

Reporte técnico

César Armando Valladares Martínez

Agosto 2018

Resumen

El siguiente documento es un reporte de la segunda tarea de la materia de programación multinúcleo donde se realiza una comparación entre el tiempo transcurrido de una operación que aplica un filtro *Blur* a una imagen, la comparación se realiza entre usando una CPU, una CPU con threads y una GPU modificando la cantidad de Threads y Blocks.

Introducción

La programación paralela se utiliza actualmente para realizar procesos de manera más rápida utilizando GPU en lugar de CPU, la ventaja de las GPU es que pueden realizar procesos de manera simultánea, es decir en paralelo, utilizando los hilos (Threads) que se dividen por bloques, específicamente hablando de las GPU de NVIDIA. Pero existen casos en los que realizar un proceso de forma paralela no es más rápido que utilizando solo la CPU.

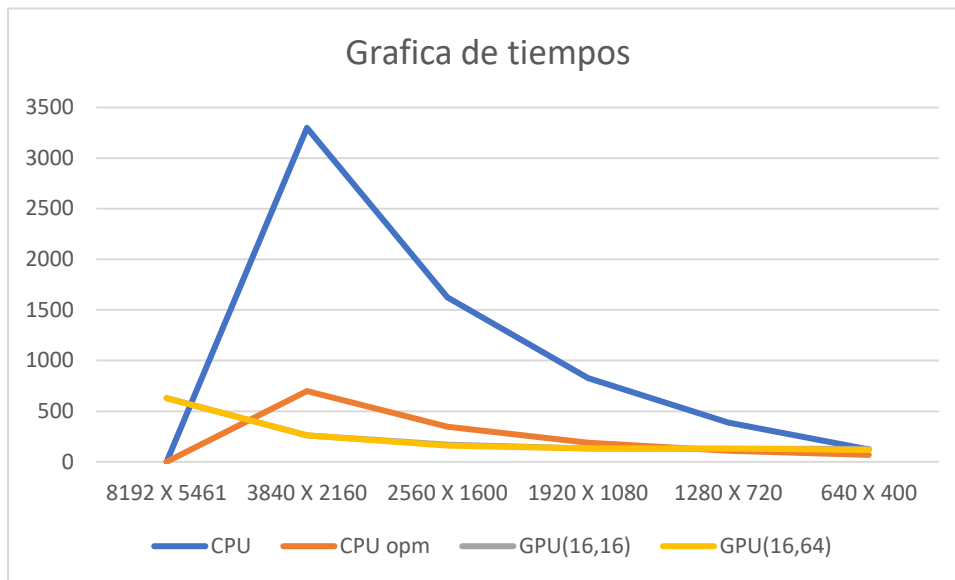
Para este ejercicio debemos tomar en cuenta que el procesador es un i7 con 6 núcleos físicos y 6 virtuales de 2.2.Ghz y una GPU GTX 1050 Ti 4GB

Desarrollo

Cabe mencionar que los programas en CPU no aguantaron una imagen con resolución 8K, se desconoce la razón pues con imágenes de menor resolución ambos programas se ejecutaron correctamente.

En la siguiente tabla podremos observar los tiempos de ejecución de cada proceso.

Tiempos de ejecución						
	8192 X 5461	3840 X 2160	2560 X 1600	1920 X 1080	1280 X 720	640 X 400
CPU	Segmentation fault	3298.396 ms	1625.115 ms	828.729 ms	387.592 ms	123.637 ms
CPUopm	Segmentation fault	698.679 ms	347.765 ms	187.243 ms	109.502 ms	66.840 ms
GPU(16,16)	629.748 ms	262.399 ms	170.222 ms	131.330 ms	127.624 ms	129.897 ms
GPU(16,64)	627.422 ms	261.946 ms	160.052 ms	132.694 ms	130.752 ms	116.562 ms



Speed Up						
	8192 X 5461	3840 X 2160	2560 X 1600	1920 X 1080	1280 X 720	640 X 400
CPU/CPUomp	NA	4.720903305	4.673026325	4.425954508	3.539588318	1.849745661
CPU/GPU(16/16)	NA	12.57015461	9.547032698	6.310279449	3.036983639	0.951807971
CPU/GPU(16/64)	NA	12.59189299	10.15366881	6.245414261	2.964329418	1.060697311
CPUomp/GPU(16/16)	NA	2.662658775	2.043008542	1.425744308	0.858004764	0.514561537
CPUomp / GPU(16/64)	NA	2.667263482	2.172825082	1.41108867	0.837478585	0.573428733
GPU(16/16)/GPU(16/64)	1.003707234	1.001729364	1.063541849	0.989720711	0.976076848	1.114402636

Análisis de resultados

En la tabla **Tiempos de ejecución** se muestra que entre las dos configuraciones de cd GPU no existe una gran diferencia de tiempos, pero se puede apreciar que la configuración de <<16,64>> es un mínimo más rápido que la configuración <<16,16>>

En la gráfica de esta misma tabla apreciamos como todos los tiempos cada vez se van acercando más, el que un proceso tarde más en GPU que en CPU con imágenes de baja resolución puede deberse a que toma más tiempo realizar el apartado de memoria y el movimiento de todos los datos a comparación de un proceso secuencial.

En la tabla **Speed Up** podemos ver mejor la comparación entre los tiempos de ejecución, los números en las casillas amarillas representan aquellos tiempos que no mejoraron al hacer el proceso en paralelo. Estas casillas se inclinan más hacia las imágenes con menor resolución,

Conclusión

Podemos concluir en base a las tablas anteriores que no siempre es más óptimo paralelizar procesos, ya que hay casos en los que toma más tiempo realizarlos de manera paralela por el apartado y manejo de memoria que realizarlo de manera secuencial.