



CSC447-PARALLEL PROGRAMMING PROJECT

Project Name: GPU Image processing Batch Launcher

Due Date: 18 May

Team : 2 Students

The Goal of this project is to realize a GPU-Cuda image processing tools, that can process a batch of jobs defined in an input text file. A Job consist of an input file name, processing algorithm name and output file name. Many image processing Algorithm can be applied to a picture, some are pixel based (like brightness, color to grayscale or RGB channels correction.), while others are neighbors based, like filters processing where pixel calculation require looking to local neighbors pixels (ex. blurring, edge detection, sharpening,...).

Jobs are executed sequentially by the Batch Launcher, one by one. But scheduling optimization are permitted in order to reduce memory management overhead for GPU memory allocation and copying.

Indications:

- Cuda Threads are allocated in a 2D grids of 2D blocks, where blocks are square and with parametrized size.
- Memory tiling optimization must be used when this apply
- Image format of the pictures is supposed in PNG on 3 Bytes, 1 Byte per RGB channel (this simplify managing any possible formats).
- Processing algorithm to be implemented are : brightness, grayscale, channel correction, blurring, sharpening and edge detection. More about those algorithm is provided in the bottom wiki section.
- Your project will get better appreciation if it reach a record score in execution time for some preselected jobs list.
- Technical details about image processing implementing is provided in the project wiki section.



Project wiki

1. Image Processing Resources

- > Gray Scale conversion :
https://www.tutorialspoint.com/dip/grayscale_to_rgb_conversion.htm
- > Brightness processing : <https://www.allaboutcircuits.com/technical-articles/digital-image-processing-point-operations>
- > Blurring, Sharpening, Edge detection :
[https://en.wikipedia.org/wiki/Kernel_\(image_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing))

2. C language resources

- > Reading text file : <https://solarianprogrammer.com/2019/04/03/c-programming-read-file-lines-fgets-getline-implement-portable-getline/>
- > C Structure : https://www.w3schools.com/c/c_structs.php

3. Cuda

- > Profiling with nvprof command line :
<https://developer.nvidia.com/blog/cuda-pro-tip-nvprof-your-handy-universal-gpu-profiler/>
- > Cuda Memory Model : <https://www.3dgep.com/cuda-memory-model/>