Adventures in Interactive Direct Volume Rendering

Mike Bailey

San Diego Supercomputer Center
University of California San Diego
9500 Gilman Drive
La Jolla, CA 92093-0505
858-534-5142
mjb@sdsc.edu

Abstract

A *volume* is data in the form of a 3D grid of points. Many scientific and engineering sampling and simulation results come that way including medical scans, fluid dynamics, and geophysics Volume visualization comprises a collection of methods for visualizing this type of data. . This paper describes an interactive program, the Volume Explorer (*vx*), which uses high-speed direct volume rendering to gain insights into volume datasets in ways that have not been possible before. The program, its user interface, and several different applications are described.

Keywords: Volume Rendering, Scientific Visualization, Collaborative Visualization, Stereo Vision

Introduction

A volume is data in the form of a 3D grid of points. Each point is referred to as a volume element, or *voxel*. Volume visualization comprises a collection of methods of visualizing these grids of voxels.

Some of the volume visualization methods, such as cutting planes and isosurfaces, work by creating compromises that simplify the data display. For example, a volume cutting-plane displays many data values, but only on one particular slice of the volume. An isosurface is displayed throughout the entire volume, but only shows one particular data value.

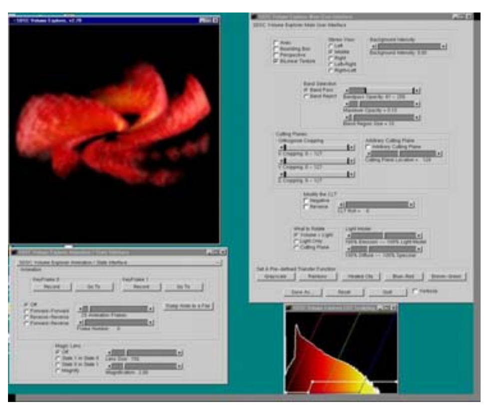
Another method, *direct volume rendering*, shows the entire volume at once – all data values throughout the entire volume. It accomplishes this by mapping the range of data values to a series of color and opacity values (this mapping is known as the transfer function), and then blending the colors and opacities as if a light was being shined through the volume.

The problem with direct volume rendering is that it is very compute-intensive. Traditionally, it has required heavy compute resources to perform, and even heavier resources to perform interactively.

The Volume Explorer Program

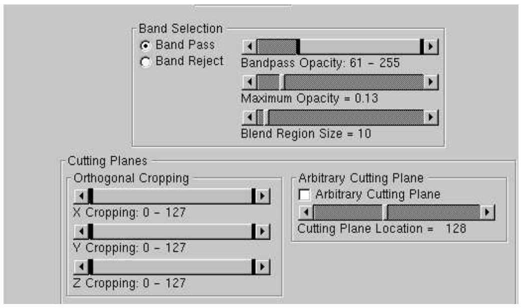
Visualization researchers at the San Diego Supercomputer Center have written a program called *Volume Explorer*, or *vx* for short, that performs interactive direct volume rendering. It accomplishes this by using the TeraReon Volume Pro 1000 volume accelerator board. *vx* uses OpenGL, GLUT (the GL Utility Toolkit), and GLUI (GL User Interface) to perform the graphics and user interaction.

The photo below shows the *vx* screen. The program consists of four windows: the main display, the transfer function sculptor, the main user interface, and the state/animation user interface.



vx screen

The main *vx* user interface takes advantage of *range sliders*, an enhancement we have added to the GLUI library. Unlike standard sliders, the range sliders have three control points: the lower end of the desired range, the upper end, or both ends simultaneously. This has been especially effective for setting the range of visible scalar values and cropping planes.

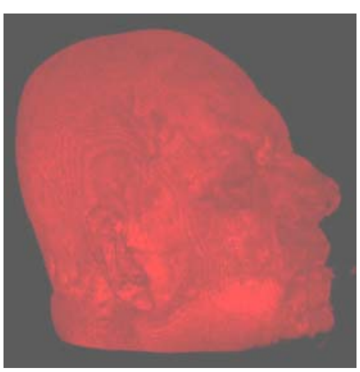


vx animation interface

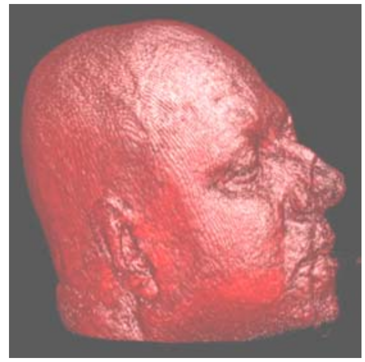
This combination of hardware and software runs on a standard desktop PC. The combination costs less than $8,000, so that it is affordable enough to be used in most any research situation. With *vx*, we have become very big fans of interactive direct volume rendering. This paper shows some of our work with *vx*. All of these images are, of course, static. But, understand that on the screen they can be changed and manipulated at update rates of 30+ frames per second.

Lighting

We have been surprised how much a moveable light source contributes to the understanding of direct volume rendered data. The VolumePro card performs a Phong lighting calculation on each sample prior to compositing it. The results below shows why this is so valuable.



Emmissive lighting only



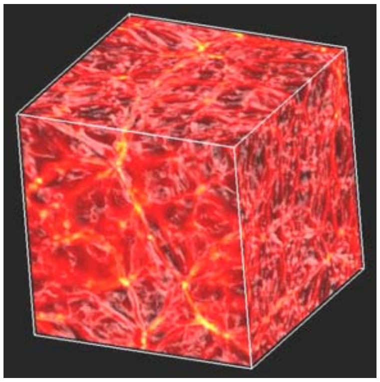
Emmissive, diffuse, and specular

The first figure shows emissive lighting only, that is, just the color of the voxels' transfer function with no lighting model. Second figure shows the same models with the Phong lighting model turned on. Notice how features such as the blood vessels near the top of the head are far more noticeable.

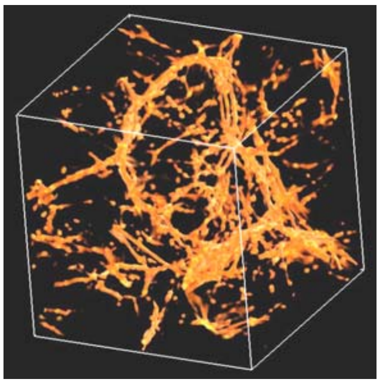
This figure shows a rear head shot from the Visible Human dataset. The lighting makes the folds in the skin visible. We have found the diffuse part of the Phong model to be the most useful in bringing out volume detail. Specular lighting is also available, but, for the most oart, we find it more distracting than helpful.

Applications

The following volume is the result of a simulation from Dr. Michael Norman, a UCSD Astrophysicist, showing the density of the universe after the big bang. *vx* is able to load several volumes simultaneously and flip between them to create a volume animation. When this volume is manipulated to eliminate the low background values, the string behavior of the universe emerges.

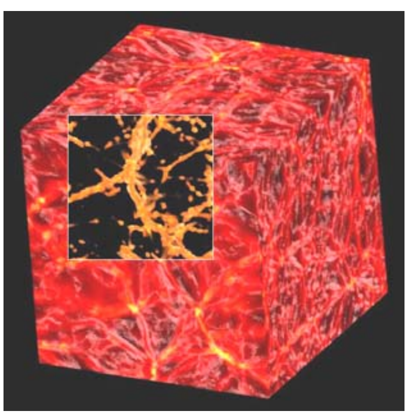


All data values

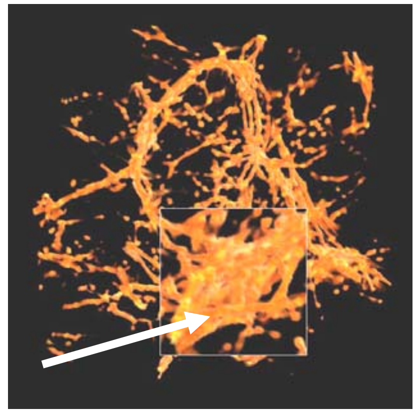


Just the string data values

Because the VolumePro plays so well with OpenGL, we can combine it with other traditional visualization techniques. The first image below shows a moving *magic lens* to see the strings and the full volume at the same time. The second image below shows a magic lens being used as a magnifier.

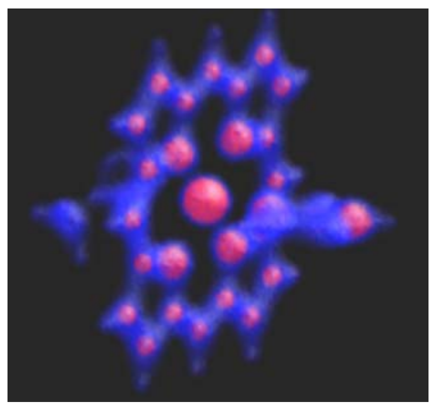


Magic lens to see inside the volume



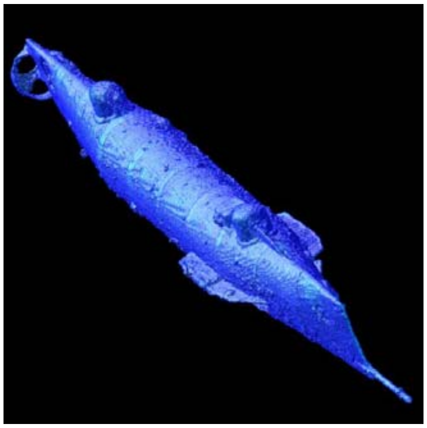
Magic lens as a magnifing glass

This technology is also changing computational chemistry. Dr. Kim Baldridge, a computational chemist from the San Diego Supercomputer Center, is using *vx* to display and understand molecules in their natural state – as probabilistic electron clouds:



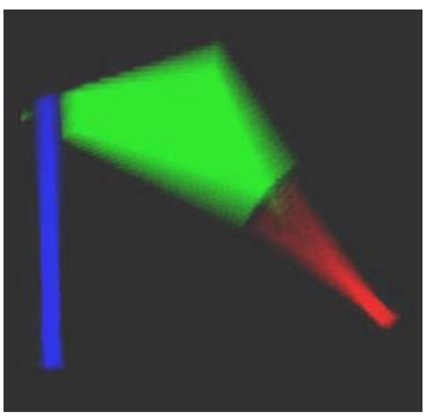
Computational chemistry volume

An interesting outcome of all of this is that we not only use this combination as a routine visualization tool, but we even turn things into volumes that did not necessarily arrive that way. For example, the following image shows the Confederate Submarine Hunley, which was recently raised and scanned into 9,000,000 3D points. To visualize the results of the scan, we bin'ed the points into a volume and displayed it using *vx* as shown below. The ability to do lighting really helped understand the surface features.

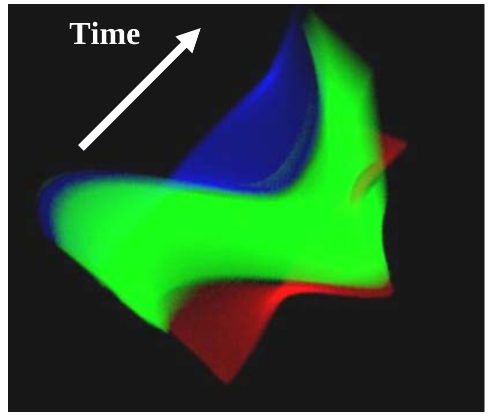


Scattered point volume

Another *vx* application that did no toriginally arrive as a volume is four-bar linkage motion visualization. Four-bar linkages are 2D planar mechanisms and form the cornerstone of the field mechanical kinematics. Often it is useful to understand the time-motion of such mechanisms to design coordinating mechanisms that do not collide anywhere in the motion cycle. The first figure below shows a four-bar mechanism in one time position. We then animated it over time, extruding it one level deeper in 3D for each time step. The result is shown in the second figure below. It is a complete picture of how this device behaves throughout its motion cycle. Other mechanism's behaviors can now be superimposed on this volume. Any volume regions of overlap would represent collisions during the cycle.

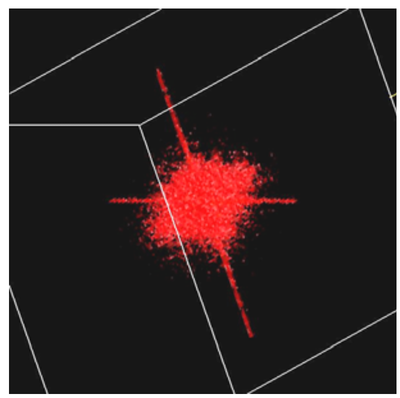


Mechanical four-bar linkage



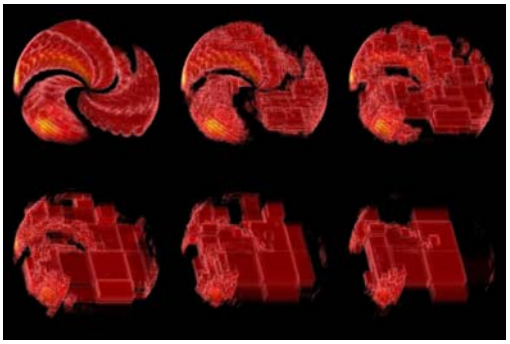
Linkage motion extruded in time

This ability is so flexible that we can create and view volumes to give us inside information on other volumes. For example, the next figure shows a volume that is the Fourier transfoirm volume of wrinkled-skin head volume shown above. Notice that the two in-plane dimensions of this volume (left-right, up-down) have lots of high frequencies, indicating crispness of th edata. The between-plane dimension (in-out) has mostly low frequencies, indicating less crispness.



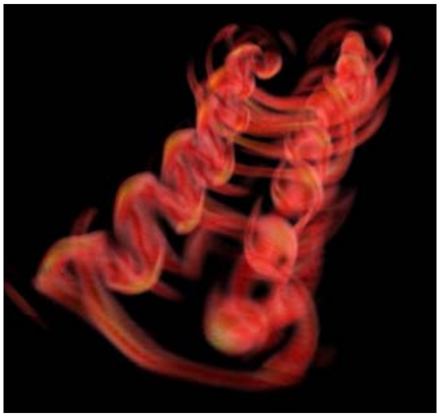
Fourier volume

Wavelet analysis of a volume likewise produces other volumes. The composite image below shows several lavels of wavelet compression of a solar wind volume dataset.

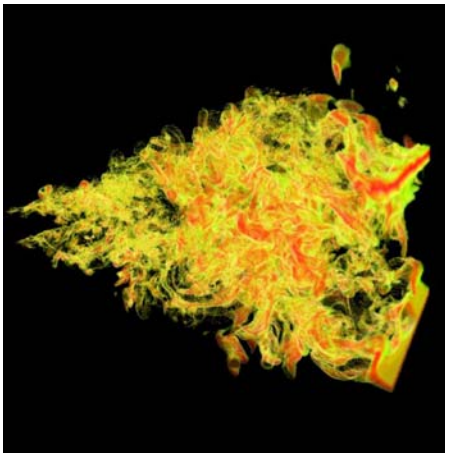


Volume wavelet compression

Computational fluid dynamics is a big producer of visualization volume datasets. The images below show two such examples: The figure is a simulation of vortices, commonly found on the tips of high speed aircraft. The second figure is a turbulent flow. The blended nature of direct volume rendering gives this sort of visualization the "wispy" look that one would expect if we could see the actual flow.



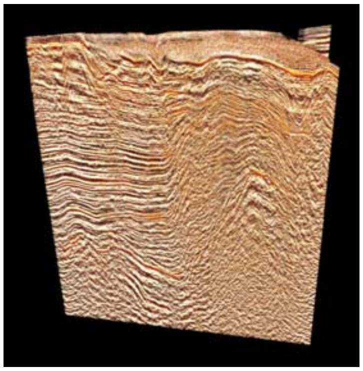
Vortex simulation



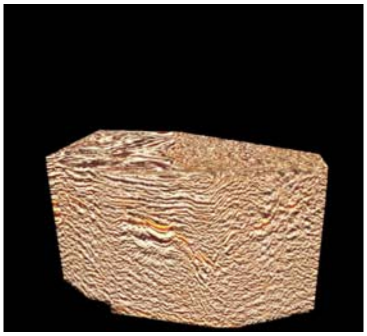
Turbulent flow simulation

The following series shows a geophysical application. This dataset is quite large – over 200 million voxels – but we can interact with it at anywhere from 5 to 2o updates per second. One of the visualization goals of interacting with this dataset was to find the significant rock layers. It can be seen that slicing through the volume helps some, but doesn't give the entire 3D picture. We then decided to isolate just the high-amplitude reflections. This got us the rock layers – but also got us the noise. We then used vx's gradient filter slider to get rid of the high frequency noise, revealing just the rock layer structure.

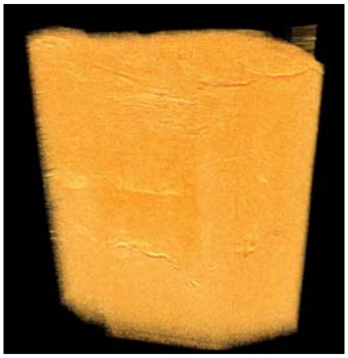
Whole geophysical volume



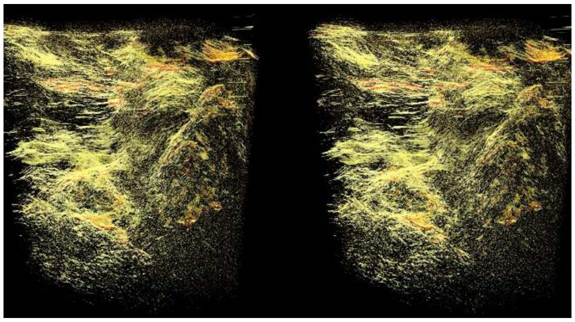
Sliced front-to-back



Sliced top-to-bottom



Just the high-amplitude reflections



Crossed-eye stereopair of the low-gradient high-amplitude reflections

Stereopairs

Because *vx* plays with full OpenGL, it is also able to use stereographics to make the internal structure of the volumes even more understandable. We use OpenGL quad-buffered stereo with LCD shutterglasses as shown below. Besides being very useful for scientific work, we have also used this technique in K-12 education.

A 4th Grader discovers human anatomy

Conclusions

Because of its speed and ease of use, the Volume Explorer program has opened up a new world of datasets that can be better understood. Because it is a desktop, low-cost solution, this sort of interaction can be made available most anywhere. As can be seen from the breadth of examples shown here, *vx* has allowed us to collaborate with a variety of other researchers, and more are popping up all the time. Even data that is not inherently in volume form, can be converted that way to take advantage of this methods.

Web Availability

vx can be downloaded from the Web at: http://dvl.sdsc.edu/vx.

References

Hanspeter Pfister, Arie Kaufman, and Tzi-cker Chiueh, "Cube-3: A Real-Time Architecture for High-Resolution Volume Visualization", *1994 Symposium on Volume Visualization*, October 1994, pp. 75-82.

Jürgen Hesser, Reinhard Männer, Günter Knittel, Wolfgang Straßer, Hanspeter Pfister, and Arie Kaufman, "Three Architectures for Volume Rendering", *Computer Graphics Forum*, Vol. 14, Number 3, August 1995. pp.111-122.

Hanspeter Pfister, Jan Hardenbergh, Jim Knittel, Hugh Lauer, and Larry Seiler, "The VolumePro Real-Time Ray-casting System", *Proceedings of SIGGRAPH 99*, pp. 251-260.

Note to reviewers: the static nature of these photos doesn't do justice to the interactive system being described here. If this paper is accepted, in the conference presentation I will bring videotape showing the interaction and how it gives such good insight into these datasets. It is also possible to bring the full system, if desired, to do live demos.