CSE 166: Image Processing, Spring 2019 – Assignment 4
Instructor: Ben Ochoa
Due: Wednesday, May 22, 2017, 11:59 PM

Instructions

- Review the academic integrity and collaboration policies on the course website.
- This assignment must be completed individually.
- This assignment contains both math and programming problems.
- Programming aspects of this assignment must be completed using MATLAB.
- Unless specified below, you may not use MATLAB functions contained in toolboxes, including the image processing toolbox. Use the MATLAB `which` command to determine which toolbox a function is contained in. If you are unsure about using a specific function, then ask the instructor for clarification.
- You must prepare a report as a pdf file. The report must contain your solutions and results, and all of your MATLAB source code as a listing in the appendix of your report.
- Additionally, you must create a zip file containing all of your MATLAB source code.
- Your source code must contain a file `main.m` which runs all code necessary to produce results for your report. `main.m` should run start to finish without error. Use relative paths to read input data. For questions which require numerical output, `main.m` should print a message indicating what question is being answered followed by the numerical output for that question. Example: `display('Problem 2b')` followed by your answer to problem 2b. The instructors should be able to reproduce your report by running `main.m`
- You must submit both files (.pdf and .zip) on Gradescope. Further, you must mark each problem on Gradescope in the pdf file.
- It is highly recommended that you begin working on this assignment early.

Problems

1. **Textbook problems (11 points)**
   (a) Problem 6.1 (5 points)
   (b) Problem 6.3 (3 points)
   (c) Problem 6.17 (1 point)
   (d) Problem 6.30 (2 points)

2. **Programming: The wavelet transform and wavelet-based image processing (35 points)**

   Use image analysis-related functions contained in the MATLAB Wavelet Toolbox to complete these problems.

   (a) **The wavelet transform (15 points)**

      i. In this problem you will implement the 1D discrete wavelet transform using Haar wavelets. Develop a MATLAB function called `haarTransform` which takes as an argument a 1-dimensional signal S and outputs the approximation and detail coefficients of the first level Haar transform. You may assume that the length of S is a power of 2. Generate 1024 linearly spaced samples (hint: use the function `linspace`) from the function $S(x) = 20x^2(1-x)^4 \cos 12\pi x$ and compute the 5th order Haar wavelet decomposition. The *energy* of a signal S is defined as the sum of the squares of its values: $e = \sum_i S_i^2$. Compute the energy of the original signal along with the energy in both the approximation and detail coefficients. Plot the fraction of energy contained in the approximation and detail coefficients separately for each level in your decomposition. Briefly discuss your results and how signal energy is transferred by the Haar decomposition.

      ii. Develop a MATLAB script called `hw4_dwt.m` that reads the input image `cameraman.tif` (included with MATLAB), computes a 3-scale Haar discrete wavelet transform (DWT) of the input image and creates a figure similar to the one shown on slide 29 of lecture 13, reconstructs the approximation coefficients for level 2 from the level 3 decomposition and writes the image to `cameraman_A2.png`, and reconstructs the approximation coefficients for level 1 from the level 2 decomposition and writes the image to `cameraman_A1.png`. Use the function `imread` to read the input image in MATLAB. Use `imwrite` to write the output image in MATLAB.
      Include in your report the input image, figure similar to the one shown on slide 29 of lecture 13, and output approximation coefficients images.

   (b) **Wavelet-based edge detection (10 points)**

      Develop a MATLAB script called `hw4_edges.m` that reads the input image `cameraman.tif` (included with MATLAB) and computes a 1-, 2-, 3-, and 4-scale Haar DWT of the input image. For each resulting DWT, set the lowest scale approximation component to zero and compute the inverse Haar DWT.

      Include in your report the input image. Additionally, include figures of the inverse Haar DWT results (with colorbars to show the scale). Briefly discuss the resulting images, including any differences between them.

   (c) **Wavelet-based noise removal (10 points)**

      Develop a MATLAB script called `hw4_nr.m` that reads the input image `rice.png` (included with MATLAB), computes a 2-scale Haar DWT of the input image, sets the highest resolution detail coefficients to zero, computes the inverse Haar DWT, and writes the results to `rice_nr.png`.

Include in your report the input image and output image. Briefly discuss the results.