Sanciangco, Alex
704050064

I was not able to implement full OpenCL functionality. I tried for hours to get my program as well as the example programs to compile – messing with the gcc flags, changing the PATH variable, hardcoding the links to the cl.h file in the #include, searching the internet for a solution, etc. – and was unable to get the program to work properly. In addition it was difficult to properly test my program without the data from the /usr/local/cs133 directory.

I eventually was able to get my code to compile, but by that time I was unable to run any tests on the server. Every time I tried to create a context, the program would just hang. When I looked, it appeared that there always seemed to be other students using the server resources.

Instead, I implemented the standard sequential speed-up techniques from the previous lab that I used as well as a few from the TA. Again, without the proper data, I wasn't able to test the implementation. On a personal note, I find the content of this class extremely interesting. CS 180 allowed us to solve problems in a more creative way that lowered overall runtime, and this class just builds on that. It allows us to take those ideas we learned in CS 180 and combine them with the natural flow of a computer to achieve even greater speed up. I know that distributed computing is the future and I am slightly disappointed I wasn't able to have a tangible result; a step into the future, if you will.

My method for parallelizing using a GPU would be to divide the work of a single "slice" of the 3d image into a workgroup for each core of the GPU. In my code I have commented out the setup for the OpenCL context, which treats 'A' as a floating point array containing the image data and 'B' as the array of the updated image. Each workgroup would take a few lines from the slice and perform the necessary operations on them in parallel, writing the resulting data into the return array B. I have commented out the code I originally had to set up the environment. But without knowing if the GPU set up was successful, I wasn't able to set up a context in the scope of this lab.

Potential problems that could have arisen were getting the kernel to function properly and accounting for overhead of sending data to devices. Obviously, however, the biggest issue was getting OpenCL to work properly on my account, which I wasn't able to do until it was too late. Then the biggest issue became sharing resources with the other eager students in our class.

Since I wasn't able to test my program on the necessary data, I do not have specific performance results from the GPU.