CS/ECE 545 Digital Image Processing
Homework 3, Spring 2014 (Due March 26, by 6PM)

Note: Some of the problems below include questions for you to answer. Put your answers in a separate Word or PDF document and submit it in a zip file that also your code.

1. **Burger & Burge Exercise 8.2 (page 153) (25 points):** Conduct a series of experiments to determine how image contrast affects the performance of the Harris detector, and then develop an idea of how you might automatically determine the parameter t_H depending on image content. Specifically, explain in your README file, what you learned from your experiments about how image contrast affects the performance of the Harris detector. Then also implement modifications to the original Harris detector code and create a modified Harris detector **`modified_harris_detector.java`** that automatically determines depending on image content.

2. **Laplacian edge detection method (53 points)**: The goal of this assignment is to implement the Laplacian edge detection method. You should implement separable convolution routines. You will just have to call these routines with different kernels (Gaussian and 1st and 2nd derivatives of Gaussian). If you do use a helper class, be sure to turn in the source code for it as well. Your filters should all work with floating point images as input. Also, all convolutions should be separable, use wrapping on the boundary, and all filters that use Gaussians should have a (σ) parameter that you can set.
   a. (8 pts) Create a plugin called **`Gradient_Magnitude`** that computes the gradient magnitude of an image.
   b. (15 pts) Create a plugin called **`Laplacian_`** that computes the Laplacian of an image. Use Gaussian 2nd derivative filters.
   c. (15 pts) Write a plugin filter called **`Zero_Crossings`** that calculates the zero crossings in an image. It should set a pixel to 1 if it is a zero crossing and 0 otherwise.
   d. (15 pts) Put it all together! Use the three filters above to do edge detection. Now you should have a threshold parameter to apply a threshold to the gradient magnitude. Follow this by an AND operation with the result of the Laplacian zero-crossing image. Your resulting images should be binary, with 1 on the edges and 0 elsewhere. Call your plugin filter **`Laplacian_Edge_Detection`**.

Submit all ImageJ plugins (`modified_harris_detector.java, Gradient_Magnitude.java, Laplacian_.java, Zero_Crossings.java and Laplacian_Edge_Detection.java`)

Written Part

In this part all questions involve applying your filters to the image **`hoos cow.tif`**.
1. (4 pts) Apply your gradient magnitude filter with $\sigma = 1$. Describe what you see in the background (behind the cow). Now apply your gradient magnitude filter with $\sigma = 8$, and describe the background. Why are they different?

2. (4 pts) Apply your Laplacian filter with $\sigma = 4$. Describe the transition at the edge on the top of the cow's head (as you move from the background down into the cow's head). Compare this to the transition at edge on the top of the cow's ears. Why are the transitions different?

3. (7 pts) Apply your Laplacian edge detection with a fixed threshold = 10, but with increasing values for σ = 1, 4, 8, 16. Describe the problems you see as σ gets larger. Now for the same values σ = 1, 4, 8, 16 try to change the threshold as you change σ so that you do not have this problem. How did you have to change the threshold?

4. (7 pts) Imagine that you were writting a computer vision algorithm that had to recognize text on any signs that it found in an image. Find good values for σ and the threshold that will outline the edges in the text but will not get many "false" edges in the background. What values did you use? Try this also for the images **`sign1.tif`** and **`sign2.tif`**. Did you have to change the parameters to get good results? What values did you use?

Turn in: A txt or pdf file with written answers to the above questions.

Submitting Your Work

Submit all plugins plugins (`modified_harris_detector.java, Gradient_Magnitude.java, Laplacian_.java, Zero_Crossings.java and Laplacian_Edge_Detection.java)` and your README file in Word or PDF put into ONE zip file named *yourfirstname_yourlastname_hw3.zip* using turnin. Your README should contain all your answers to the questions above.

DON'T EMAIL ME YOUR HOMEWORK. ALSO, TEST YOUR CODE IN THE ZOOLAB BEFORE SUBMITTING!!