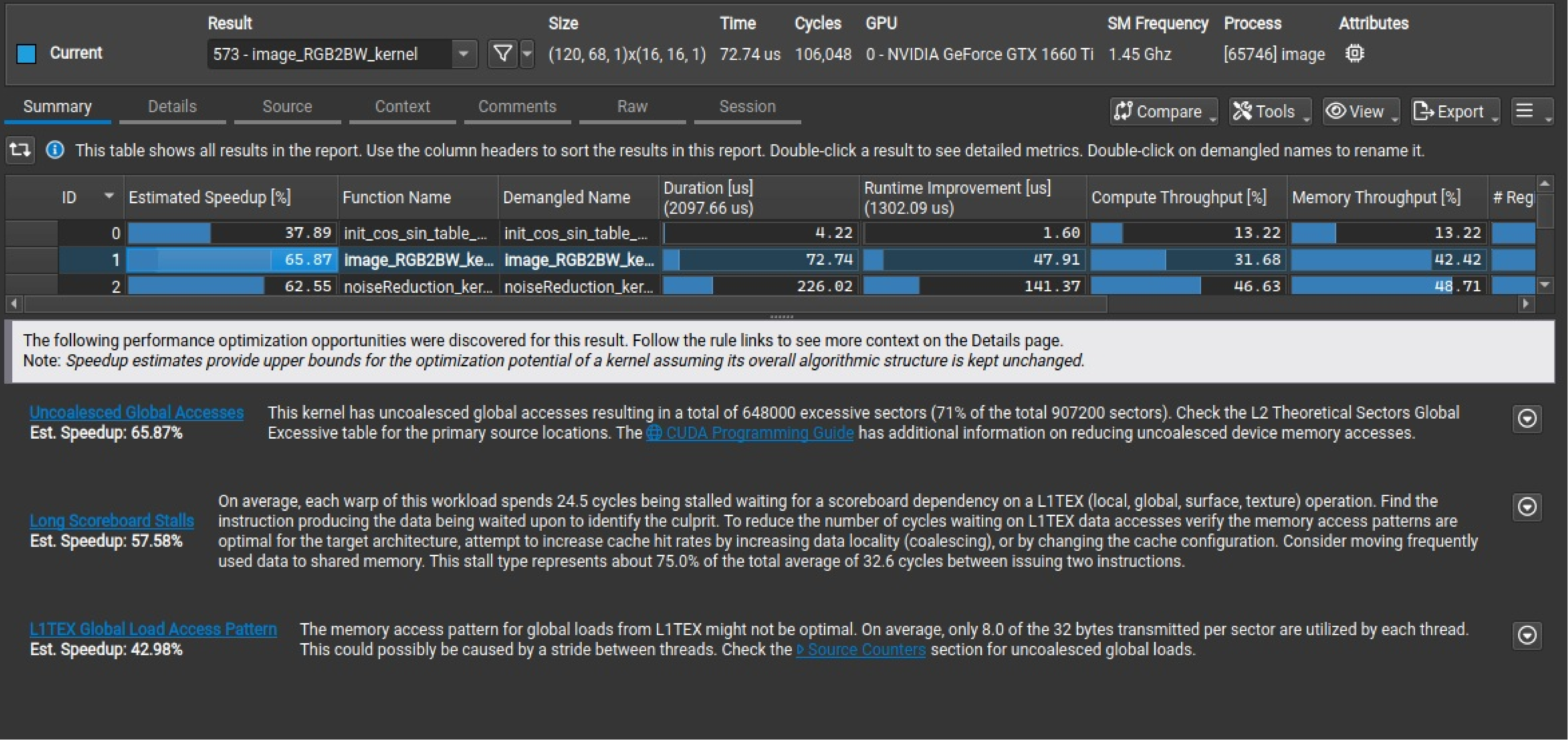

	Result	Size	Time	Cycles	GPU	SM Frequency	Process	Attributes
Current	573 - image_RGB2BW_kernel	(120, 68, 1)x(16, 16, 1)	72.74 us	106,048	0 - NVIDIA GeForce GTX 1660 Ti	1.45 Ghz	[65746] image	

This table shows all results in the report. Use the column headers to sort the results in this report. Double-click a result to see detailed metrics. Double-click on demangled names to rename it.

ID	Estimated Speedup [%]	Function Name	Demangled Name	Duration [us] (2097.66 us)	Runtime Improvement [us] (1302.09 us)	Compute Throughput [%]	Memory Throughput [%]	# Reg
0	37.89	init_cos_sin_table_...	init_cos_sin_table_...	4.22	1.60	13.22	13.22	
1	65.87	image_RGB2BW_ke...	image_RGB2BW_ke...	72.74	47.91	31.68	42.42	
2	62.55	noiseReduction_ker...	noiseReduction_ker...	226.02	141.37	46.63	48.71	

The following performance optimization opportunities were discovered for this result. Follow the rule links to see more context on the Details page.
Note: *Speedup estimates provide upper bounds for the optimization potential of a kernel assuming its overall algorithmic structure is kept unchanged.*

Uncoalesced Global Accesses
Est. Speedup: 65.87%

This kernel has uncoalesced global accesses resulting in a total of 648000 excessive sectors (71% of the total 907200 sectors). Check the L2 Theoretical Sectors Global Excessive table for the primary source locations. The CUDA Programming Guide has additional information on reducing uncoalesced device memory accesses.

Long Scoreboard Stalls
Est. Speedup: 57.58%

On average, each warp of this workload spends 24.5 cycles being stalled waiting for a scoreboard dependency on a L1TEX (local, global, surface, texture) operation. Find the instruction producing the data being waited upon to identify the culprit. To reduce the number of cycles waiting on L1TEX data accesses verify the memory access patterns are optimal for the target architecture, attempt to increase cache hit rates by increasing data locality (coalescing), or by changing the cache configuration. Consider moving frequently used data to shared memory. This stall type represents about 75.0% of the total average of 32.6 cycles between issuing two instructions.

L1TEX Global Load Access Pattern
Est. Speedup: 42.98%

The memory access pattern for global loads from L1TEX might not be optimal. On average, only 8.0 of the 32 bytes transmitted per sector are utilized by each thread. This could possibly be caused by a stride between threads. Check the ▷ Source Counters section for uncoalesced global loads.