CSE 166: Image Processing, Fall 2017 – Assignment 4

Instructor: Ben Ochoa

Due: Wednesday, November 22, 2017, 11:59 PM

Instructions

- Review the academic integrity and collaboration policies on the course website.
- 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 containing your solutions and results.
- Your report will be a pdf file named `CSE_166_hw4_lastname_studentid.pdf`, where `lastname` is your last name and `studentid` is your student ID number.
- All of your MATLAB source code must be included as a listing in the appendix of your report.
- Submit your report on Gradescope.
- Additionally, you must create a zip file named `CSE_166_hw4_lastname_studentid.zip`, where `lastname` and `studentid` is your last name and student ID number, respectively. This zip file will contain the pdf file and a directory named `code` that contains all of your MATLAB source code.
- Submit your completed assignment by email to `rkollipa@eng.ucsd.edu` **and** `asrikant@ucsd.edu`. The subject of the email message must be `CSE 166 Assignment 4`. Attach the zip file to the message.
- It is highly recommended that you begin working on this assignment early to ensure that you have sufficient time to correctly implement the algorithms and prepare a report.

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 (30 points)**

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

   (a) **The wavelet transform (10 points)**

   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 19 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 19 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). Comment on 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. Comment on the results.