HW1 document

Huayan Sun

How to run:

Please test my program on Zoolab computer. Clicking the “GLSLExperiment.sln” should open Visual Studio and then you can run the program in normal way (Ctrl + F5). Note that there is something different between my laptop and Zoolab computers. At Zoolab, when I asked for a window by glutInitWindowSize (640, 480), it actually gave me a window with size (632, 472) instead of (640, 480) on my laptop, so I modified the codes accordingly for Zoolab in my reshape() function to make sure it functions good on Zoolab computers.

I started HW1 on base of the starting codes given by Professor, which contains GLEW/GLUT as well as some starting header files and source files that I need. I have a folder named "data" which contains all .dat files. My main program is in example1.cpp, and I have vertex shader (vshader1.glsl) and fragment shader (fshader1.glsl).

Program starts at main() in example.cpp. It first initializes basic settings and creates a window. Then it reads all .dat files all at once and stores all points to an array by calling readPolylineFiles(). Then it creates GPU buffers and connects this .cpp file to shader file. It sets point size to 2 pixels and set default color to be red. Then it registers callback functions. Then it enters the drawing loop.

My programs can do everything that is required by HW1 on course website, and I won’t copy and paste the behavior requirements here. In my mouse function, it captures the selected thumbnail in state p and captures the point coordinate in state e. In my keyboard function, I used switch statement to switch between different states and goes to display() function. In display() function, it first cleans the window, and calls different functions according to the current state via switch statement.

In vertex shader (vshader1.glsl), I defined a uniform mat4 Proj for window-to-viewport mapping by generating a 4x4 matrix Proj and making gl\_Position = Proj \* vPosition. In fragment shader (fshader1.glsl), I defined a uniform vec4 myColor for color control by making fColor = myColor.