

Lab Assignment 1 – CSE 332/564, Spring 2008

Due: Monday, February 16, 11:59pm

This lab will get you started with the GUI, OpenGL, and digital image manipulation. Please check out the course lab page for skeleton code, image samples, and instructions on the programming environment. 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. Through the course of the semester, you will be extending this GUI first with image processing functionalities, and then with 3D visualization, flow visualization, and information visualization capabilities.

1. Explore the provided skeleton software distribution

Compile and check out the provided code to learn about the use of OpenGL and the event-driven programming style of FLTK. Compare with the notes discussed and provided in class.

2. Add menu items to the GUI

Use FLUID to add two menu items to the GUI: One that allows capturing the right GUI window (with the image) and one that allows capturing the left GUI window (with the transfer function, not used at the moment). Check out the file Application.cpp and go to the two routines WriteTransScreen() and WritePicScreen(). These are the routines your GUI callbacks will need to call. With FLUID, inspect gui.fl to see what happens when the File→ Read menu item is selected. Replicate this for the window capturing. Use FLUID's Write Code functionality to write a corresponding gui.h and gui.cpp file.

3. Add support for rectangular images

Try to load a non-square image and you will see it distorted. Check out the code in DisplayWindow(), look for DisplayWindow::draw() and read the comments there. You need to change the way a polygon is specified to support rectangular images, using its width and height. Right now the size is fixed to square. But make sure your image still fits into the display window. Try a few provided images and some of your own.

4. Implement interactive zooms/magnification (and reduction)

Check out DisplayWindow() and see how the image translations are being implemented and facilitated by the mouse interactions (using the m_translate vector). Now use the m_scale variable to control the zoom. See the comments in the code to get some idea how to do this.