Final Technical Report

DOE ECPI Award: DE-FC02-06ER25776
Title: SciDAC Institute for Ultrascale Visualization
Site at University of Virginia

Principal Investigator: Greg Humphreys
Contact: 151 Engineer's Way, P.O. Box 400740
Charlottesville, VA 22904-4740
Phone: 434-982-2220
Email: humper@cs.virginia.edu

Participants

In the current project period, the project has provided funding support for two PhD students in the University of Virginia Computer Science Department (Daniel Lepage and Blake Sutton). A post-doctoral research associate in the department has also received training and contributed to the project (David Koller). The University of Virginia research team has also collaborated with investigators at Sandia National Laboratories (Kenneth Moreland).

Research: Interactive Remote Rendering for Scientific Visualization

The Institute for Ultrascale Visualization aims to address visualization needs of SciDAC science domains, including research topics in advanced scientific visualization architectures, algorithms, and interfaces for understanding large, complex datasets. During the current project period, the focus of the team at the University of Virginia has been interactive remote rendering for scientific visualization.

With high-performance computing resources enabling increasingly complex simulations, scientists may desire to interactively visualize huge 3D datasets. Traditional large-scale 3D visualization systems are often located very close to the processing clusters, and are linked to them with specialized connections for high-speed rendering. However, this tight coupling of processing and display limits possibilities for remote collaboration, and prohibits scientists from using their desktop workstations for data exploration. In this project, we are developing a client/server system for interactive remote 3D visualization on desktop computers.

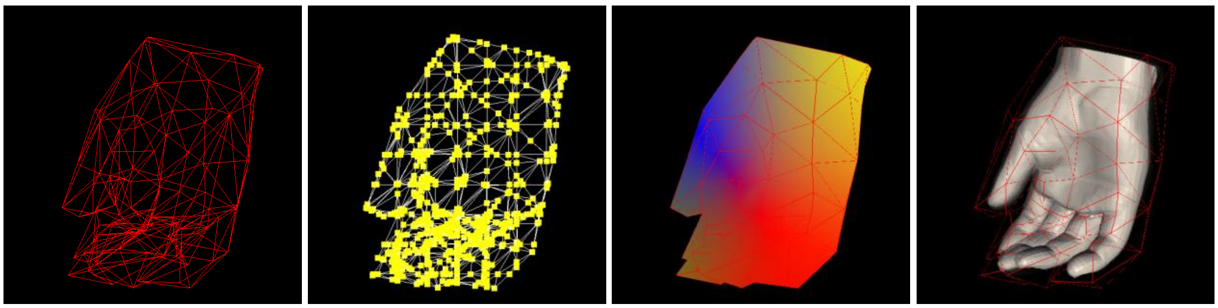
Our remote visualization system utilizes hybrid client-side/server-side 3D rendering and two innovative techniques to achieve low-latency, high-quality 3D display using widely available networks and commodity client workstations:

1) Adaptive, predictive prefetching. Although the full space of possible visualization parameters is high-dimensional, scientists' interactions with simulation and sensor data are typically isolated to changing only a few parameters at a time. We thus are able to roughly model and predict the user's navigation through the visualization space, and accurately prefetch corresponding rendered images from the server. Our prefetcher adapts the volume of its requests to the state of the network traffic and server load.

2) Image-based rendering. Our predictions of the user's navigation path are unlikely to exactly match the future requested visualization parameters. We therefore use image-based rendering on the client to interpolate between the set of images previously acquired from the server; our interpolation method is an extension of Unstructured Lumigraph Rendering (Buehler *et al., SIGGRAPH* 2001). These synthesized views allow low-latency, real-time intermediate image display until the exact visualization is

rendered by the server and arrives over the network. This image-based rendering approach is illustrated in Figure 1, and is described in more detail in Publication [3] below.

We have developed a prototype implementation of our remote visualization algorithms, and have evaluated its performance for scientific visualization applications under varying network conditions. The prototype system works with a variety of server-side scientific visualization rendering frameworks, including the VolPack volume rendering library, the ParaView/VTK visualization application, and general 3D polygonal mesh datasets. With collaborators at Sandia National Laboratories, we are pursuing further incorporation of our remote rendering system with the ParaView/VTK visualization software package; this integration will allow our research to be easily utilized by the larger scientific community.



(a) (b) (c) (d)

Figure 1. Stages of the image-based rendering algorithm for remote visualization: (a) coarse geometric proxy for a higher-resolution mesh; (b) sample points of the blending field; (c) resulting blending field; (d) final synthesized view, with the proxy mesh overlaid.

Publications

(1) David Luebke and Greg Humphreys, "How GPUs Work", **IEEE Computer**, Vol. 40, No. 2, pp. 96-100, February, 2007.

(2) Kwan-Liu Ma, Robert Ross, Jian Huang, Greg Humphreys, Nelson Max, Kenneth Moreland, John D. Owens, and Han-Wei Shen, "Ultrascale Visualization: Research and Education", **Journal of Physics: Conference Series (SciDAC 2007)**, Vol. 78, No. 012088, 2007.

(3) Kenneth Moreland, Daniel Lepage, David Koller, and Greg Humphreys, "Remote Rendering for Ultrascale Data", **Journal of Physics: Conference Series (SciDAC 2008)**, Vol. 125, No. 012096, 2008.