Interactive Multimodal Volume Visualization for a Distributed Radiation-Treatment Planning Simulator

Ulrich Neumann, Andrei State, Hong Chen, Henry Fuchs,
Tim J. Cullip, Qin Fang, Matt Lavoie, John Rhoades

University of North Carolina
at Chapel Hill

Abstract

We describe the VISTAnet gigabit testbed's visualization system, a parallel, interactive, multimodal renderer implemented on the Pixel-Planes 5 graphics engine. The system displays multiple superimposed data sets characterizing a 3D radiation treatment scenario. Its features are designed to facilitate comprehension of spatial relationships among geometrically complex anatomy and radiation dose structures. The algorithms and visualization options of this mature system are general and should be useful for other applications. The VISTAnet system is currently in clinical evaluation.

1. Introduction

The VISTAnet Project is one of several national gigabit network testbeds coordinated by the Corporation for National Research Initiatives (CNRI) focusing on technology and applications of high-speed networks. Our testbed features a (near-) gigabit/sec communications network connecting three research sites (Fig. 1):

• The UNC Department of Radiation Oncology (UNC-RO), located on the Chapel Hill campus, with a high-end graphics workstation, termed Medical Workstation (MWS).

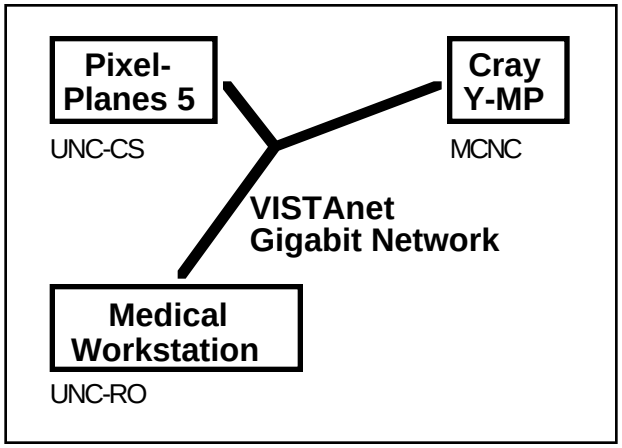


Fig. 1 - VISTAnet testbed sites

• The UNC Department of Computer Science (UNC-CS), also on the Chapel Hill campus, with Pixel-Planes 5, a high-performance graphics engine designed and built at UNC-CS.

• The Microelectronics Center of North Carolina (MCNC), located approximately 18 miles from the UNC campus, with a 4-processor Cray Y-MP supercomputer.

VISTAnet's principal application is interactive Radiation Treatment Planning (RTP), a cancer treatment method which attempts to deliver lethal radiation to cancerous tissue, while keeping the doses received by healthy tissue at non-lethal levels. The strategy is based on intersecting multiple beams onto a predetermined 3D target region, a complex task requiring thorough understanding of shape and sensitivity of the anatomical structures involved. The visualization system allows a physician user to interactively steer a supercomputer simulation of radiation therapy and thus develop and modify radiation treatment plans. Although the visualization system is tailored for radiation treatment planning its methods and features are generally applicable to other domains.

The VISTAnet visualization system uses principles outlined in [Sabe 88], [Dreb 88], [Levo 88], [Levo 89], and [West 89]. The initial experimental renderer was briefly described in [Yoo 92] and [Rose 92]. The version described here reflects the current state of our knowledge on interactive visualization for this demanding supercomputing application.

2.VISTAnet System Operation

The physician user works at the MWS (Fig 2). The MWS software allows the user to load anatomy data sets (we use CT scans) and define or modify radiation treatment beam parameters. These are transmitted over the gigabit network to the Cray, which computes the dose distribution produced within the anatomy by the treatment beam configuration. The dose distribution data is transferred over the high-speed network to Pixel-Planes 5, which renders a combined 3D image of anatomy, treatment beams, and resulting dose; the physician examines the rendering (Fig. 3) and continues to adjust the treatment plan parameters. The current processing rate is several such adjustments per second for anatomy

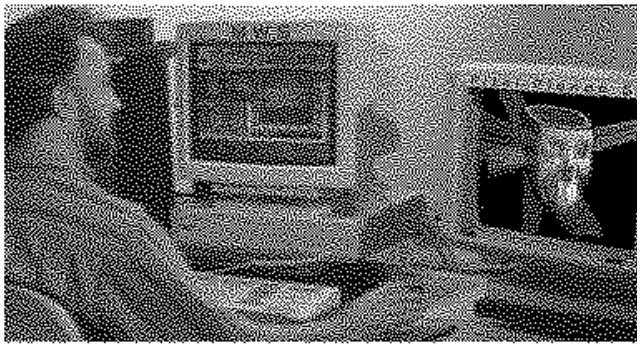


Fig. 2 - VISTAnet user at MWS and Pixel-Planes 5 monitor (Pixel-Planes 5 image simulated)

data sets containing about a million voxels. By comparison, a physician preparing a conventional 2D treatment plan waits minutes to hours to be able to examine the radiation dose distributions resulting from a single tentative scenario. Using VISTAnet's simulation power, UNC-RO researchers have created radiation treatment plans which are in theory superior to those resulting from conventional 2D slice-based methods [Sail 93].

2.1. Visualization Requirements

Interactive RTP places high demands on real-time visualization of the results issued from the interactively steerable supercomputation on the Cray Y-MP. Numerous parameters (position, orientation, shape, and intensity of each of a variable number of treatment beams) are continually adjusted in order to optimize the computed 3D distribution of radiation dose patterns applied to the anatomy. The tentative treatment scenario can change several times per second; the visualization subsystem must hence be able to quickly convey the scenario's characteristics to the user. In other words, it must help the user quickly comprehend spatial relationships among simultaneously displayed, geometrically complex dose and anatomy structures.

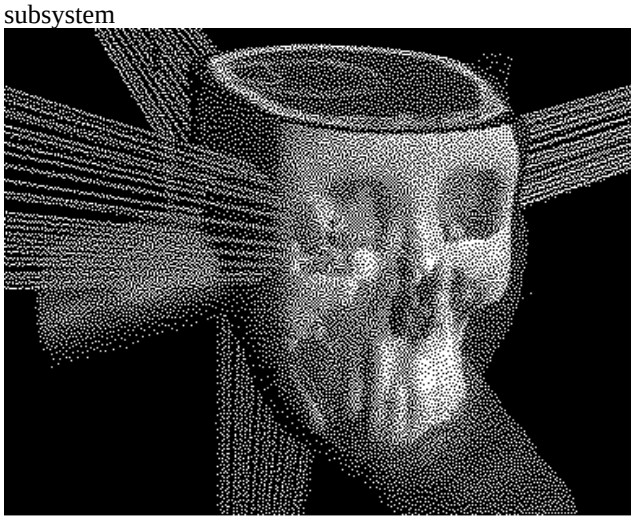


Fig. 3 - Typical VISTAnet display showning treatment plan for a patient with a tumor near right ear. Cut plane shows dose isocurves.

2.2. Display Content

Many visualizations convey *qualitative* information about the spatial distribution of data values. RTP requires *quantitative* dose information to be conveyed by an image. The answers to questions such as

• What is the radiation dose at some specific point in the anatomy?
• Which parts of the anatomy receive a specific radiation dose?

must be obvious from the presented display; hence the graphic content of the display must include both the (static) anatomy data and the dynamically changing dose distribution.

Two additional requirements emerged over the three-year course of the project:

• The capability to label areas of the anatomy data set as belonging to specific organs or to tumor volumes, required in order to ease orientation and spatial comprehension, to avoid damage to vital and/or particularly sensitive healthy organs, and to improve targeting on the cancerous tissue.

Fig. 4 - Visualization showing the pelvis of a prostate cancer patient. Anatomy is represented via a translucent skin isosurface, an opaque bone isosurface (yellow), and presegmented anatomical structures: prostate (red), bladder (green). A cutaway plane shows radiation dose isocurves. The translucent and opaque blue elements are 3D radiation dose isosurfaces.

The capability to display significant information about the radiation beam configuration, thought to ease comprehension of the way treatment beam changes affect the dose distribution.

Typical display images (Figs. 3, 4) are a mixture of volume renderings, surface drawings, and line drawings, and consist of

1) a volume rendering of anatomy merged with
2) a volume rendering of dose isosurfaces, combined with
3) a cutaway plane showing original volume data,
along with
4) radiation dose isocurves and
5) radiation beam outlines presented as wireframes.

Figures 3 and 4 show both the static and the dynamic data sets represented by multiple, user-changeable, superimposed, interpenetrating, translucent or opaque surfaces. For the anatomy, the surfaces are usually set to delineate tissue boundaries such as skin or bone surfaces, or previously identified structures such as internal organs or tumor volumes. For the dose, the isosurfaces show multiple radiation dose threshold levels. The cutting plane with its radiation dose isocurves is often moved quickly by the user to different places in the patient anatomy to gather detailed information about local dose levels. Figure 3 shows a treatment scenario for which shape, position, and orientation of the treatment beams are indicated by wireframe elements. For physician users to optimize a treatment plan it is vital that visualizations, such as those shown in figures 3 and 4, update at interactive rates.

3. Visualization System

The visualization system, developed jointly by UNC-RO and UNC-CS researchers, is implemented on Pixel-Planes 5, a high-performance graphics engine with general-purpose computing nodes based on the Intel i860 microprocessor, and special-purpose rendering nodes based on massively parallel SIMD processor-enhanced memories [Fuch 89]. All nodes are interconnected via the system's internal 5 Gigabit/sec token ring network. Also connected to the token ring are frame buffers, the Network Interface Unit (NIU) providing access to the external VISTAnet Gigabit network, and the Sun-4 host computer. Although Pixel-Planes 5 is used, the algorithms are general and may be implemented efficiently on any multicomputer. The visualization system was designed to operate within VISTAnet or stand-alone.

3.1. The Rendering Pipeline

Interactive rendering rates must be provided so the system is structured to facilitate progressive refinement of the image. This entails computing images at lower than screen resolution while the view parameters are changing and increasing the computed resolution when the parameters remains fixed. The variable resolution adds an additional stage to the rendering pipeline (Fig. 5). There are six components in the pipeline: the host, the master GP, ray casters, compositors, splatters, and the frame buffer. The host provides UNIX services and allows users real-time control through an X-window interface and joysticks. The master GP is an i860 node responsible for system synchronization and load balancing the ray casting nodes. Most of the i860 nodes are allocated as ray casters which compute image pixels of their assigned data subsets and screen regions. A few i860 nodes are used as compositors which combine the ray caster images into a final image. The variable-resolution image produced by the ray casters and compositors is sent to splat processors which interpolate the image over the full frame buffer resolution and write

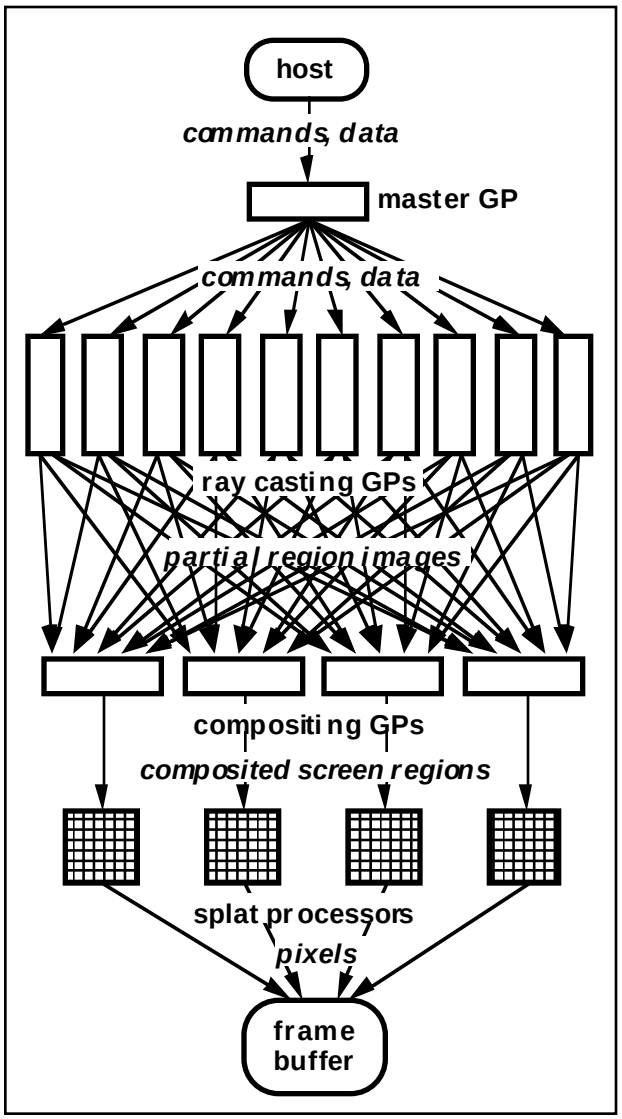


Fig. 5 - Visualization pipeline

the result to the 24-bit color frame buffer.

3.2. Ray Casting GPs

Ray casters compute pixel values in their assigned screen regions for their assigned data sets. The data distributions are static although the dose data set values are dynamically updated. Local memory on the GPs can hold about 1 Mvoxels of anatomy and dose data. If the combined anatomy and dose data does not fit in a ray casting GP's memory, slab subsets of the volumes are assigned to groups of ray casters (Fig. 6) such that each GP in a group is assigned the same slab, and separate slabs are assigned to separate groups. The (dynamic) dose data set (8 bits per voxel) is transmitted from the CRAY to the ray casting GPs through the NIU. Dose values may arrive asynchronously and do not interact with the rendering operations. Ray caster GPs send their computed pixel values to the compositors for merging into a final image.

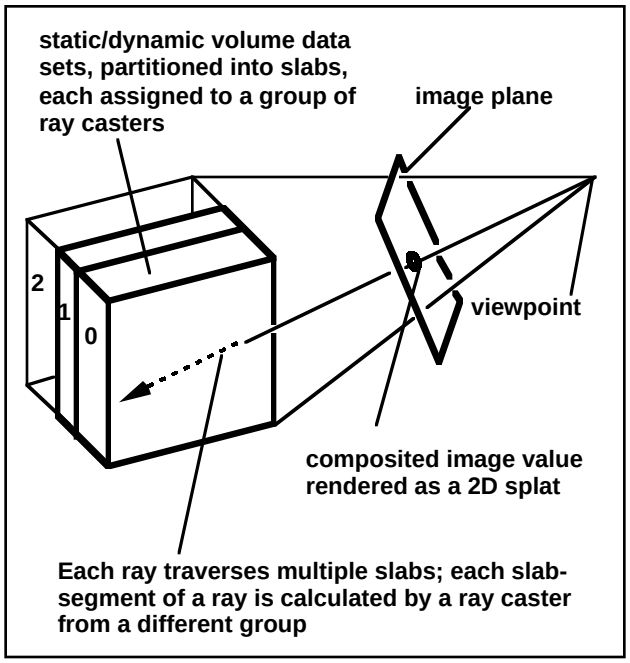


Fig. 6 - Object space data partition into slabs

3.3. Compositing GPs

The partial images from different ray casting GPs are combined by the compositor GPs to get the final pixel values in the image plane. Typically four nodes are allocated to the compositing task, each responsible for a quarter-height stripe of the final image. The resolution of the computed image varies due to the use of progressive refinement, so the array of computed pixel values is sent to the splat processors to generate the final image at the frame buffer's fixed 640×512 resolution.

3.4. Splat Processors

In the Pixel-Planes 5 implementation, SIMD rendering nodes are used as splat processors due to their availability and efficiency at this task. Composited ray values from the compositors are convolved with a 2D filter kernel to resample the image at frame buffer resolution. Several filter kernels are implemented with selection under user control, among them box, bilinear, biquadratic, piecewise quadratic, and bicubic filters. Figure 7 is a split screen with the upper half showing only the pixel values from the compositors and the lower half showing the interpolated image. The resampled values are sent to the frame buffer for display.

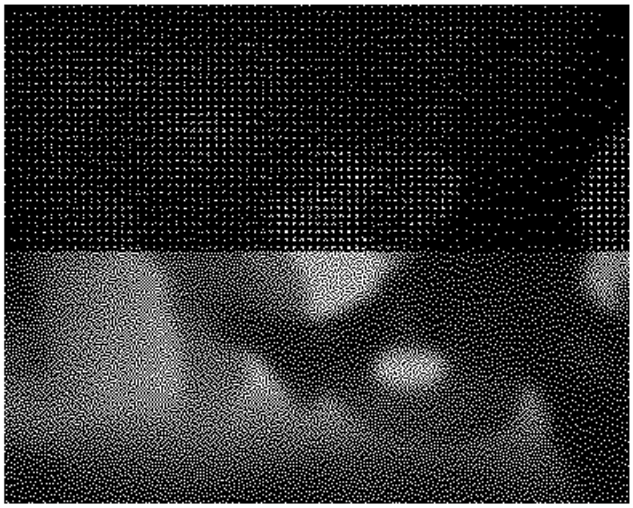


Fig. 7 - Resampling low resolution in splat processors

4. Rendering Modes

The visualization system has evolved over a period of time driven by the needs of the experimental RTP application. As a result many visualization and rendering options have been tested, found useful, and incorporated into the system.

4.1. Mutimodal Data

Visualizing merged anatomy and radiation dose volumes is a core requirement of the RTP application. The dynamic nature of the dose volume complicates its display since precomputed gradients or shading values are not available. The dose is displayed as one, two, or three isosurfaces of variable levels and the gradients for surface shading are computed on-the-fly. Proper compositing of the dose and anatomy isosurfaces must be ensured, especially when multiple surfaces lie between voxels (Fig. 8). Each surface's distance from the viewpoint is computed and sorted to establish the correct order for compositing.

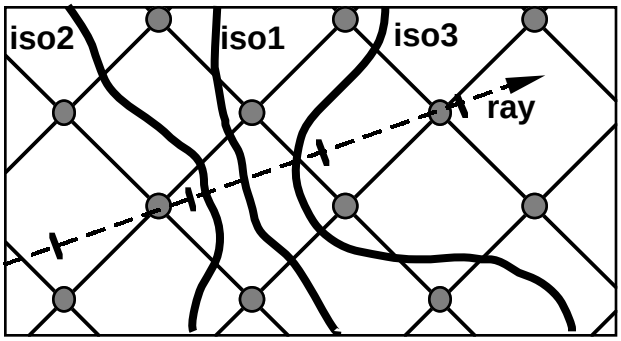


Fig. 8 - Isosurfaces between samples along the ray must be sorted properly. Surfaces detected in arbitrary ofder must be composited in depth-order. For example, surfaces detected in <1,2,3>-order must be composited in <2,1,3>-order.

4.2. Adaptive-Progressive Refinement

It takes several seconds to render a high-resolution VISTAnet image on Pixel-Planes 5; we therefore have to judiciously balance the allocation of computing resources between image quality and image generation speed. Without sufficient quality the user cannot see necessary details in the image; without sufficient speed, the user cannot fully comprehend spatial relationships due to lack of depth cues (such as kinetic depth effect), which diminish rapidly with decreasing frame rate.

We adopted a resolution/speed tradeoff, sacrificing some image space resolution in favor of higher frame rates, in particular in image areas which are fairly uniform in appearance (adaptive refinement). Furthermore, the system displays coarse frames at interactive rates and takes advantage of spare cycles to increase overall resolution as the user pauses to examine a gradually refining image (progressive refinement). Figure 9 contrasts the resolution of a moving image (9b) and the full resolution of the final image (9a).

Adaptive-progressive refinement is made more flexible by giving the user full interactive control over the resolution/speed trade-off parameters. Kinetic depth effect is also provided for high-quality
images by appending to the progressive refinement sequence a series of seven highest-resolution frames displaying the visualized structures in animated oscillatory rotation (rocking cineloop). Thus, if the user pauses long enough, he or she will notice gradually increasing image resolution, followed by increasingly smooth left-right rocking of the displayed high-resolution structures. Depth cues are also provided by directional lighting with diffuse and specular reflections from the anatomical data.

(a) (b)

Fig. 9 - a) Fully refined image (1 ray per image pixel)
b) Adaptively sampled (1 ray every 4-16 pixels)

4.3. Geometric Primitives

To aid in the visualization of the radiation beams and dose, lines and polygons were added as renderer primitives. Antialiased lines are used to represent to beam outlines outside of the volume or to annotate the display. Lines are Z-buffered against isosurfaces and each other. They are added by the splat processors (Fig. 10) after the image is resampled to frame buffer resolution. Polygons are used to create cut planes and add reference geometry to the scene. As axial cut-planes, they may be textured with anatomical data and isocontours of dose level (Fig. 3). This feature has proven to be very useful for quantifying the dose levels on a feature of interest. The operator may set up to seven adjustable contour levels for display on the cut plane and move the plane through the feature while watching the contours. Polygons are rendered by the ray casting GPs since they must be composited properly with the volume data. Since there are typically few polygons, their rendering cost is minimized by testing a bounding-box to ascertain which rays might hit which polygons. Ray setup involves computing the intersection distance to the polygons to eliminate intersection tests for every ray step.

4.4. Load Balance & Adaptive Sampling

Ray casting nodes are assigned screen regions dynamically to adjust the load balance between them. Two assignment methods are implemented and user selectable. In the simplest approach the master GP assigns sequential scan lines (Fig. 11a) to the ray casting nodes on a first-come-first-serve (FCFS) basis. This provides good balance but precludes adaptive sampling since screen areas, not lines, produce the greatest adaptive-sampling efficiency.

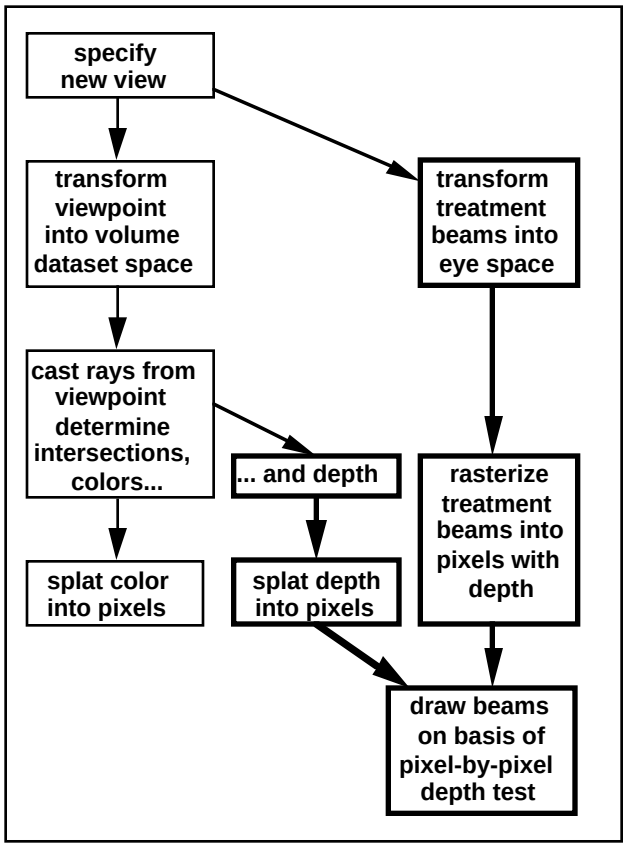
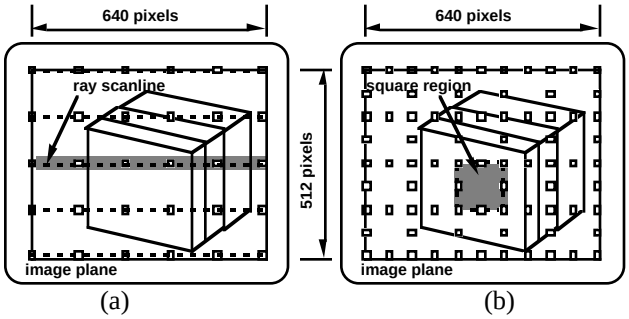


Fig. 10 - Algorithm for adding line primatives

In the second load balancing approach the master GP distributes 65×65-pixel screen regions (Fig 11b) on a FCFS basis to the ray casting nodes. Regions are distributed in order of descending cost where cost is the time taken to render the region in the previous frame. Assignment of regions on the basis of descending cost provides approximately 10-20 % increase in frame rate under most conditions. The region size includes two edges of replicated rays to support adaptive sampling without seams.



(a) (b)

Fig. 11 - Image space partitioning by a)lines and b)square regions

Both approaches produce good load balance with the latter approach giving better performance (Fig 12) for full-screen images (due to the adaptive sampling) and more consistent frame rates over varied image sizes.

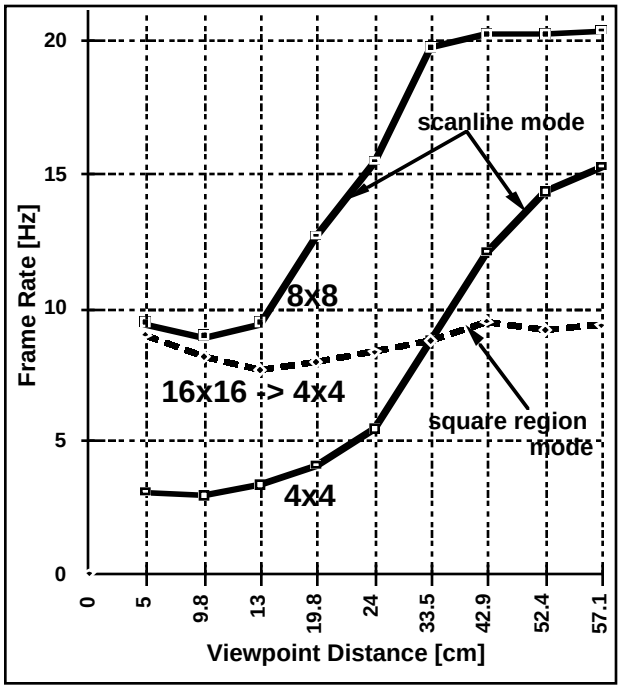


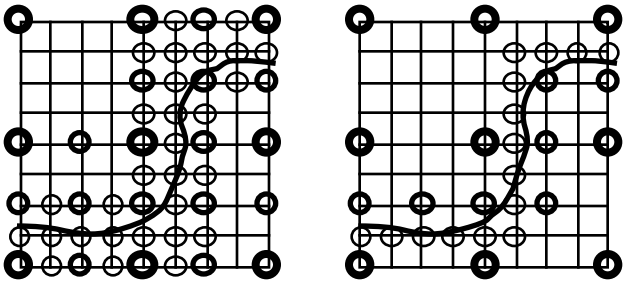
Fig. 12 - Frame rate comparison between line and square region partitions for varying object size. (Full screen images are produced by 5 cm viewpoint distance, 60 cm distance produces 1/16 screen coverage.) Line partiions cast rays every 8x8 or 4x4 pixels. Square regions are adaptively sampled initially at one ray per 16x16 pixels and refined up to one ray per 4x4 pixels.

The adaptive sampling approach is a modified form of recursive square subdivision. It produces fewer rays and provides similar results to that used in [Levo 90]. Figure 13 illustrates its operation. The conventional approach (Fig 13a) fully subdivides a square area by computing five new samples when any pair of four corner values exhibit variance. Our partial-subdivision approach (Fig 13b) computes new samples only between varying point pairs with the center sample taken if any points vary. The example shows the new approach requires fewer samples than the full subdivision method. A triangular subdivision method [Shu 91] has similar economy, but is less suited to small rectangular regions.

4.5. Rendering Modes

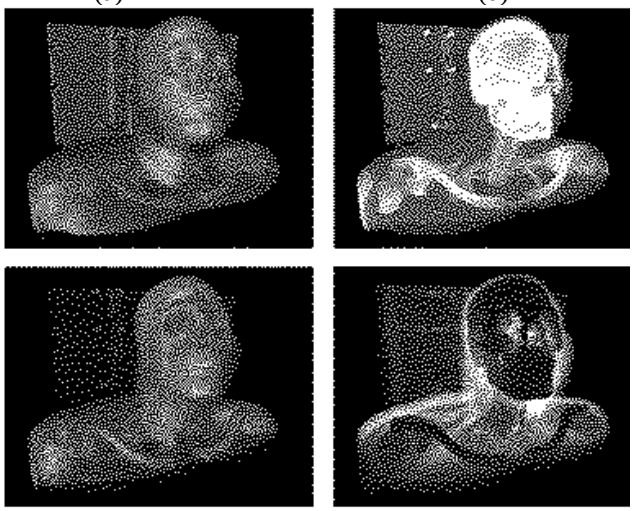
Although VISTAnet uses only isosurface rendering data visualization modes vary with the application and subjective bias of the user. To support other applications and users the rendering system incorporates many of the more common modes that have proven useful over time including isosurface rendering, direct rendering with and without shading, and maximum intensity projection

(MIP). Figure 14a,b,c,d illustrates the visualizations obtained by these modes from the same data.



(a) (b)

Fig. 13 - Conventional adaptive subdivison (a) causes 51 samples to be taken while partial subdivision (b) requires only 31 samples. Samples taken at successive levels of subdivision are represented by progressively finer circles.



(a) (d)

(b) (c)

Fig. 14 - Examples of rendering modes

Test modes are also included to illustrate the behavior of an algorithm . Figure 15 shows the adaptive refinement sampling pattern against a background of screen region assignments. Differences in the sequnces of isosurfaces encountered by neighboring rays triggers adaptive sampling along contours and isosurface intersection curves. Figure 16 shows a bar graph of the ray casting GP workloads normalized to the highest load. These test modes have proven very useful for developing and testing new functions.

4.6. Optimizations

To obtain satisfactory frame rates, the 8-bit anatomy voxels values are stored with 13-bit precomputed normals. A shading table is computed at the start of every frame that encodes the Lambertian coefficient for the given light direction(s) as a function of the surface normal. Voxel shading is efficiently performed by lookup into this table. Threshold bits at each voxel speed up ray processing by flagging whether an 8-voxel cell has an isovalue within it. A total of 40 bits (including flags for anatomical structures) are used for a static anatomical voxel. Rays are terminated when an opacity threshold is reached.

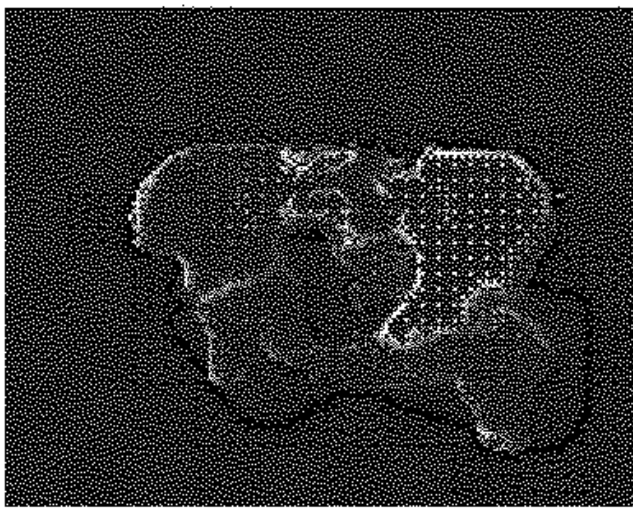


Fig. 15 - Adaptive sampling pattern against the square regin partitions shown in red and blue. The view is the same as in Fig. 16.

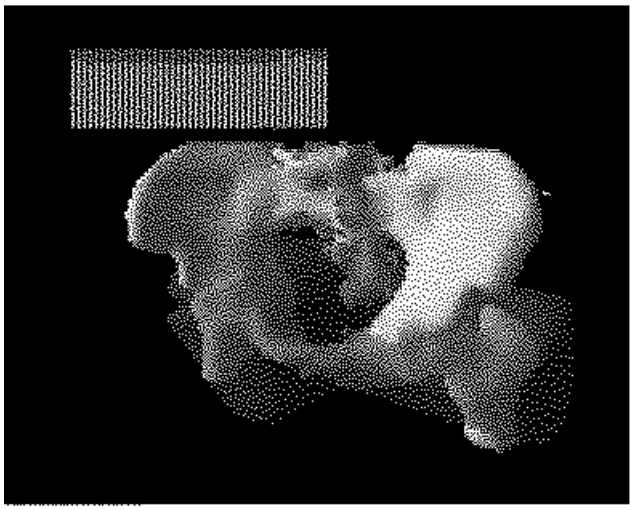


Fig. 16 - Samples from Fig. 15 interpolated by splat processors to fill screen. Load balance graphs show work distribution among the rays casting GPs.

5. Conclusions and Future Work

It has been extremely difficult to achieve the current visualization frame rates despite the availability of a high performance graphics engine. To do so, we had to specifically target and optimize much of the visualization code to Pixel-Planes 5. We are not satisfied with the current performance/resolution tradeoff since it weakens the kinetic depth cues. We have made both high frame

rates and kinetic depth effect available in our system; however, lack of sufficient computing power has so far prevented us from offering high frame rates *together* with high-resolution images, in other words: interactive kinetic depth. We therefore need to significantly increase our image generation speed.

We have been very pleased with the flexibility and power of the visualization system for other applications. Its many capabilities have been useful in unexpected ways. Figure 17 shows the dynamic data loading ability used to acquire ultrasound data in real-time for resampling and rendering as a volume. Figure 18 shows the polygon rendering ability used to "insert" the isosurface rendered fetus in the mothers womb. Lines are used in both cases to delimit areas. Stereo rendering has been incorporated to support alternate-frame stereo displays using LCD shutter glasses. Perspective projections are produced from any viewpoint, including inside the volume, to support head-mounted displays.

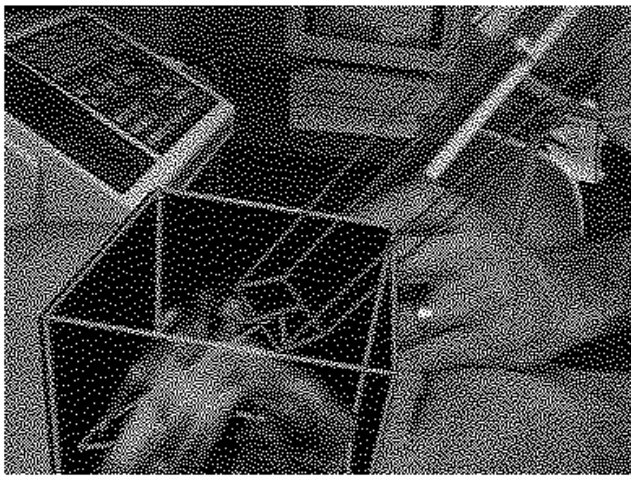


Fig. 17 - Head-mounted display view of real time 2D ultrasound slice acquisition, volumetric reconstruction, and volume rendering. Tracked ultrasound probe and volume boun daries are represented by line segments.

On the next-generation UNC graphics engine (PixelFlow [Moln 92]) a VISTAnet-like visualization could benefit from at least a factor of 5 speedup, and considerably more by taking advantage of that system's image-based texturing capability. In fact, such capabilities are available today on high-end graphics workstations such as the Silicon Graphics Reality Engine, if we accept limitations such as lower-speed radiation dose update and fixed lighting.

The 3D treatment plan definition/optimization task in VISTAnet is akin to interactively steering the radiation dose supercomputation based on real-time visualization of the treatment scenario. As high-bandwidth communications systems become more ubiquitous, we expect visualization systems that are embedded within tightly coupled supercomputing networking environments to become more commonplace. We hope that our experimental work in this area will contribute towards the usefulness and effectiveness of such systems.

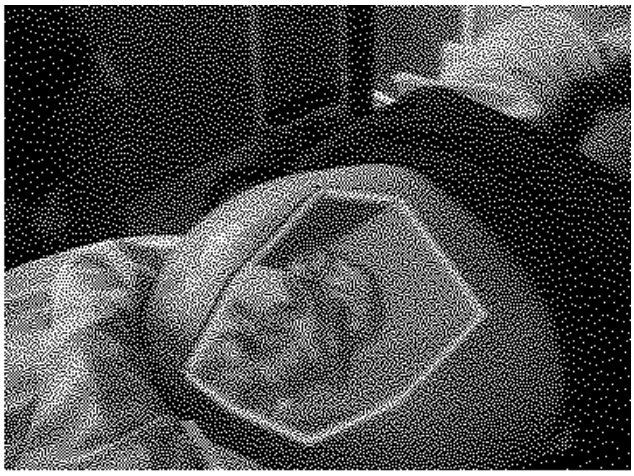


Fig. 18 - Volume rendering of fetus registered with video imagery acquired by a tracked moving camera.

6. Acknowledgments

We thank Jim Symon, MWS software manager, Suresh Balu and David T. Chen for their coding efforts, Stephen M. Pizer (UNC-CS Senior Investigator), and Scott Sailer, M. D. (UNC-RO physician user) for their significant contributions and constructive criticism. This work was supported by NSF and ARPA under Cooperative Agreement NCR-8919038 with CNRI ("VISTAnet: A Very High Bandwidth Prototype Network for Interactive 3D Imaging"), by BellSouth, and by GTE.

7. References

[Dreb 88] Drebin, Robert A. et al., "Volume Rendering," *Computer Graphics*, 22(4) (Proceedings of SIGGRAPH '88), pp. 65-74.

[Fuch 89] Fuchs, Henry et al., "Pixel-Planes 5: A Heterogeneous Multiprocessor Graphics System Using Processor-Enhanced Memories," *Computer Graphics*, 23(3) (Proceedings of SIGGRAPH '89), pp. 79-88.

[Levo 90] Levoy, Marc, "Volume Rendering by Adaptive Refinement," *The Visual Computer* 1990, Vol 6, pp 2-7.

[Levo 89] Levoy, Marc, "Design for a Real-Time High-Quality Volume Rendering Workstation," *Proceedings of the Chapel Hill Workshop on Volume Visualization*, 1989, pp. 85-92.

[Levo 88] Levoy, Marc, "Display of Surfaces from Volume Data," *IEEE Computer Graphics and*

Applications, May 1988, pp. 29-37.

[Moln 92] Molnar, Steven et al., "PixelFlow: High-Speed Rendering Using Image Composition," *Computer Graphics*, 26(2) (Proceedings of SIGGRAPH '92), pp. 231-240.

[Rose 92] Rosenman, Julian et al., "VISTAnet: Interactive Real-Time Calculation and Display of 3-Dimensional Radiation Dose: An Application of Gigabit Networking," *Int. J. Radiation Oncology Biol. Phys.*, Vol. 25, pp. 123-129.

[Sabe 88] Sabella, Paolo, "A Rendering Algorithm for Visualizing 3D Scalar Fields," *Computer Graphics*, 22(4) (Proceedings of SIGGRAPH '88), pp. 51-58.

[Sail 93] Sailer, S. L. et al., "The Tetrad and Hexad: Maximum Beam Separation as a starting point for Noncoplanar 3-D Treatment Planning," *Int. J. Radiation Oncology Biol. Phys.*, Vol. 27:138.

[Shu 91] Shu, Renben and Alan Liu, "A Fast Ray Casting Algorithm Using Adaptive Isotriangular Subdivision," *Visualization '91 Proceedings*, October 1991

[West 89] Westover, Lee, "Interactive Volume Rendering," *Proceedings of the Chapel Hill Workshop on olume Visualization*, 1989, pp. 9-16.