

Lab Assignment 3 – CSE 332/564, Spring 2008

Due: Tuesday, March 10, 11:59pm

This lab will introduce you to local image manipulations. Continue to use the code you developed for lab 1. Upon completion of the assignment you will submit the following, via blackboard:

- the complete software (all that is needed to build the executable: source code, project files, etc)
- an executable (.exe file) of your work
- a comprehensive report that illustrates with images and narrative text all aspects of your work
- any images not part of the provided collection that you have used for testing your software

All of these components are equally important. Please note the policies posted on the class webpage. Also please note: While this assignment looks short, it will take time to do. If graphics programming is new to you, it will take some time to understand the mechanisms here. But you will need these concepts throughout the course, so it pays off to understand them now.

Check out the new demo on the lab page to see suggestions how the following will work in practice.

1. Add new items to the GUI

On the menu bar, to the submenu 'Image Processing', add three menu items 'Box', 'Sobel', 'Median Filter', and 'Subsample'.

Use FLUID to add two value sliders: 'Box Mask size;' and 'Sub-sample factor'. The former should go from 3-7, the latter from 2-4. Write the corresponding callbacks that set these values in your program whenever the user moves the sliders. Initialize the values to 3 and 2, respectively.

2. Box filtering

Convolve the image with the box filter mask (the averaging filter). The mask size should be according to the selected box mask size. When the size is even (4,6) just use (3,5). No need to filter the seams of the image. Just copy the image first into the working image and then compute only those pixels that have full mask coverage of their neighbors. For color images, filter (convolve) R,G,B separately. Make sure you can undo this operation at the end. Observe the effects obtained with the various box sizes.

3. Sobel filtering

Convolve the image with the 3x3 Sobel filter mask to find the edges. Again, there is no need to filter the seams of the image. Just copy the image first into the working image and then compute only those pixels that have full mask coverage of their neighbors. Use floating point to compute the edges because you will get values beyond the byte interval [0, 255]. Then combine the dx and dy edge images using $\sqrt{(dx^2)+(dy^2)}$. You will get positive values beyond 255. For display, find the maximum value *maxval* in this combined image and scale all values by $255/maxval$. The result can be stored into a byte image for display.

For color images, experiment with computing the edges for R,G,B separately. Also try to compute the edges of the V channel (after HSV conversion), and then transform the V edge image and the original S, H channels back to RGB for display. Again, make sure you can undo this operation at the end.

4. Median filter

Implement median filtering. Try the noise.ppm image and compare with what you get when you use the box filter for filtering.

5. Sub-sampling

Depending on the set slider position p , compute a reduced image where you only use every p^{th} pixel and row. Observe the aliasing effects you get with images with lots of detail, such as starImg.ppm or text.ppm. You may have to first scale down the displayed image using the mouse interaction. In particular the starImg, first make it smaller such that you do not see the aliasing in the center anymore and then subsample. In the demo the undo function has not been implemented, but you can try to do it.

6. Anti-aliasing

Explore how prior smoothing (in your case, box filtering) can help to overcome the aliasing effects created by sub-sampling. Use high-detail images to illustrate these effects. How much blurring do you need for a given sub-sampling factor. Is the median filter a good filter for anti-aliasing?

Extra credit (+10%)

Implement high-quality filters for the smoothing, such as bi-linear and Gaussian.