

RGB color to Grayscale image using CUDA

Hirokatsu (Hiro) Suzuki

Problem:

Use CUDA and luminosity method to convert RGB color image to Grayscale image.
The code should read in a jpg image file using OpenCV library. A kernel function was used to read the image as pixel which contains 3 chars (red, green, blue). Each pixel is converted to grayscale using luminosity method with equation of $NewImage = 0.3R + 0.59G + 0.11B$.



Solution

I used CUDA C++ code to convert RGB to Grayscale image.
Compiled with “nvcc -g <cuda file> -I/home/USERNAME/<path to link> -L/home/USERNAME/<path to library> -lstdc++ -lopencv_imgcodecs -lopencv_core”.

Test image



Pixel: 1697x2434

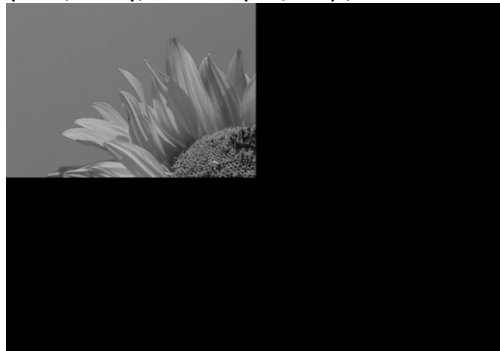
Result Images



Grid: (152, 105), Block: (32, 32) , 0.000014sec



Grid: (152, 105), Block: (16, 16), 0.000010sec



Grid: (152, 105), Block: (8, 8), 0.000013sec



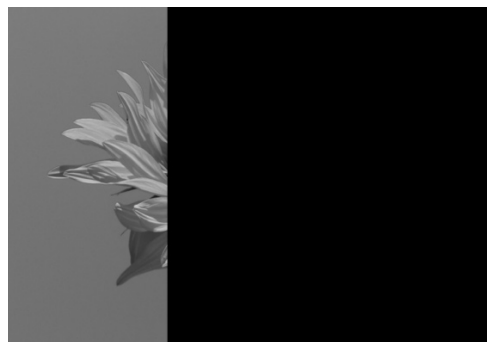
Grid: (152, 105), Block: (4, 4), 0.000013sec



Grid: (50, 50), Block: (32, 32), 0.000012sec



Grid: (100, 50), Block: (32, 32), 0.000011sec



Grid: (25, 100), Block: (32, 32), 0.000012sec



Grid: (200, 200), Block: (32, 32), 0.000011sec

Gray image conversion was tested on different grid and block sizes which are shown above. We can see that with enough (more than) grid (152x105) and Block (32x32) sizes, the image will be completely converted to grayscale. Using less values less than these numbers will result in partial conversions.