Parallelizing The Game of Life

Samuel Murtaugh
smurtau@g.clemson.edu
C16058866

Jeremy Stacy
jcstacy@g.clesmon.edu
C10849481

Introduction

The goal of our project was to implement Conway's game of life using CUDA and openGL. We use CUDA to handle processing the rules of the game and to update the game's "grid". We use openGL to present a rendering of the grid, so that the results of the game can be watched in real time. We create two versions of this program; One running with OpenGL on the CPU, and another running with CUDA on the GPU.

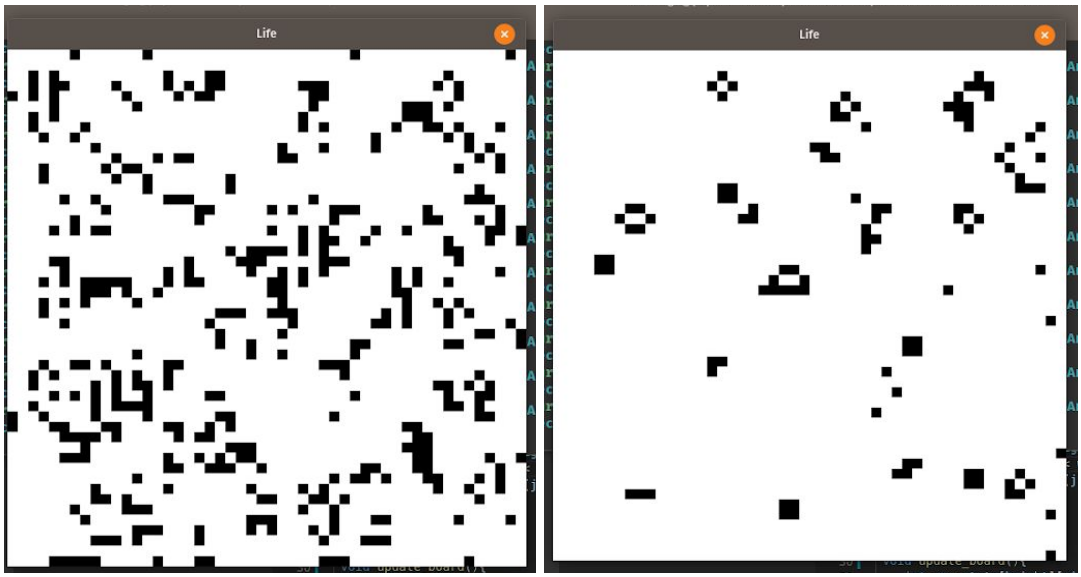
What is the Game of Life

The game of life is a cellular automaton proposed by mathematician John Horton Conway[2]. The "game" is played by supplying the configuration of a grid, where each cell is flagged as "alive" or "dead." The grid is then updated according to the following rules[1]:

1. Any living cell with 0 or 1 live neighbors dies(due to underpopulation).
2. Any living cell with 2 or 3 neighbors stays alive(because its environment is ideal).
3. Any living cell with more than 3 neighbors dies (because of overpopulation).
4. Any dead cell with exactly 3 neighbors becomes alive (because of reproduction.

The game can be used to create some visually striking visualizations and complex patterns.

We began our project by first creating a crude version of the game in C running on the CPU. We simply use a randomly generated 2D array to represent our grid and our cells, and then use numerous if checks to count the "living cells" and update the board. We then parallelized and optimized this program so that it could be made to run on the gpu, and accommodate vastly larger grids and provide better performance. The CPU version uses almost the same OpenGL renderer we used in the final version. Below are some examples of a few frames of a game of life from our CPU simulation, with a grid size of 50x50.



Translating the game into CUDA

There were several challenges when creating the code on the CPU over the GPU. Firstly, we had to convert our 2D array to a 1D array. This was as simple as changing how we allocated memory and using cuda dimension and thread indices to calculate the correct offsets. Secondly, we needed to set up our program to have one tile (cell) of the grid being checked against the rules per thread, while still accessing its neighbors. We did this by making sure the old board was shared. Ideally, each thread would correspond to a single cell on the grid. In cases where the grid size is larger than the total number of threads available, each thread would be responsible for multiple cells. Thirdly, we had to figure out how to use CUDA and OpenGL together. Ultimately we decided that simply calling the kernel in the OpenGL update function, so that it would run on each iteration of the main loop, was the simplest solution.

Our Approach to Rendering

We use openGL to render the game of life. In our display callback function, we create a simple 2D grid view of the same size as our game of life array. We render each square either white (for a dead cell) or black (for a live cell). The view is updated every second. We place the kernel call for our parallelized game of life inside the update function. We provide a reshape callback function to fit our grid to the size of our window, which is 600x600 by default in our CPU version. Below are some screenshots from the game of life running on a GPU; we use a much larger size grid (1000) and window (1000x1000)than in our CPU version to make better use of the hardware.

Conclusion

We were ultimately able to recreate the game of life on both the CPU and the GPU. The CUDA, GPU version of the program is capable of rendering much larger size games than the CPU version. Both versions of the program render and display the game in real-time, at one second intervals. OpenGL is used to render both versions of the program.

References

[1] Heaton, R. (2018, July 20). *Programming Projects for Advanced Beginners #2: Game of Life*. Robert Heaton. https://robertheaton.com/2018/07/20/project-2-game-of-life/

[2] Wikipedia contributors. (2020, November 28). *Conway's Game of Life*. Wikipedia. https://en.wikipedia.org/wiki/Conway%27s_Game_of_Life