse transform.
- after all the 8×8 subblocks for all the components have been processed, return the entire compressed image as `Y` and the drop ratio as `drop`.

b) **Compression Levels**

Determine (by trial and error on different `tol`, not by writing any code) four values of `tol` resulting in drop ratios of 0, 0.4, 0.5 and 0.7. Write a MATLAB script to do the following:

- Execute `Compress.m` with these set of `tol` values.
- Display the four compressed images using `subplot` for each compressed image `Y`. Each plot should have a title, the `tol` value used, and the resulting drop ratio.
- Plot the normalized mean square error between the original image and the compressed image vs the drop ratio for the compressed image (refer to Appendix F in the course notes for normalized mean square error).

What to hand in: Please submit

(a) A listing of `Compress.m`.

(b) A listing of the MATLAB script.

(c) A figure with 4 plots of the DFT compressed images.

(d) The error plot for the DFT compressed images.

(e) A brief commentary on the compressed images.