GPU Programming Lab Assignment

Nilanka Manoj Rathnayake – 150532E

1. Device Details

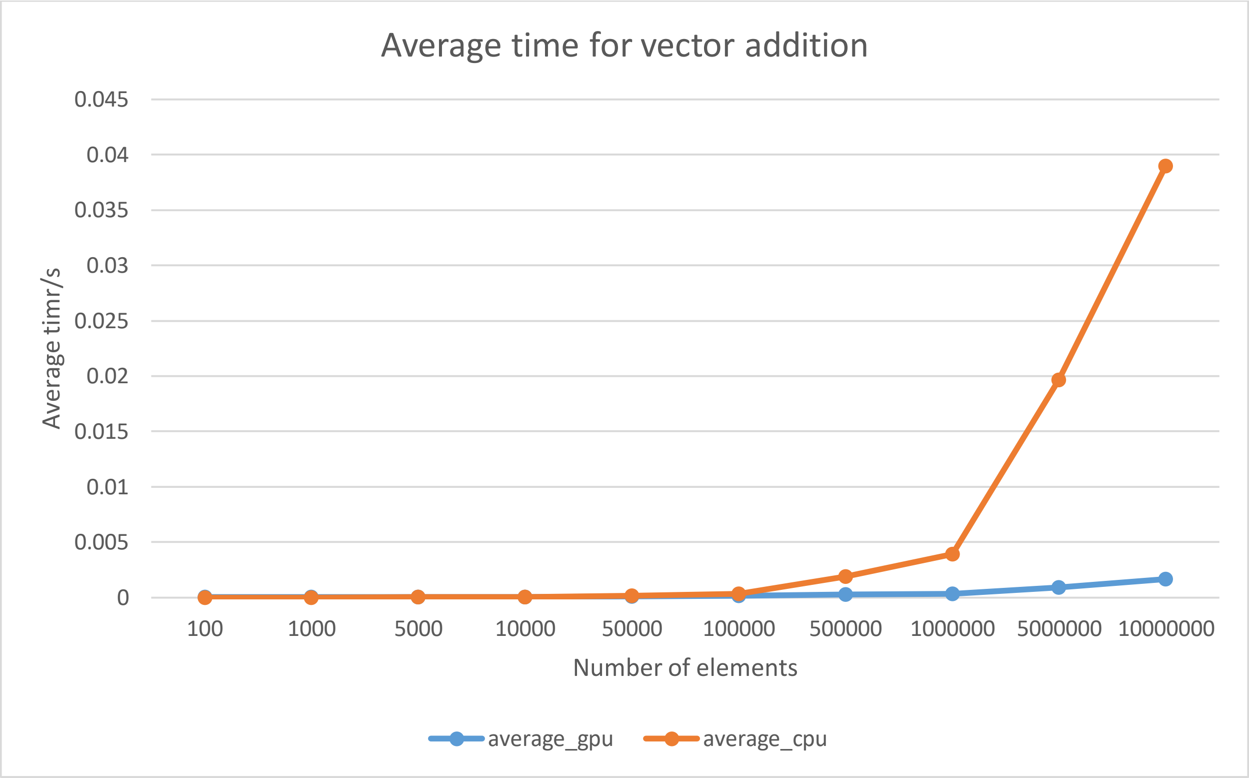
property	description	value
name	Name of the device	GeForce GTX 480
Compute capability	The general specifications and features of a compute device depend on its compute capability	2
total global memory(KB)	total amount of global memory available on the device in bytes	1509504
shared mem per block	maximum amount of shared memory available to a thread block	49152
regs per block	maximum number of 32-bit registers available to a thread block	32768
warp size	number of threads in a warp	32
max threads per block	maximum number of threads per block	1024
max thread dim	maximum size of each dimension of a block	x:1024 y:1024 z:64
max grid size	maximum size of each dimension of a grid	z:65535 y:65535 x:65535
total constant memory (bytes)	total amount of constant memory available on the device in bytes	65536
multiprocessor count	number of multiprocessors on the device	15
memory bus width	Size of memory bus	384
memory clock rate (KHz)	Clocking rate for memory bus synchronizations	1848000
L2 cache size (bytes)	Level2 cache size in bytes	786432
max threads per SM	Maximum thread count for a shared memory space	1536

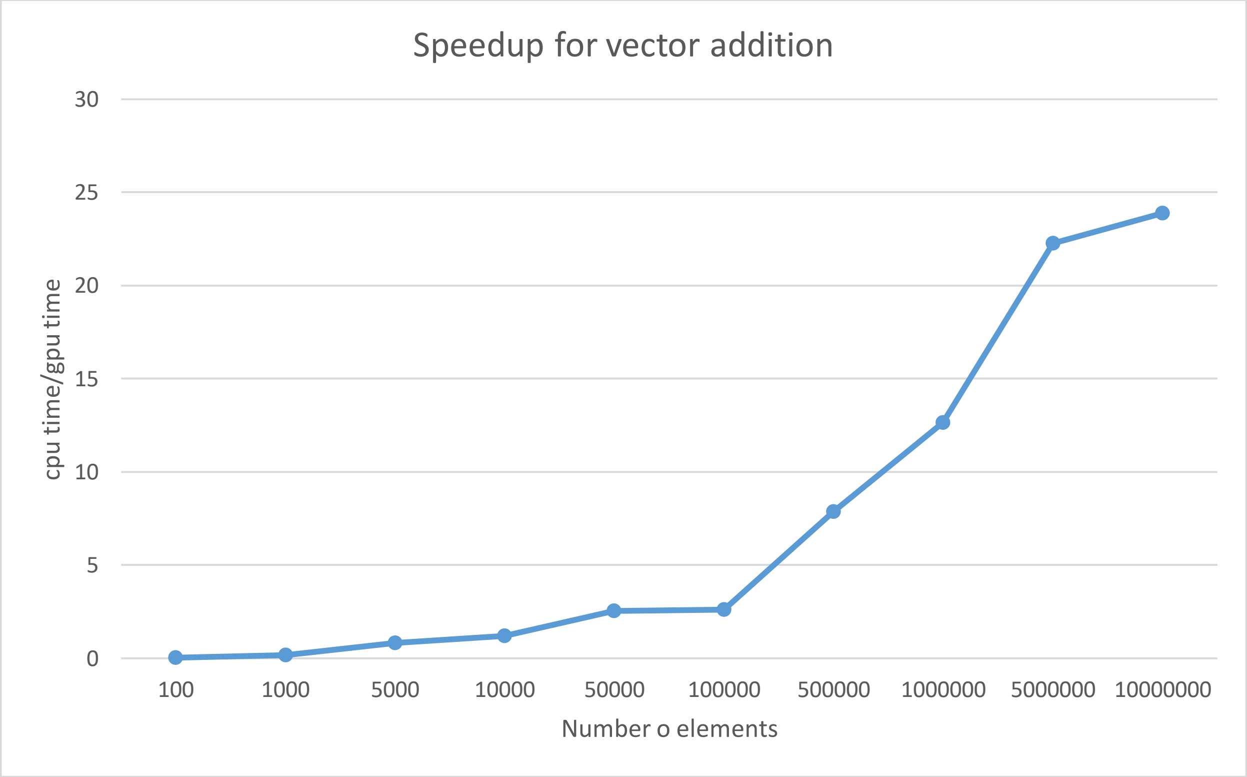
2. Vector Addition

GPU						
size	attempt1	attempt2	attempt3	attempt4	attempt5	average_gpu
100	0.000034	0.000033	0.00003	0.000027	0.000031	0.000031
1000	0.000033	0.000033	0.000031	0.000032	0.000029	0.0000316
5000	0.000029	0.000032	0.00003	0.000028	0.00003	0.0000298
10000	0.000029	0.000032	0.000029	0.000033	0.00003	0.0000306
50000	0.000064	0.000069	0.000069	0.000066	0.000072	0.000068
100000	0.000132	0.00013	0.000133	0.000133	0.000134	0.0001324
500000	0.000245	0.000244	0.000248	0.00024	0.000244	0.0002442
1000000	0.000313	0.0003	0.000305	0.00031	0.000311	0.0003078
5000000	0.000886	0.000878	0.000884	0.000881	0.000881	0.000882
10000000	0.001633	0.001632	0.001633	0.001639	0.001631	0.0016336

CPU						
size	attempt1	attempt2	attempt3	attempt4	attempt5	average_cpu
100	0.000001	0.000001	0.000001	0.000001	0.000001	0.000001
1000	0.000006	0.000005	0.000006	0.000005	0.000006	0.0000056
5000	0.000025	0.000021	0.000024	0.000024	0.000026	0.000024
10000	0.000035	0.000038	0.000033	0.000044	0.000033	0.0000366
50000	0.000171	0.000164	0.000185	0.000166	0.00017	0.0001712
100000	0.000336	0.000333	0.000334	0.000358	0.000352	0.0003426
500000	0.001912	0.0019	0.001932	0.001936	0.001903	0.0019166
1000000	0.00389	0.003914	0.003846	0.003872	0.003929	0.0038902
5000000	0.019676	0.019634	0.01949	0.019749	0.019622	0.0196342
10000000	0.038884	0.039018	0.039045	0.038851	0.039081	0.0389758

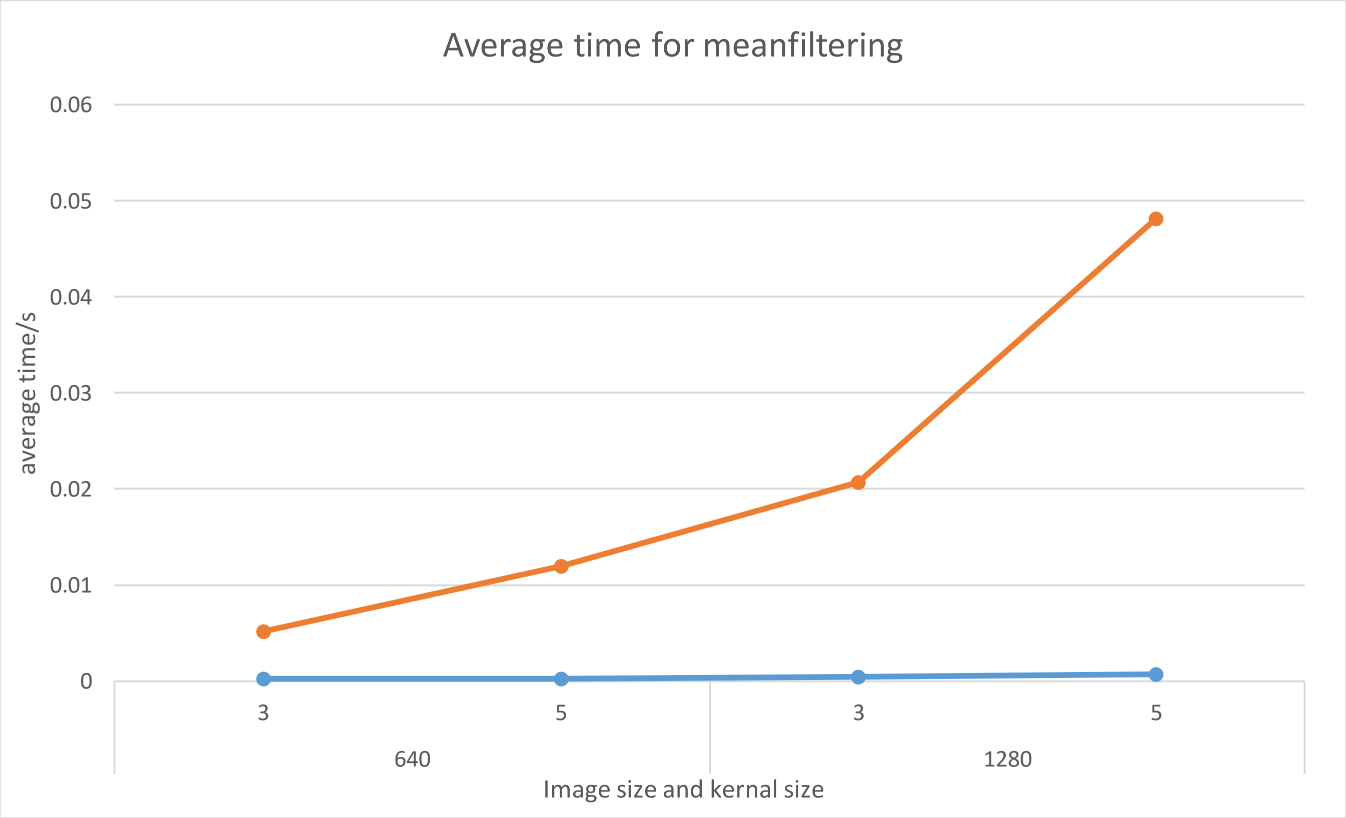
Speed up						
size	attempt1	attempt2	attempt3	attempt4	attempt5	avg_speed_up
100	0.029412	0.030303	0.033333	0.037037	0.032258	0.032258065
1000	0.181818	0.151515	0.193548	0.15625	0.206897	0.17721519
5000	0.862069	0.65625	0.8	0.857143	0.866667	0.805369128
10000	1.206897	1.1875	1.137931	1.333333	1.1	1.196078431
50000	2.671875	2.376812	2.681159	2.515152	2.361111	2.517647059
100000	2.545455	2.561538	2.511278	2.691729	2.626866	2.587613293
500000	7.804082	7.786885	7.790323	8.066667	7.79918	7.848484848
1000000	12.42812	13.04667	12.60984	12.49032	12.63344	12.63872645
5000000	22.20767	22.36219	22.04751	22.41657	22.27242	22.26099773
10000000	23.81139	23.90809	23.90998	23.70409	23.96137	23.85883937

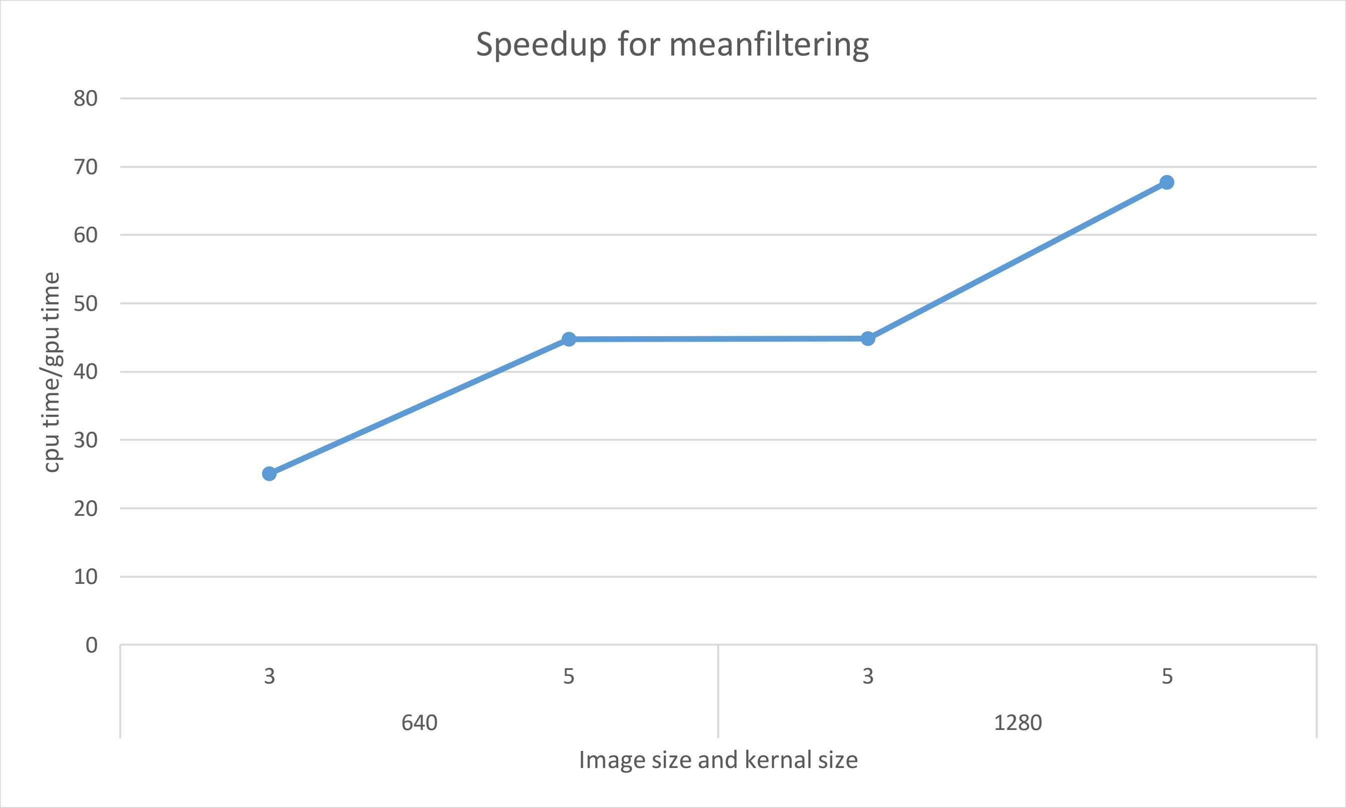

Average time for vector addition
Average timr/s
0.045
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
100
1000
5000
10000
50000
100000
500000
1000000
5000000
10000000
Number of elements
average_gpu
average_cpu


Speedup for vector addition
cpu time/gpu time
30
25
20
15
10
5
0
100
1000
5000
10000
50000
100000
500000
1000000
5000000
10000000
Number o elements

3. Mean filter

		GPU					
size	kernal	attempt1	attempt2	attempt3	attempt4	attempt5	average_gpu
640	3	0.000203	0.000214	0.000202	0.000208	0.000206	0.000207
	5	0.000273	0.000262	0.000267	0.000271	0.000267	0.000268
1280	3	0.000458	0.000465	0.000459	0.000458	0.000466	0.000461
	5	0.000712	0.000716	0.00071	0.000717	0.000698	0.000711
		CPU					
size	kernal	attempt1	attempt2	attempt3	attempt4	attempt5	average_cpu
640	3	0.005177	0.005182	0.005181	0.005179	0.005187	0.005181
	5	0.011978	0.012013	0.011988	0.011979	0.011981	0.011988
1280	3	0.020686	0.020691	0.020674	0.020631	0.02065	0.020666
	5	0.048136	0.048172	0.048155	0.048122	0.048124	0.048142
		Speedup					
size	kernal	attempt1	attempt2	attempt3	attempt4	attempt5	avg_speed_up
640	3	25.50246	24.21495	25.64851	24.89904	25.17961165	25.07841
	5	43.87546	45.85115	44.89888	44.20295	44.87265918	44.7306
1280	3	45.16594	44.49677	45.04139	45.04585	44.31330472	44.81006
	5	67.60674	67.27933	67.82394	67.11576	68.94555874	67.7481





- **Speed up is lowest for image size 640 and kernel size 3** –
  - CPU iterations by loop is lower than 1280 size that parallelized by GPU. (reason for speedup (640,3) < speedup (1280,3)).
  - Adding more numbers in GPU is faster than CPU because large cache memory and higher bandwidth in GPU. For kernel size 3: only 9 additions and for kernel size 5: 25 additions. (reason for speedup (640,3) < speedup (640,5)).
- **Speed up is highest for image size 1280 and kernel size 5** –
  - CPU iterations by loop is higher than 640 size that parallelized by GPU. (reason for speedup (640,5) < speedup (1280,5)).
  - Adding more numbers in GPU is faster than CPU because large cache memory and higher bandwidth in GPU. For kernel size 3: only 9 additions and for kernel size 5: 25 additions. (reason for speedup (1280,3) < speedup (1280,5)).