DPS915 Presentation
Ray Tracing Parallelization

Soutrik Barua • Faiq Malik

Assignment Breakdown

Assignment 1

- Sudoku
- Ray Tracing

Assignment 2

- Porting code for CUDA
- CPU vs CPU+GPU execution times

Assignment 3

- Further optimization
- Final analysis and conclusion

Ray Tracing

What is Ray Tracing?

- Rendering Technique for generating an image by tracing the path of light.
- Simulating the effects of its trajectory path with virtual objects.

Practical applications

- To calculate and simulate path of particles and waves.
  - Optical Lens Design
  - Acoustics
  - Radio Signal Propagation
  - Astrophysics
  - many more

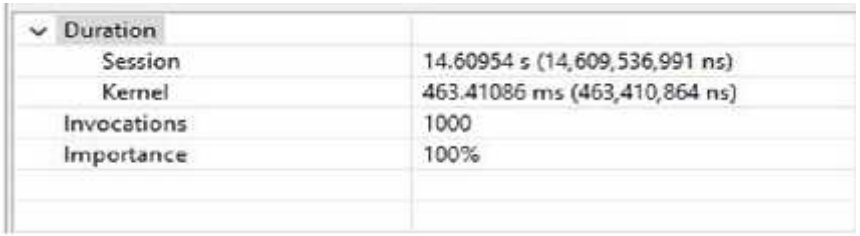
Progress

Accomplishments

- Optimized code for CUDA enabled device
- Decreased run time
- Enhanced image processing speed

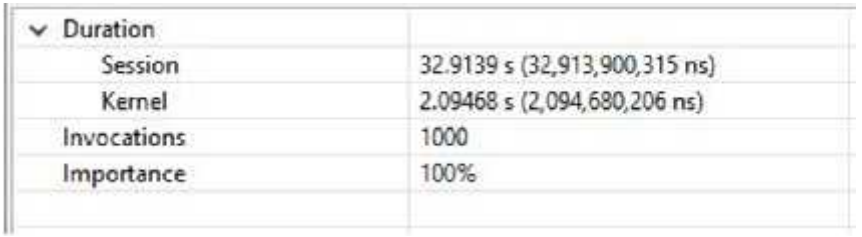
GPU runtimes by Image Resolution

Image Resolution at 512



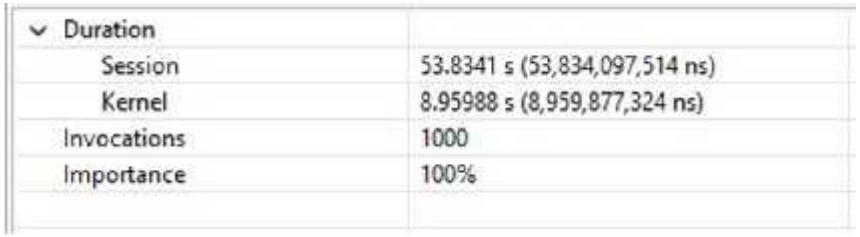
Duration	
Session	14.60954 s (14,609,536,991 ns)
Kernel	463.41086 ms (463,410,864 ns)
Invocations	1000
Importance	100%

Image Resolution at 1024



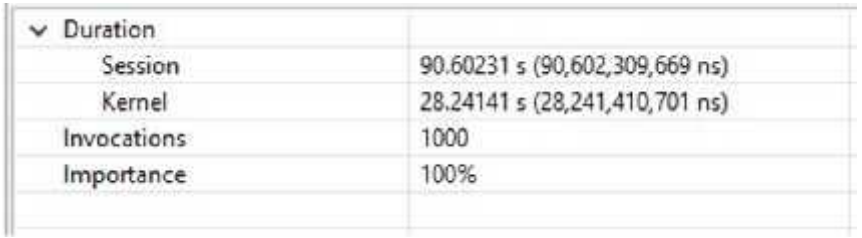
Duration	
Session	32.9139 s (32,913,900,315 ns)
Kernel	2.09468 s (2,094,680,206 ns)
Invocations	1000
Importance	100%

Image Resolution at 2048



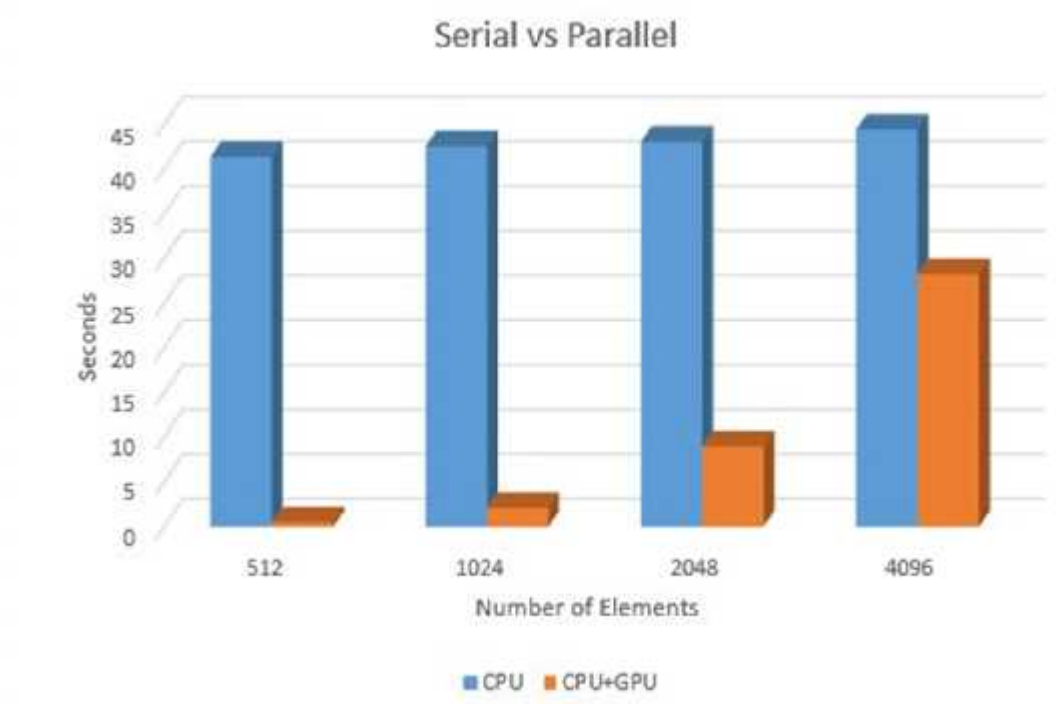
Duration	
Session	53.8341 s (53,834,097,514 ns)
Kernel	8.95988 s (8,959,877,324 ns)
Invocations	1000
Importance	100%

Image Resolution at 4096



Duration	
Session	90.60231 s (90,602,309,669 ns)
Kernel	28.24141 s (28,241,410,701 ns)
Invocations	1000
Importance	100%

CPU vs GPU


Serial vs Parallel
45
40
35
30
25
20
15
10
5
0
Seconds
512
1024
2048
4096
Number of Elements
CPU
CPU+GPU

Potential Ways to Optimize Further

- Use more available bandwidth
- Have better memcpy efficiency
- Better compute overlap
- More use of cores depending on GPU/compute capability
- Reduce memory access times.

End of Presentation

Questions?