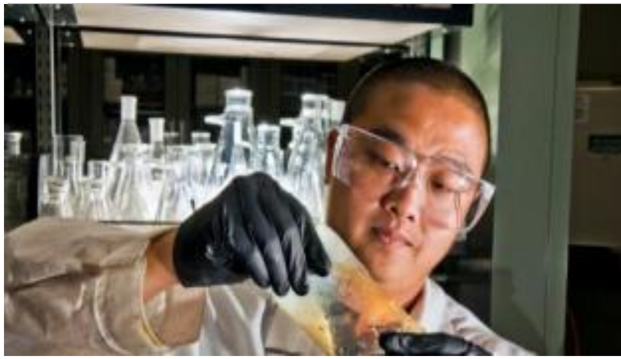
SAND2014-16872PE

Exceptional service in the national interest



A High-performance GPU-based Forward-projection Model for Computed Tomography Applications

Ismael Perez^{a}, Matthew Bauerle^{a}, Edward S. Jimenez^{a}and Kyle R. Thompson^{b}

Sandia National Laboratories, Software Systems and R&D^{a}

Sandia National Laboratories, Structural Dynamics and X−ray/NDE^{b}

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

Outline

- Introduction
  - Radiography (X-ray Imaging)
  - Imaging Operator
- Approach
  - Discretization of Continuous Space
  - Radiography Simulation
- Graphics Processing Unit (GPU) Computing and Implementation
  - Graphics Processing Unit (GPU) Advantages
- Results
- Conclusions

Introduction

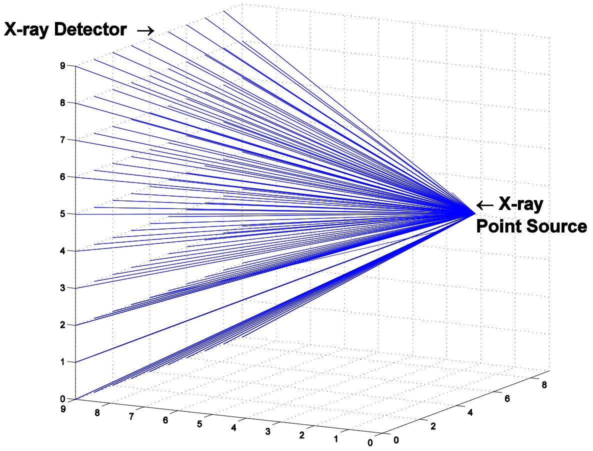
- Radiography
  - The formation of radiographs is more generally known as the Forward-projection imaging model
  - Imaging Operator
  - Discretized Imaging Operator
- Computational Challenge
  - Industrial Radiography
    - Teravoxel data sets
    - Better Image Resolution
  - Numerically unstable calculation
- GPU Computing
  - Higher memory bandwidth
  - Massive multi-threading parrelization

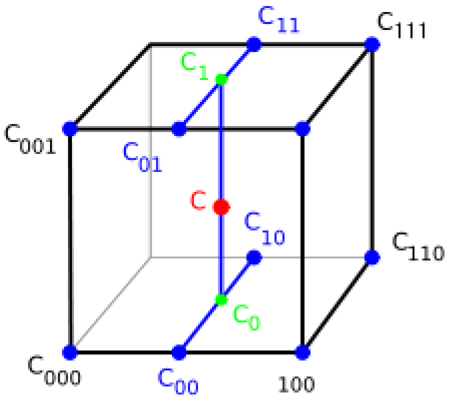
Approach

- Discretization of Continuous Space
  - Voxel-based discretization
- Radiograph Simulation
  - Lambert-Beer’s law of attenuation

$$I(\varepsilon) = I_o(\varepsilon)\exp\left(\int_{\vec{s}}^{\vec{p}} \frac{\mu(\varepsilon, x)}{\rho} x dx\right)$$

    - Mono-energetic energy levels
  - Ray tracing algorithm
    - Distance of each path length
  - Trilinear Interpolation
    - Approximate the continuous space





GPU Computing

- GPU advantages
  - Texture Memory
    - Cached on the GPU
    - Higher bandwidth by reducing request to off-chip DRAM
    - Accelerate Access Patterns
  - Hardware-based Texture Interpolation (for the trilinear interpolation)
    - Reduces memory traffic when reads have spatial locality
  - Fast Device Memory
    - Enables data to be fetched in less time
    - Coalesced memory fetches increase effective bandwidth
  - Massive Multi-threading
    - Launch tens of thousands of threads in parallel

GPU Implementation

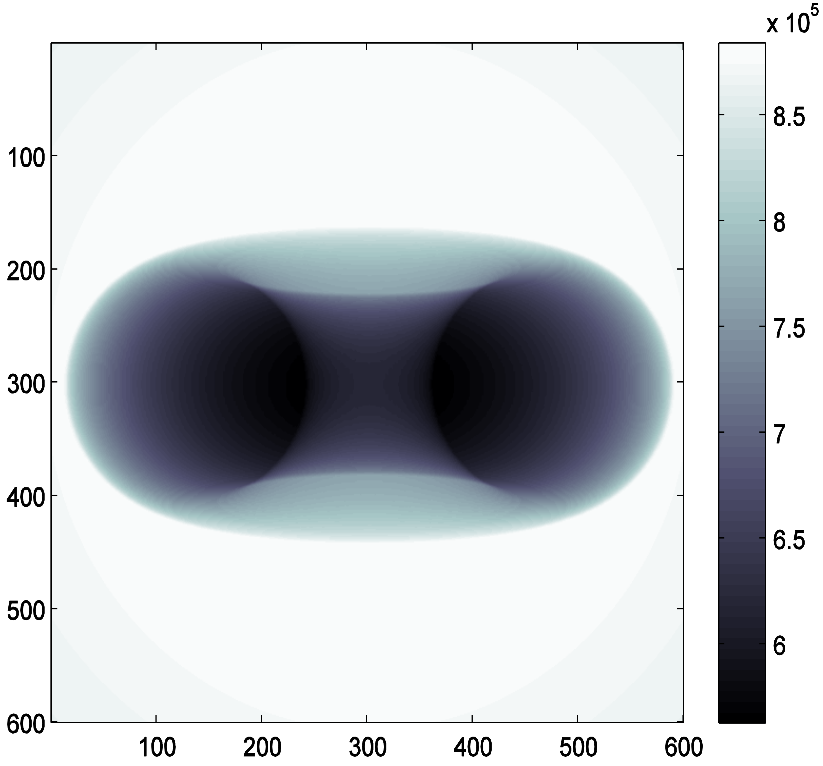
- Texture Memory
  - Store the object of interest in a texture-based array
- Hardware-based Texture Interpolation
  - Linear Filtering Mode (Trilinear Interpolation)
  - Approximation of the attenuation values of the object
- Massive Multi-threading
  - Each threads will calculate Lambert-Beer’s law for each ray path from the point source to detector
  - Avoids shared memory and thread conflicts or race conditions

Evaluation

- The numerical simulations were performed on two high-end-workstations.
    - Supermicro X9DRG-QF Motherboard, 512 GB DDR3 system memory, dual Intel Xeon E5-2687W octo-core processors clocked at 3.1 GHz with hyper-threading for a total of 32 CPU threads and Nvidia Tesla S2070 Device connected via PCI-E 2.0 x16 host interface card.
    - Dell Precision T7600, 16GB DDR3 system memory, dual Intel Xeon E5-2667 octo-core processors clocked at 2.9 GHz and Nvidia GeForce GTX 690 Device connected via PCI-E x16 host interface card. The GTX 690 unit contains two Kepler GK104 GPU with 4GB of GDDR5.
- The forward projection kernel was tested using datasets ranging from 100^3 voxels to 1000^3 voxels. The GPU kernel was written in CUDA (Version 5.0) and host code was written in C/C++ using Microsoft Visual Studio 2008.
- Performance metrics consist of average kernel runtime with respect to voxel sizes. This work also measured data transfers between host and device, and vice versa. This was done using Nvidia Nsight Visual Studio Edition 4.0.

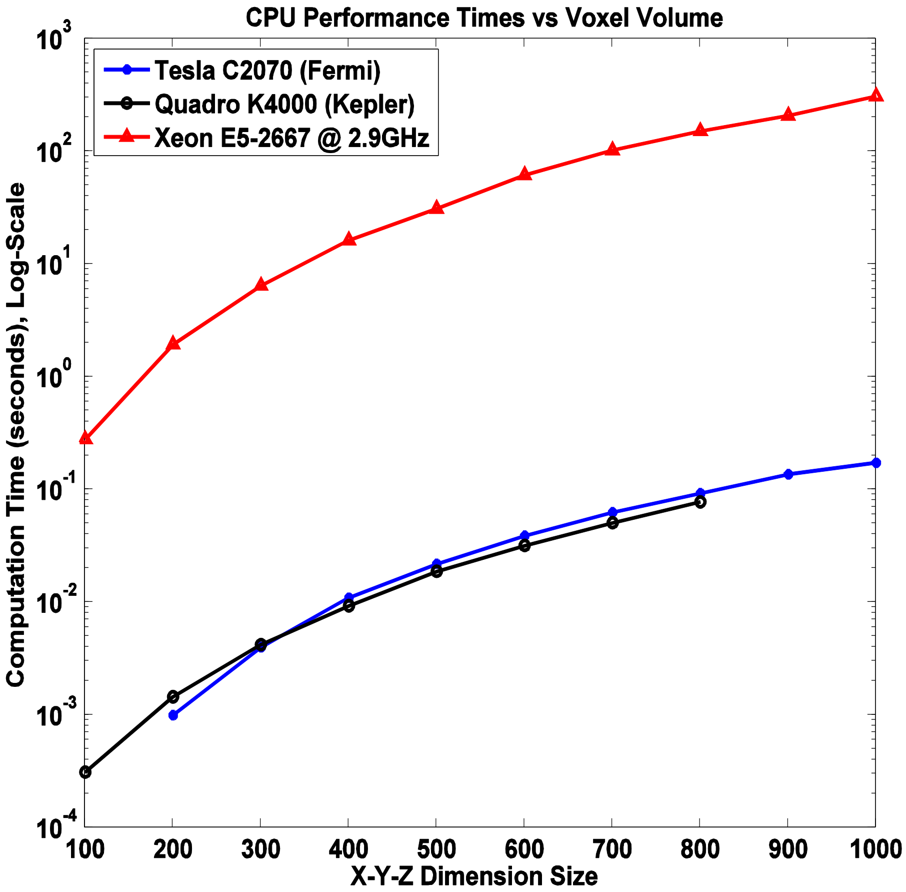
Results

- Simulated x-ray image of a homogenous torus in 3D using the GPU-based Forward-projection Model.



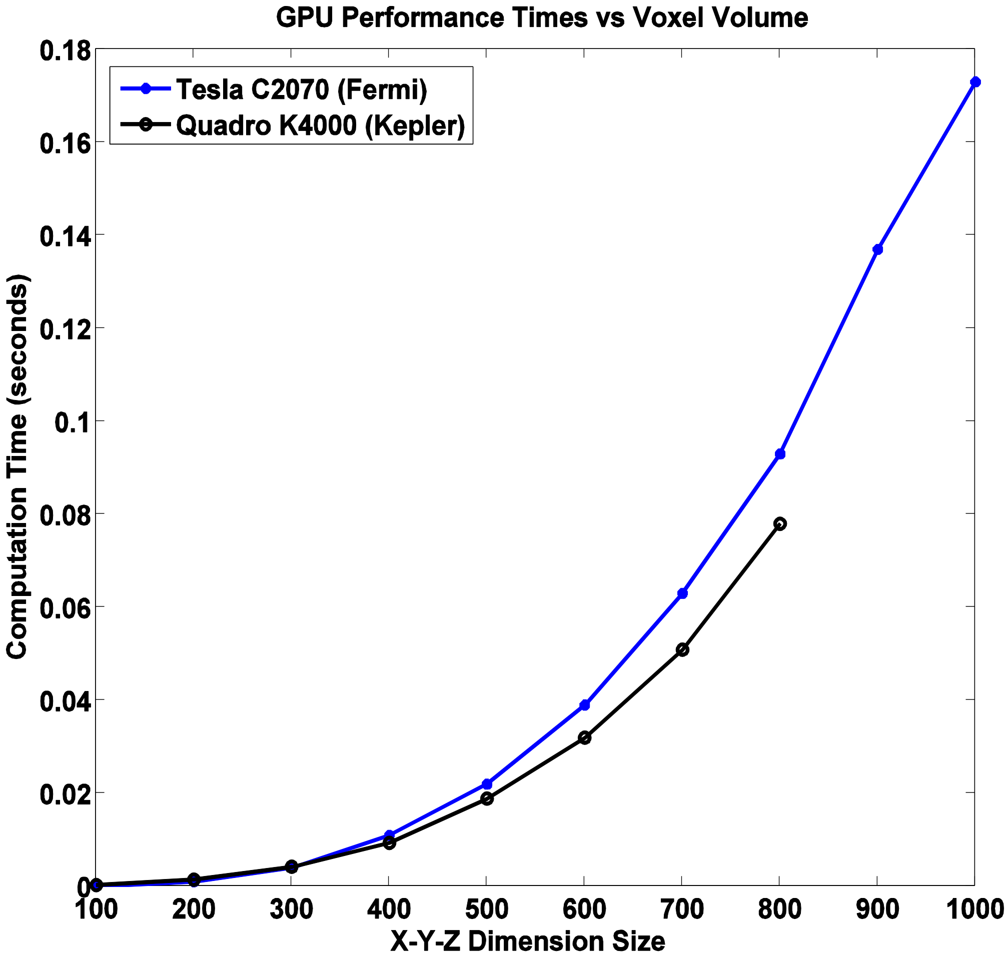
Results(cont.)

- CPU performance, single-threaded compared to two different GPU architecture performance.
- Using a GPU can improve the computational performance by three orders-of-magnitude compared to using a CPU-based implementation.



Results(cont.)

- Comparisons of computational time for different voxel sets ranging from 100^3 to 1000^3 voxels with two different GPU architectures.



Nsight Results

- The Nsight Results were obtained using the Dell Precision T7600 workstation with the GeForce GTX 690 GPU.
- Occupancy
  - High number of active warps (32 threads) per Streaming Processor, which prevents poor instruction issue efficiency
  - 96.60% per Streaming Processor
- Cache Hit Rates
  - For texture cache, 83.84% was achieved.
  - For L2 Cache, 89.1% was achieved.
- Bandwidth
  - For texture cache, it was 1.38 TB/s
  - For L2 Cache, it was 218.63 GB/s

Future Work

- Teravoxel data sets
    - Partition Texture memory into smaller sizes that will fit on the GPU Device memory.
- Multi-GPU-based Implementation
    - OpenMP

Conclusions

- Utilizing a GPU’s specific capabilities has resulted in a high-performance forward-projection model that is three orders-of-magnitude faster than the CPU-based implementation.
- Capability to simulate various materials without the need to actually handle them.
- For Industrial Radiography, voxel sets that are larger than 1000^3 are of much importance to obtain better image resolution.

Acknowledgments

- Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.
- Ismael Perez was funded in part by a grant from the National Institutes of Health: SDSU MARC U*STAR 5T34GM008303-23.