

Assignment 2

Due: 11:59 pm Monday, February 19th. 2018

Purpose: The primary purpose of this assignment is for you to learn how to use shared and constant memory to optimize your program.

Target Machines: For this assignment you will use one of the GPUs made available in ECE. Instructions for accessing these GPUs can be found on the via the Resources page and at <http://ece8823-sy.ece.gatech.edu/resources/accessing-gpus-on-the-ece-machines/>. If the PACE instructional cluster comes online in the interim, you will have access to those machines.

Assignment: Implement the CUDA kernel for 2D convolution to perform image blur on an RGB image. The CUDA source software framework has been provided courtesy of UIUC and the sequential CPU code has also been provided as a functional reference. Your goal is to implement the kernel and to do this efficiently – defined as minimizing the execution time as best you can by exploring the interactions between *at least* the following elements.

1. Grid and block dimensions
2. Use constant memory
3. Use shared memory

These are the minimum requirements. You may wish to employ any other optimizations that you wish.

Execution elements are as follows.

4. Use the software infrastructure provided you. You only need to add the kernel code.
5. The input will be an RGB image which will be converted into a 256x256 float matrix. You will also receive a filter matrix as input.
6. Your kernel should implement 2D convolution on the input matrix using the filter matrix.
7. You will have to add the device side memory allocation, memory copy, and kernel calls. Note that the grid and block dimensions will not be provided as input.
8. The provided script, `img2tex.py` will convert the input image to a text file.

Input file format generated by the script:

Image_width Image_height R0 G0 B0 R1 G1 B1 R2 G2 B2 ...

9. Print the output matrix to a file in the same format. The provided script, `tex2img.py` will convert this output into a png image for viewing.
10. Your programs will be executed with the following sequence of command lines.
 - a. Script to convert image into matrix: `python img2tex.py image_file input_text_file`
 - b. Execute your code: `cat input_text_file filter.txt | ./a.out output_text_file`
 - c. Script to convert output file into an image: `python tex2img.py output_text_file image_file`
11. Submit a report with the following elements.
 - a. A concise description of the algorithmic approach that describes i) how you made use of constant memory and ii) shared memory.

- b. The final execution time results that include the reductions due to the use of each
i) constant memory, and ii) shared memory

Grading Guidelines

For your information here are the grading guidelines

- Program compiles without errors (and appears to be correct): 25 points
- Program executes correctly for provided input: 25 points
- Program executes correctly across a set of input images: 35 points
- Project report: 15 points
- This assignment will also be graded on performance, so you are free to exploit various kernel optimization techniques to improve the run time performance of your kernel. The top 10 assignments will be given 5 bonus points. If you have 15 bonus points at the end of the course, you do not have to take the final.

Submission Guidelines:

All program submissions should be electronic. Submit a zip file with i) the complete software infrastructure including files that were not modified, and ii) the PDF files of the report. The zip file should be named *<last_name>.Assignment-2.zip*. Submissions must be time stamped by midnight on the due date. Submissions will be via T-square.

Note: No late assignments will be graded. Remember, you are expected to make a passing grade on the assignments to pass the course!