	Result	Size	Time	Cycles	GPU	SM Frequency	Process	Attributes
Current	576 - noiseReduction_kernel	(120, 68, 1)x(16, 16, 1)	2.03 ms	2,943,230	0 - NVIDIA GeForce GTX 1660 Ti	1.44 Ghz	[11724] image	

Summary | Details | Source | Context | Comments | Raw | Session

Compare | Tools | View | Export

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 [ms] (6.45 ms)	Runtime Improvement [ms] (4.27 ms)	Compute Throughput [%]	Memory Throughput [%]	# Registers [regi
0	93.70	init_cos_sin_table_...	init_cos_sin_table_...	0.00	0.00	0.32	2.36	
1	85.54	image_RGB2BW_ke...	image_RGB2BW_ke...	0.23	0.20	87.88	12.78	
2	81.61	noiseReduction_ker...	noiseReduction_ker...	2.03	1.66	82.03	5.03	

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.*

FP64/32 Utilization
Est. Speedup: 81.61%

The ratio of peak float (fp32) to double (fp64) performance on this device is 32:1. The workload achieved close to 0% of this device's fp32 peak performance and 44% of its fp64 peak performance. If ▷ Compute Workload Analysis determines that this workload is fp64 bound, consider using 32-bit precision floating point operations to improve its performance. See the Kernel Profiling Guide for more details on roofline analysis.

Tex Throttle Stalls
Est. Speedup: 17.97%

On average, each warp of this workload spends 178.0 cycles being stalled waiting for the L1 instruction queue for texture operations to be not full. This stall reason is high in cases of extreme utilization of the L1TEX pipeline. Try issuing fewer texture fetches, surface loads, surface stores, or decoupled math operations. If applicable, consider combining multiple lower-width memory operations into fewer wider memory operations and try interleaving memory operations and math instructions. Consider converting texture lookups or surface loads into global memory lookups. Texture can accept four threads' requests per cycle, whereas global accepts 32 threads. This stall type represents about 71.5% of the total average of 248.8 cycles between issuing two instructions.

Uncoalesced Global Accesses
Est. Speedup: 17.72%

This kernel has uncoalesced global accesses resulting in a total of 2883680 excessive sectors (61% of the total 4755920 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.