ECE 4270: Computer Architecture, Fall 2021
Lab 4: CUDA Programming

Scope

CUDA is great for any compute intensive task, and that includes image processing. In this lab, you will be writing a CUDA program for basic image processing. A single high-definition image can have over 2 million pixels. Many image processing algorithms require dozens of floating-point computations per pixel, which can result in slow runtime even for the fastest of CPUs. The slow speed of a CPU is a serious hindrance to productivity. In CUDA, we can generally spawn exactly one thread per pixel. Each thread will be responsible for calculating the final color of exactly one pixel. Since images are naturally two dimensional, it makes sense to have each thread block be two dimensional.

Most image processing algorithms do something like the following

```
for (int i=0; i < height; i++)
{
        for (int j=0; j < width; j++)
        {
                result[i*width+j] = ProcessPixel(j,i);
        }
}
```

In order to convert this to CUDA, you literally just replace the for loops with a simple calculation involving each thread's id and block id.

```
int i = blockIdx.y * blockDim.y + threadIdx.y;
int j = blockIdx.x * blockDim.x + threadIdx.x;
```

The following papers are meant to give you clarity in this assignment:

1. Parallel Image Processing Based on CUDA, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4722322&tag=1
2. Image and Video Processing on CUDA: State of the Art and Future Directions, http://www.wseas.us/e-library/conferences/2011/Catania/Catania-07.pdf

You can access other materials to facilitate your understanding on this topic.

Tasks

1. In this lab, write a CUDA program that processes images using any filter of your choice. You can look into box filter, sobel filter, gaussian filter, etc. for your implementation.
2. In addition to any source image of your choice, you program should generate at two variants of the image (e.g, blur, sobel, etc.) depending on the implemented filters.

Grading Rubric

Code: CUDA program, and solutions to the problems (75)
Report: 25 points

Code and CUDA programs (75 points):

In order to get a full credit for the code, your CUDA program should be able to take an image or video of your choice and demonstrate steps in processing the image or video. Add a readme file detailing how you program has to run as well as the input used in testing your program.

Lab report (25 points):
Your report should give details about the work distribution within the group (who did what), milestones in your work and your implementation decisions (why did you choose the way you did it, and/or how did you do that).

Submission

One submission per group is required. Please, generate a pdf file of your report and name it lab4_report_groupX.pdf, then place it into the folder called lab4_groupX (where X is your group number). The folder lab4_groupX should also contain the src/ and input/ folders that contain your CUDA code you wrote for the problems, respectively. Then, please compress the lab4_groupX folder as lab4_groupX.tar.gz and submit it through the Canvas.

Due Date

Your lab is due on:
11/30 for Section I (before the lab)
12/3 for Section II (before the lab)