ICS-E4020: Week 4 - Correlated pairs GPU

Néstor Castillo García (472081)
nestor.castillogarcia@aalto.fi

May 18, 2015

1 Correlated pairs in GPU CP9 optimised

1.1 Description

Using CUDA, an efficient working solution that solved the image correlation problem on the GPU was done.

1.2 Hardware

The computers had the following specifications: Intel Xeon E3-1230v2, 4 cores, 8 thread, 3,3 GHz, RAM: 16 GB, GPU: Nvidia K2000.

1.3 Performance

As the focus of this exercise in performance, it was heavily optimised. This GPU version did the correlated pairs task of a 4000 x 4000 image in less than 1s; 10 times faster than a eight-threaded. The focus of this approach was to maximise the number of arithmetic operation per memory transfer and also to used shared memory per block of threads because it is much more faster than the global memory.

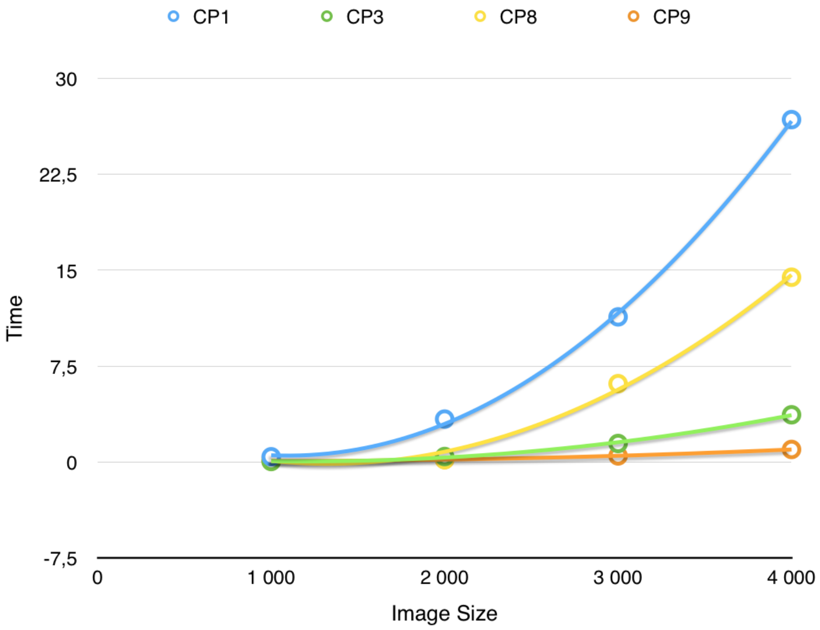
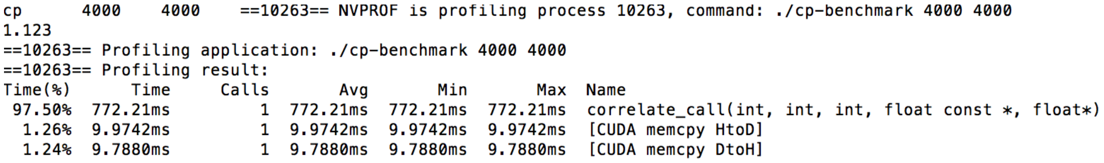


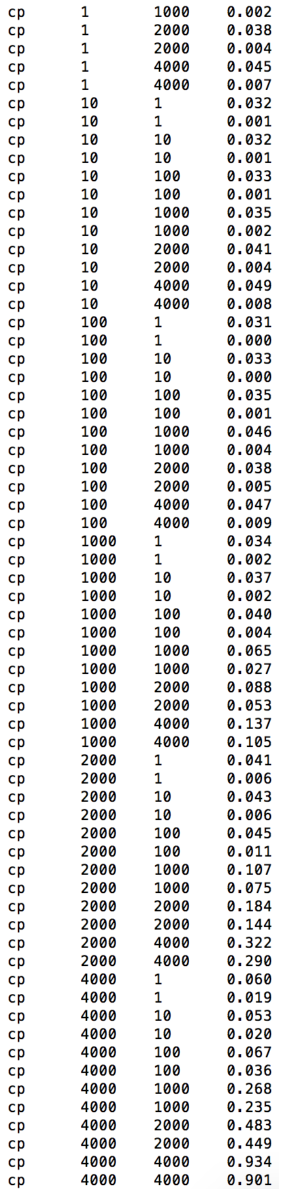
Figure 1: Scalability comparison between single threaded, multithreaded, multithreades vectorized and GPU implementations

The code was profiled with nvidia profiler and it was found that the kernel execution time occupies 97,5% of the total time. Thus, the cudaMemCopy instructions occupy a small amount of time compared to the kernel execution.



```
cp      4000    4000    ==10263== NVPROF is profiling process 10263, command: ./cp-benchmark 4000 4000
1.123
==10263== Profiling application: ./cp-benchmark 4000 4000
==10263== Profiling result:
Time(%)      Time     Calls       Avg       Min       Max  Name
 97.50%  772.21ms         1  772.21ms  772.21ms  772.21ms  correlate_call(int, int, int, float const *, float*)
  1.26%  9.9742ms         1  9.9742ms  9.9742ms  9.9742ms  [CUDA memcpy HtoD]
  1.24%  9.7880ms         1  9.7880ms  9.7880ms  9.7880ms  [CUDA memcpy DtoH]
```

Figure 2: Detailed execution time in a 4000 x 4000 image in comparison to cp8 in cp9 memcopy appears to be a little more relevant



```
cp      1       1000    0.002
cp      1       2000    0.038
cp      1       2000    0.004
cp      1       4000    0.045
cp      1       4000    0.007
cp      10      1       0.032
cp      10      1       0.001
cp      10      10      0.032
cp      10      10      0.001
cp      10      100     0.033
cp      10      100     0.001
cp      10      1000    0.035
cp      10      1000    0.002
cp      10      2000    0.041
cp      10      2000    0.004
cp      10      4000    0.049
cp      10      4000    0.008
cp      100     1       0.031
cp      100     1       0.000
cp      100     10      0.033
cp      100     10      0.000
cp      100     100     0.035
cp      100     100     0.001
cp      100     1000    0.046
cp      100     1000    0.004
cp      100     2000    0.038
cp      100     2000    0.005
cp      100     4000    0.047
cp      100     4000    0.009
cp      1000    1       0.034
cp      1000    1       0.002
cp      1000    10      0.037
cp      1000    10      0.002
cp      1000    100     0.040
cp      1000    100     0.004
cp      1000    1000    0.065
cp      1000    1000    0.027
cp      1000    2000    0.088
cp      1000    2000    0.053
cp      1000    4000    0.137
cp      1000    4000    0.105
cp      2000    1       0.041
cp      2000    1       0.006
cp      2000    10      0.043
cp      2000    10      0.006
cp      2000    100     0.045
cp      2000    100     0.011
cp      2000    1000    0.107
cp      2000    1000    0.075
cp      2000    2000    0.184
cp      2000    2000    0.144
cp      2000    4000    0.322
cp      2000    4000    0.290
cp      4000    1       0.060
cp      4000    1       0.019
cp      4000    10      0.053
cp      4000    10      0.020
cp      4000    100     0.067
cp      4000    100     0.036
cp      4000    1000    0.268
cp      4000    1000    0.235
cp      4000    2000    0.483
cp      4000    2000    0.449
cp      4000    4000    0.934
cp      4000    4000    0.901
```

Figure 3: Benchmark results

I