MERL – A MITSUBISHI ELECTRIC RESEARCH LABORATORY
http://www.merl.com

Architectures for Real-Time Volume Rendering

Hanspeter Pfister

TR-97-04 February 1997

Abstract

In this dissertation we present special-purpose volume rendering architectures that are capable of rendering high-resolution (e.g., 1024^3) datasets at 30 frames per second. The architectures, Cube-3 and Cube-4, exploit a high degree of parallelism to provide real-time parallel and perspective viewing from arbitrary directions. They allow for interactive control of rendering and projection parameters and the direct visualization of dynamically changing volume data.

The underlying algorithm uses lookup-tables for parallel and perspective ray-casting with tri-linear interpolation of samples along rays. The algorithm performs surface normal estimation from cached interpolated samples. Shading and classification are part of the pipeline and do not require any pre-processing. The algorithm allows for a pipelined implementation of ray-casting where each voxel is accessed exactly once per projection.

All architectures presented in this dissertation use a high-bandwidth vector memory system. A linear skewing of the address space allows for conflict-free access to scanlines of the volume data. Cube-3 implements ray-parallel ray-casting where data along viewing rays are processed in parallel. However, the architecture requires a global communication network which ultimately limits the performance and the scalability of Cube-3.

Cube-4 operates on data of several neighboring rays. Depending on how the algorithm proceeds we call this approach beam- or slice-parallel. The beam-parallel approach follows a group of rays in viewing direction, whereas the slice-parallel approach operates on data slices parallel to a face of the volume data. We developed an innovative parallel dataflow scheme that requires no global communication except at the pixel level. This combines the benefits of very high memory bandwidth, modularity, and scalability, a result which has not been achieved before. Possible hardware implementations are PCI boards for 256^3, VME boards for 512^3, and multiple boards for 1024^3 datasets, all with 16-bit voxels and 30 projections per second.

We have simulated the algorithm and architectures and have implemented a working prototype of the Cube-4 slice-parallel hardware on the Teramac configurable custom computer at Hewlett-Packard Labs. Our results indicate true real-time performance for high-resolution datasets and linear scalability of performance with the number of processing pipelines.

PhD Thesis, State University of New York at Stony Brook, December 1996