M2-BIG DATA
GPGPU - Chapter 8

Exercice 2

Réalisé par:
Gaby Maroun

Encadré par:
Dr. Etancelin JM

February 7, 2021

Objectives

Improve the convolution kernel from previous exercise.

Instructions

From your previous program, implement the following elements :

- tile in shared memory, using dynamic shared memory allocation (extern __shared__ float tile[]; in the kernel and the size in byte is given as the third parameter of «<»> kernel call syntax).
- threads grid dimensions must be computed from output tile size and block dimensions is computed from tile size.
- Implement the constraints on tile and output tile in the kernel
- Do not forget appropriate threads synchronizations.

Questions

1. **How many floating operations are being performed in your convolution kernel ? explain.**
   There are
   $$channels(2 \times maskWidth^2)$$
   *floating operations for cycling through the mask twice, one for + and the other to * while taking into consideration each color. It's obvious that there are less floating operations in this exercise than the previous one.*
2. **How many global memory reads are being performed by your kernel ? explain.**
   There are
   $$channels(2 \times maskWidth^2)$$
   *global memory reads for cycling to read through the mask twice, one for + and the other to * while taking into consideration the colors*
3. **How many global memory writes are being performed by your kernel ? explain.**
   There are
   $$imgCols \times imgRows \times channels$$
   global memory writes for writing the image for different colors
4. **Compute the arithmetic intensity of the kernel.**
   The arithmetic intensity is a FLOP/Byte number standing for the number of floating point operations performed per byte of global memory accessed.

$$\frac{channels(2 \times maskWidth^2)}{channels(2 \times maskWidth^2) + imgCols \times imgRows \times channels}$$

$$=> \frac{1}{1 + \frac{imgCols \times imgRows \times channels}{channels(2 \times maskWidth^2)}}$$

$$=> \frac{1}{1 + \frac{imgCols \times imgRows}{2 \times maskWidth^2}} (FLOP/Byte)$$

5. **Measure the kernel computational time of the kernel, using the profiler. Then, compute the computational power of the kernel (in GFLOPS). Compare with the CPU version given.**

Sequential Version:

```
Read image of size 512x512 3 channels
Convolution run in 2.34505 s.
Write image 512x512 3 colors into LenaSeqBlur.png
```

*As it seems, it takes 2.34505seconds for the kernel to compile with a 25*25 mask on a 512x512 image size.*

Parallel version:

```
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap8/Exo2$ make
make: Avertissement : le fichier « 2-tiledConvolutionCPU.cu » a une date de modification 133 s dans le futur
nvcc -c  2-tiledConvolutionCPU.cu -o 2-tiledConvolutionCPU.o
nvc++ -fPIC -c  img_utils.cxx -o img_utils.o
nvcc  2-tiledConvolutionCPU.o img_utils.o -o 2-tiledConvolutionCPU   `pkg-config --libs opencv` -lm
make: AVERTISSEMENT : décalage d'horloge détecté. La construction peut être incomplète.
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap8/Exo2$ nvprof --print-gpu-trace ./2-tiledConvolutionCPU Lena.png Lena2Blurprof.png
Read image of size 512x512 3 channels
==29927== NVPROF is profiling process 29927, command: ./2-tiledConvolutionCPU Lena.png Lena2Blurprof.png
Write image 512x512 3 colors into Lena2Blurprof.png
==29927== Profiling application: ./2-tiledConvolutionCPU Lena.png Lena2Blurprof.png
==29927== Profiling result:
   Start  Duration            Grid Size      Block Size     Regs*    SSMem*    DSMem*      Size  Throughput  SrcMemType  DstMemType           Device   Context    Stream
  Name
333.87ms  299.80us                    -               -         -         -         -  3.0000MB  9.7721GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
334.18ms  1.2800us                    -               -         -         -         -  2.4414KB  1.8190GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
336.11ms  2.0817ms          (16 16 1)       (32 32 1)        54  12.250KB        0B         -           -           -           -  Quadro RTX 4000         1         7
  convolution_2D_tiled_kernel(float*, float const *, float*, int, int, int) [112]
338.20ms  558.42us                    -               -         -         -         -  3.0000MB  5.2464GB/s      Device    Pageable  Quadro RTX 4000         1         7
  [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or tools and can be more than what the compiler sh
ows.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
DstMemType: The type of destination memory accessed by memory operation/copy
```

*As it seems, it takes 2.0817milliseconds for the kernel to compile with a 25*25 mask on a 512x512 image size.*

For the computational power of the kernel for the GPU version, it is equal to :

$$\frac{FloatingOperations}{ExecutionTime}$$

$$=> \frac{channels(2 \times maskWidth^2)}{2.0817ms}$$

So,

$$=> \frac{channels(2 \times maskWidth^2)}{2.0817 \times 10^3 s} =? \frac{imgCols \times imgRows \times channels(2 \times maskWidth^2)}{2.34505 s}$$

$$=> \frac{1}{2.0817 \times 10^3 s} =? \frac{imgCols \times imgRows}{2.34505 s}$$

$$=> \frac{1}{imgCols \times imgRows \times 2.0817 \times 10^3 s} < \frac{1}{2.34505 s}$$

That means, the kernel's computation power of this exercise is $10^3 \times imgCols \times imgRows$ *more powerful than the CPU's.*

6. **Compare the computational power evolution using different images sizes. Compare with the evolution from previous version. Compare with the theoretical power obtained from chapter 2 exercise 2 ? Give an explanation.**

Ivy :

```
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap8/Exo1$ nvprof --print-gpu-trace ./1-convolutionCPU Ivy.png Ivy1Blur.png
Read image of size 605x750 3 channels
==1572== NVPROF is profiling process 1572, command: ./1-convolutionCPU Ivy.png Ivy1Blur.png
Write image 605x750 3 colors into Ivy1Blur.png
==1572== Profiling application: ./1-convolutionCPU Ivy.png Ivy1Blur.png
==1572== Profiling result:
   Start  Duration            Grid Size      Block Size     Regs*    SSMem*    DSMem*      Size  Throughput  SrcMemType  DstMemType           Device   Context    Stream
  Name
331.76ms  534.36us                    -               -         -         -         -  5.1928MB  9.4900GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
332.31ms  1.2800us                    -               -         -         -         -  2.4414KB  1.8190GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
334.18ms  4.2393ms            (19 24 1)       (32 32 1)        38        0B        0B         -           -           -           -  Quadro RTX 4000         1         7
  convolution_1D_basic_kernel(float*, float const *, float*, int, int, int) [112]
338.42ms  1.6282ms                    -               -         -         -         -  5.1928MB  3.1145GB/s      Device    Pageable  Quadro RTX 4000         1         7
  [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or tools and can be more than what the compiler sh
ows.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
DstMemType: The type of destination memory accessed by memory operation/copy
```

*it takes the kernel from exercise 1, 4.2393 milliseconds to compile with a 25*25 mask on a 605x750 image size*

```
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap8/Exo2$ nvprof --print-gpu-trace ./2-tiledConvolutionCPU Ivy.png Ivy2cBlurprof.png
Read image of size 605x750 3 channels
==28961== NVPROF is profiling process 28961, command: ./2-tiledConvolutionCPU Ivy.png Ivy2cBlurprof.png
Write image 605x750 3 colors into Ivy2cBlurprof.png
==28961== Profiling application: ./2-tiledConvolutionCPU Ivy.png Ivy2cBlurprof.png
==28961== Profiling result:
   Start  Duration            Grid Size      Block Size     Regs*    SSMem*    DSMem*      Size  Throughput  SrcMemType  DstMemType           Device   Context    Stream
 Name
324.26ms  529.08us                    -               -         -         -         -  5.1928MB  9.5847GB/s    Pageable      Device  Quadro RTX 4000         1         7
 [CUDA memcpy HtoD]
324.80ms  1.3120us                    -               -         -         -         -  2.4414KB  1.7746GB/s    Pageable      Device  Quadro RTX 4000         1         7
 [CUDA memcpy HtoD]
326.55ms  3.4158ms            (19 24 1)       (32 32 1)        54  12.250KB        0B         -           -           -           -  Quadro RTX 4000         1         7
 convolution_2D_tiled_kernel(float*, float const *, float*, int, int, int) [112]
329.97ms  1.6691ms                    -               -         -         -         -  5.1928MB  3.0382GB/s      Device    Pageable  Quadro RTX 4000         1         7
 [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or tools and can be more than what the compiler sho
ws.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
DstMemType: The type of destination memory accessed by memory operation/copy
```

While for the exercise 2, 3.4158milliseconds is enough for the same image as we can start to see the improvement when increasing the size of the image.

Tiger4K :

```
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap8/Exo1$ nvprof --print-gpu-trace ./1-convolutionCPU tiger4k.png tiger4k1Blur.png
Read image of size 7680x4320 3 channels
==2168== NVPROF is profiling process 2168, command: ./1-convolutionCPU tiger4k.png tiger4k1Blur.png
Write image 7680x4320 3 colors into tiger4k1Blur.png
==2168== Profiling application: ./1-convolutionCPU tiger4k.png tiger4k1Blur.png
==2168== Profiling result:
   Start  Duration            Grid Size      Block Size     Regs*    SSMem*    DSMem*      Size  Throughput  SrcMemType  DstMemType           Device   Context    Stream
  Name
317.58ms  38.696ms                    -               -         -         -         -  379.69MB  9.5821GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
356.29ms  1.3440us                    -               -         -         -         -  2.4414KB  1.7324GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
358.44ms  255.44ms          (240 135 1)       (32 32 1)        38        0B        0B         -           -           -           -  Quadro RTX 4000         1         7
  convolution_1D_basic_kernel(float*, float const *, float*, int, int, int) [112]
613.89ms  143.45ms                    -               -         -         -         -  379.69MB  2.5848GB/s      Device    Pageable  Quadro RTX 4000         1         7
  [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or tools and can be more than what the compiler sh
ows.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
DstMemType: The type of destination memory accessed by memory operation/copy
```

*it takes the kernel from exercise 1, 255.44milliseconds to compile with a 25*25 mask on a 7680x4320 image size*

```
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap8/Exo2$ nvprof --print-gpu-trace ./2-tiledConvolutionCPU tiger4k.png Tiger2cBlurprof.png
Read image of size 7680x4320 3 channels
==29094== NVPROF is profiling process 29094, command: ./2-tiledConvolutionCPU tiger4k.png Tiger2cBlurprof.png
Write image 7680x4320 3 colors into Tiger2cBlurprof.png
==29094== Profiling application: ./2-tiledConvolutionCPU tiger4k.png Tiger2cBlurprof.png
==29094== Profiling result:
   Start  Duration            Grid Size      Block Size     Regs*    SSMem*    DSMem*      Size  Throughput  SrcMemType  DstMemType           Device   Context    Stream
 Name
293.10ms  38.628ms                    -               -         -         -         -  379.69MB  9.5990GB/s    Pageable      Device  Quadro RTX 4000         1         7
 [CUDA memcpy HtoD]
331.74ms  1.3440us                    -               -         -         -         -  2.4414KB  1.7324GB/s    Pageable      Device  Quadro RTX 4000         1         7
 [CUDA memcpy HtoD]
333.62ms  185.70ms          (240 135 1)       (32 32 1)        54  12.250KB        0B         -           -           -           -  Quadro RTX 4000         1         7
 convolution_2D_tiled_kernel(float*, float const *, float*, int, int, int) [112]
519.32ms  144.94ms                    -               -         -         -         -  379.69MB  2.5582GB/s      Device    Pageable  Quadro RTX 4000         1         7
 [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or tools and can be more than what the compiler sho
ws.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
DstMemType: The type of destination memory accessed by memory operation/copy
```

While for the exercise 2, 185.70milliseconds is all what it takes to compile which prove the improvement clearly.

The following profiler represent the profile of the kernel from exercise 3 chapter 3, compiled on the Lena.png with BLURSIZE=25

```
gmaroun@scinfe054:/import/etud/3/gmaroun/Bureau/stockage/Semestre 3/GPGPU/Chap3/Chap3Ex3$ nvprof --print-gpu-trace ./3-imgToBlur Lena.png LenaBlurprof.png
Read image of size 512x512 3 channels
==5000== NVPROF is profiling process 5000, command: ./3-imgToBlur Lena.png LenaBlurprof.png
Write image 512x512 3 colors into LenaBlurprof.png
==5000== Profiling application: ./3-imgToBlur Lena.png LenaBlurprof.png
==5000== Profiling result:
   Start  Duration            Grid Size      Block Size     Regs*    SSMem*    DSMem*      Size  Throughput  SrcMemType  DstMemType           Device   Context    Stream
  Name
363.95ms  278.97us                    -               -         -         -         -  3.0000MB  10.502GB/s    Pageable      Device  Quadro RTX 4000         1         7
  [CUDA memcpy HtoD]
364.23ms  6.3908ms             (16 16 1)       (32 32 1)        30        0B        0B         -           -           -           -  Quadro RTX 4000         1         7
  blurKernel(float*, float*, int, int, int) [110]
370.63ms  561.11us                    -               -         -         -         -  3.0000MB  5.2213GB/s      Device    Pageable  Quadro RTX 4000         1         7
  [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or tools and can be more than what the compiler sh
ows.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
```

We can assume that compared to the execution time from the previous question 5 of the Lena.png, the improved parallel code takes less time (2.0817ms< 6.3908ms). I believe that the difference in time is caused by the tiled mask used and the less number of floating operations.

La fin.