Fast and Parallel Implementation of Adaptive Segmentation of 3D Vessel Images

by

Saritha Dwarakapuram

A Thesis Submitted to the Graduate
Faculty of Rensselaer Polytechnic Institute
in Partial Fulfillment of the
Requirements for the degree of
MASTER OF SCIENCE
Major Subject: ELECTRICAL ENGINEERING

Approved:

_____

Dr. Badrinath Roysam, Thesis Adviser

Rensselaer Polytechnic Institute
Troy, New York

December, 2007

ABSTRACT

Adaptive Surface Segmentation using robust hypothesis testing is a novel technique used for segmenting blood vessel and membrane efficiently. It is a computationally expensive algorithm. The CPU implementation (non- parallelized) of this algorithm is very slow as it involves lots of sequential processing for each voxel of the image to determine if that voxel is a surface voxel or not and also the voxels are processed one at a time. Our primary goal in this project is to achieve a significant performance acceleration compared to CPU implementation. We parallelized this algorithm using a highly parallel vector co-processor called Graphical Processing Unit (GPU). Graphics processing units have evolved into a very attractive hardware platform for general purpose computations due to their extremely high floating-point processing performance, huge memory bandwidth and their comparatively low cost. This algorithm uses the data and task parallelism on the GPU to perform memory-intensive and compute- intensive tasks while the CPU is used to perform I/O and resource management. GPU‘s capability of reducing the processing time and producing a quality output makes it a perfect choice for biological image analysis. NVIDIA Geforce 8800 GTX provides a fast and parallel programming interface called CUDA (Compute Unified Device Architecture) which made the general purpose applications easier to implement than traditional SPMD/thread-based approaches. This specialized programming interface provides a relatively simple path for users familiar with the C programming language to easily write programs for execution, by the device. The results on an NVIDIA G80 and an AMD Opteron 244 are compared. We achieved a speed-up up to a factor 2 compared to the original 64 bit CPU implementation. The results suggest that a GPU co-processor can significantly improve performance on large data processing tasks.