

Reporte Técnico
Gualberto Casas Medina - A00942270
Septiembre 2018

Introducción:

Aunque los procesadores de hoy en día son multinúcleo y permiten el procesamiento de múltiples instrucciones al mismo tiempo, comúnmente no se explota esta característica de los procesadores. Al no utilizar los varios núcleos disponibles, el procesador tiene mucho potencial que no se está utilizando. Para poder utilizar los varios núcleos de un procesador ya sea cpu o gpu, es necesario programar instrucciones que puedan correr en paralelo.

En la siguiente actividad realizamos un *blur* sobre una imagen. Para esto se desarrollan tres algoritmos diferentes:

1. CPU sin threads
2. CPU con threads (omp)
3. GPU con threads (cuda)

Desarrollo:

Hay un archivo principal para éste proyecto y se llama *main.cu* el cual podemos compilar con el Makefile usando el comando *make*. El resultado del comando *make* es un archivo de nombre *main.exe*. Este programa corre cada uno de los dos algoritmos en CPU así como también el algoritmo en GPU. Se utiliza el archivo *common.h* para llamar de manera segura las funciones de CUDA.

```
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ls
LICENSE Makefile README.md common.h image.jpg main.cu main.exe main.o
```

Resultados:

```
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 169.066742 ms
Time passed (CPU OMP): 113.426659 ms
Time passed (GPU CUDA): 41.523968 ms
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 109.131096 ms
Time passed (CPU OMP): 109.290382 ms
Time passed (GPU CUDA): 46.063347 ms
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 130.312439 ms
Time passed (CPU OMP): 109.313400 ms
Time passed (GPU CUDA): 40.788433 ms
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 159.153732 ms
Time passed (CPU OMP): 109.760780 ms
Time passed (GPU CUDA): 49.202095 ms
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 108.579704 ms
Time passed (CPU OMP): 109.510864 ms
Time passed (GPU CUDA): 45.045517 ms
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 109.080490 ms
Time passed (CPU OMP): 109.246819 ms
Time passed (GPU CUDA): 39.047955 ms
A00942270@alien1-lab:~/.../assignment-2-image-blurring-gualcm$ ./main.exe
Starting...
Time passed (CPU NO THREADS): 108.610321 ms
Time passed (CPU OMP): 109.666069 ms
Time passed (GPU CUDA): 37.130993 ms
```

Los promedios de los experimentos fueron los siguientes:

CPU No Threads = 127.428 ms

CPU Threads = 109.57 ms

GPU Threads = 42.428 ms

Características del Sistema:

GPU:

- GeForce GTX 670
- Cuda Cores: 1344
- 915 MHz Graphics Clock

- 980 MHz Processor Clock
- Memory Capacity 2GB

CPU:

- Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz
- Max Frequency: 3900 MHz
- Min Frequency: 800 MHz
- Cores: 4

Conclusión:

A diferencia de la actividad anterior en la que los resultados entre *CPU No Threads* y *CPU Threads* fueron muy similares en pruebas con matrices de tamaño pequeño, en este experimento podemos ver una mejora del 15%. El tiempo de procesamiento cuando utilizamos el GPU se reduce drásticamente a aproximadamente 30% de lo que tomaría si hacemos la misma operación en *CPU No Threads*.