Eurographics Italian Chapter Conference (2008)
V. Scarano, R. De Chiara, and U. Erra (Editors)

Towards advanced volumetric display of the human musculoskeletal system

Marco Agus, Andrea Giachetti, Enrico Gobbetti, José Antonio Iglesias Guitián, Fabio Marton

Visual Computing Group, CRS4, Pula, Italy – http://www.crs4.it/vic/

Abstract

We report on our research results on effective volume visualization techniques for medical and anatomical data. Our volume rendering approach employs GPU accelerated out-of-core direct rendering algorithms to fully support high resolution, 16 bits, raw medical datasets as well as segmentation. Images can be presented on a special light field display based on projection technology. Human anatomical data appear to moving viewers floating in the light field display space and can be interactively manipulated.

Categories and Subject Descriptors (according to ACM CCS): B.4.2 [Input/Output and Data Communications]: Input/Output Devices Image Display I.3.3 [Computer Graphics]: Picture/Image Generation I.3.7 [Computer Graphics]: Three-dimensional graphics and realism

1. Introduction

Nowadays, medical acquisition devices, and especially medical scanners, are able to produce a large amount of information in the form of high-resolution volumes, temporal sequences, or functional images. This information is more-and-more difficult to analyze and visualize. In other words, we measure much more than we understand. In this context, higher-level information, such as anatomical and functional models, is increasingly required to support diagnosis and treatment. That's why the scientific community is putting a lot of efforts to manage the proliferation of medical data in order to have a comprehensive view of human anatomy. For example, the 3D Anatomical Human network (EU Project FIXME) aims to develop realistic functional three-dimensional models for the human musculoskeletal system, with special emphasis on the lower limb. In order to develop realistic and accurate functional models, 3D volume visualization systems are needed to visualize medical data, both in raw and segmented form. In particular, volume renderers are important to compare segmented reconstructions with raw data coming from medical scanners. While high quality volume rendering techniques exist, managing huge datasets is still considered a challenge. Furthermore, it is difficult to use 2D screens and human interfaces to understand and manipulate 3D representations of the imaged volumes. To overcome these limitations, high resolution multi-scopic visualization displays, i.e., displays that are engineered to produce a light field that is perceived by naked eye observers as coming from an actual three dimensional scene, look promising. In this paper, we report on our research results on effective volume visualization techniques for medical and anatomical data, using advanced 3D displays and new interaction paradigms [ea07]. In previous work we have shown that light field displays can be used to display medical data and perform diagnostic tasks [BFB*05, BFG*06, ABGP07, BAGZ07, AGI*08]. These results are extended by integrating recently introduced [GMG] out-of-core accelerated direct rendering algorithms to fully support high resolution, 16 bits, raw medical datasets as well as segmentation. The resulting new visualization system can handle, at interactive speed, very large datasets, thus removing the dataset size limitations of the previous solution.

2. Light field display overview

Several approaches have been proposed to support a real 3D visualization without classical single point of view or single user limitations of traditional stereo displays. A review on the subject can be found on [Dod05]. One way to visualize volumetric data is to display 2D images on rapidly moving surfaces [Fav05, JMY*07, CNH*07]. Another technology for volumetric visualization is the evolution of multi-view displays showing multiple 2D images in multiple zones in space (i.e. the method used for auto-stereoscopic displays) [RR05, MP04]. The light field display considered here uses the distributed image generation approach of projector-based multi-view technology, but removes some of its optical

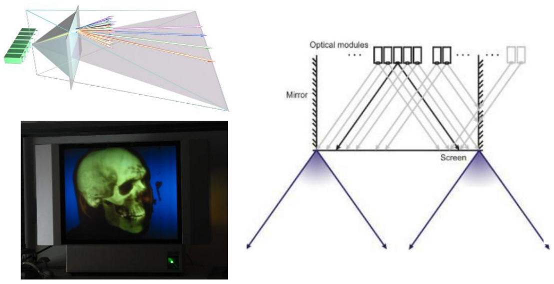

Figure 1: ***Light field display.*** *The light field display is composed by an array of projectors densely arranged in an horizontal array behind the screen, all of them projecting their specific image onto the holographic screen to build up a light field.*

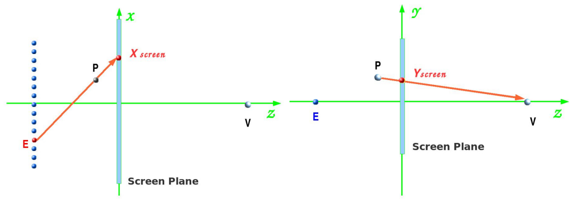

Figure 2: ***Light field geometry****: Left: horizontally, the screen is sharply transmissive and maintains separation between views. Right: vertically, the screen scatters widely so the projected image can be viewed from essentially any height.*

limitations, since it offers a fully continuous transition among views, currently only along the horizontal direction, thanks to the light shaping capabilities of a holographically recorded screen. For more information on the technology, we refer the reader to [BFA*05]. The system is composed by an array of projectors densely arranged in an horizontal array behind the screen, all of them projecting their specific image onto the holographic screen to build up a light field (see figure 1). The latter is the key element in this design, as it is the optical element enabling selective directional transmission of light beams. In the horizontal parallax design, the projectors are arranged in a horizontal linear array and the angular light distribution profile induced by the screen is strongly anisotropic. Horizontally, the surface is sharply transmissive, to maintain a sub-degree separation between views. Vertically, the screen scatters widely so the projected image can be viewed from essentially any height. Since humans perceive depth using horizontally-offset eyes and move their viewpoint more easily from side to side than up and down, the horizontal parallax only approach is adequate for most applications and provides significant speed-up. The display hardware here employed is manufactured by Holografika and is capable of visualizing 7.4M beams/frame by composing optical module images generated by 96 fast 320x240 LCD displays fed by FPGA input processing units that decode an input DVI stream. The on screen 2D pixel size of the display is $s_0 = 1.25mm$, and the angular accuracy is 0.8°.

The first step to build a rendering pipeline on the display consists in determining where 3D points should be drawn on a given projector to produce a perspective correct image for the virtual viewer. In fact, the linear perspective is not sufficient to determine how a 3D graphics application should project points, because it ignores the transformation performed by the holographic screen. Since the screen is selective only in the horizontal direction, but scatters widely in the vertical one, the displayed light field's dimensionality is reduced, and the application must decide how to deal with the missing degree of freedom. In order to provide a full perspective effect, the vertical viewing angle must thus be known, which amounts at fixing the viewer's height and distance from screen.

The projection of a point **P**, on the screen **S** for a given emitter E can be computed for the x coordinate by intersecting the ray originating from the emitter E with the screen plane at $z = 0$ and for the y coordinate by intersecting it with the ray arriving to the virtual viewer eye positioned at coordinates **V** (see figure 2). In rasterization applications, **S** is then remapped to normalized projected coordinates by transforming to the image rectangle and associating a depth (for Z-buffering) based on the distance to the screen.

3. Volume rendering

Given the display technology, it is necessary to design and implement efficient rendering techniques to represent volumes as simplified light fields. We should be able, in other words, to create light beams propagating in specific directions from the screen pixels, exactly corresponding to those that would be emitted from physical objects at fixed spatial locations. After a brief review of the techniques more strictly related to our work, we describe our methods to drive the spatial light field display in order to get scalable interactive volume ray casting visualization of large human dataset. Many sophisticated techniques for real-time volume rendering have been proposed in the past, taking advantage of CPU acceleration techniques, GPU acceleration using texture mapping, or special purpose hardware. We refer the reader to the recent book of Engel et al. for a recent survey [EHK*06]. Furthermore, the out-of-core organization of massive volumetric data into a volume octree is a classic problem. Previous approaches proposed a multiresolution sampling of octree tile blocks according to view-dependent criteria [BNS01] Other methods separately render blocks using volumetric raycasting on the GPU and devise propagation methods to sort cells into layers for front-to-back rendering, therefore reducing frame-buffer demands [HQK05, KWAH06]. The separate rendering of blocks does not easily allow an implementation of optical models to render refracting global illumination effects. Our method, is based on a full-volume GPU ray-casting approach [KW03, RGW*03], with a fragment shader that performs the entire volume traversal in a single pass [SSKE05]. Such an approach, made possible by modern programmable GPUs, is more general, but, until very recently, has been limited to moderate size volumes that fit entirely into texture memory. We exploit GPU vertex shaders to render proxy geometry that activates a fragment shader performing the actual ray-casting. Our factorization of the ray computation operations is however different to [SSKE05], since our rendering pipeline cannot rely on the interpolation performed by the rasterizer to pass down combined 3D and projected data from the vertex shader. The result is an interactive real-time scalable volume visualization system able to provide multiple freely moving naked-eye viewers the illusion of presence of virtual volumetric objects floating at fixed physical locations in the display workspace. Details can be found in [AGI*08, GMG]. The method follows the two-pass approach typical of contemporary multi-projector displays [BMY05]. In the first one, for each projector view, the scene is rendered off-screen to a frame buffer object using the previously described light field geometry model. In the second one, small non-linear view and color distortions are corrected by streaming the first pass texture through a fragment shader that warps the geometry and modifies colors thanks to per-pixel look up tables stored as precomputed textures. In order to allow the volume rendering of very large datasets, we designed an adaptive technique based on the decomposition of a volumetric data set into small cubical bricks, which are then organized into an octree structure maintained out-of-core [GMG]. The octree contains the original data at the leaves, and a filtered representation of children at the inner nodes. Each node also stores the range of values, as well as, optionally, precomputed gradients. In order to efficiently support runtime local operations based on neighboring voxels, i.e. linear interpolation or gradient computations, we replicate neighboring samples inside the bricks. At runtime, a working set of bricks is then generated and incrementally maintained on CPU and GPU memory by asynchronously fetching data from the out-of-core octree. The working set is created by an adaptive loader on the basis of the selected view and transfer function. At each frame, a compact indexing structure, which spatially organizes the current working set into an octree hierarchy, is encoded in a small texture. In this structure, neighbor pointers directly link each leaf node of the octree via its six faces to the corresponding adjacent node of that face, or to the smallest node enclosing all adjacent nodes if there are multiple ones. We create such a structure on-the-fly at each frame directly from the view-dependent octree, and encode it into a 3D texture that acts as a spatial index. The spatial index structure is exploited by an efficient stackless GPU ray-caster, which computes the volume rendering integral by enumerating non-empty bricks in front-to-back order, adapting sampling density to brick resolution, and stopping as soon as the accumulated opacity exceeds a certain threshold, updating both the frame-buffer and and depth-buffer. Our method relies on the ability to rapidly traverse a octree structure and is based on the stackless ray traversal method for kd-trees [HBS98] recently extended to GPUs for surface rendering [PGSS07]. Our method exploits the regular structure of octrees to reduce costly texture memory accesses by computing bounding boxes on-the-fly. In addition, our algorithm takes advantage of occlusion queries to avoid loading occluded data.

The system has been developed in order to fulfill all requirements involved in the analysis of high quality and high resolution anatomical data. Volume data are represented as 16 bit scalars, and 32 bit gradients (8 bits for each direction component, and 8 bits for the norm), and the rendering pipeline is at fully floating point precision. The gradients are precomputed by employing high quality 5x5x5 Sobel filtering. The system is also able to manage and render segmented datasets. In that case, the precomputation of levels of detail is modified to chose for each value the most popular label instead of the average. Moreover, at run-time, trilinear filtering is substituted with nearest neighbors. Jittering of ray start position is employed to reduce aliasing artifacts.

4. Implementation and results

We have implemented a prototype hardware and software system based on the technologies discussed in this paper. The software system consists of a framework written in C++ and OpenGL, a set of Cg shaders that implement the basic ray-casting engine, and a number of shader functions that implement different compositing techniques. The octree is stored in an out-of-core structure, based on Berkeley DB, and data is losslessly compressed with the LZO compression library.

The DVI channel feeding the light field display works at 1280x1024 at 75Hz. Each 1280x1024 frame collects 16 320x240 projector images, plus a color-encoded header in the top rows that encodes the ids of the projectors that have to be updated. A full 3D frame is created by sequentially generating all the projector images into the frame buffer. The graphics application runs on an Athlon64 3300+ PC with a NVIDIA8800GTX graphics board working in twin-view mode. One DVI output is used for control on a 2D monitor, while the second one feeds the 3D display.

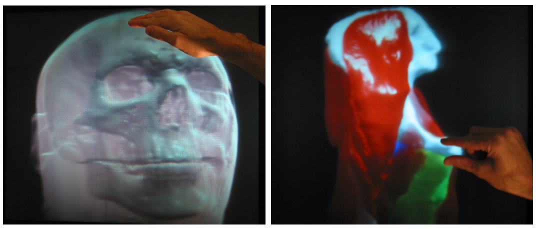

Figure 3: ***Visible Human visualization.*** *Direct capture with a digital camera of an interactive inspection of anatomical human datasets visualized on the light field display with Direct Volume Rendering plus Non Photo-realistic transparency effects. Left: Visible Male head dataset with 512x512x512 resolution. Right: 3D Anatomical Human leg dataset with 220x272x1270 resolution.*

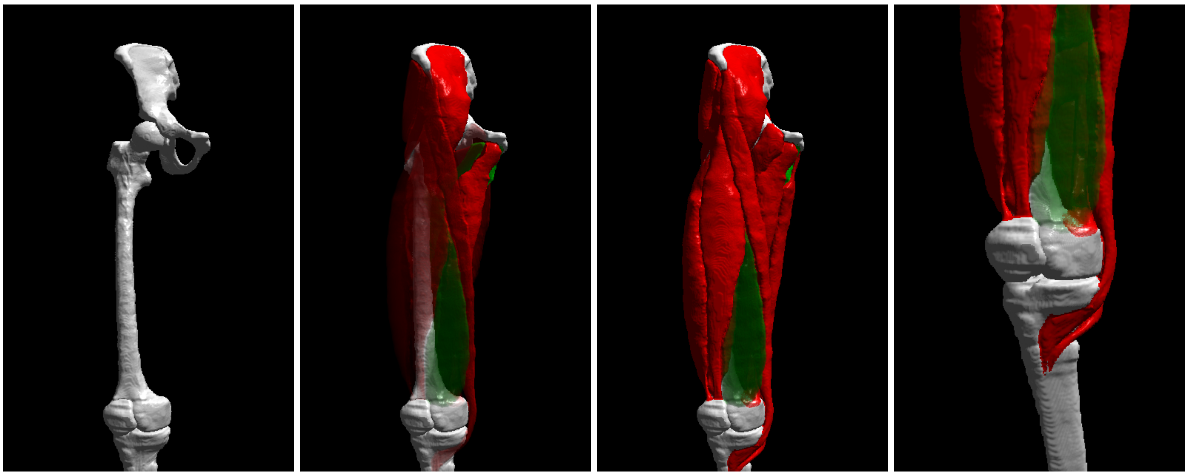

Figure 4: ***Interactive human muscolo/skeletal volume visualization.*** *Snapshots taken from direct interaction with the 3D Anatomical Human leg dataset. Different transfer functions are employed to highlight various biological tissues.*

It is obviously impossible to fully convey the impression provided by our holographic environment on paper or video. As a simple illustration of our system's current status and capabilities, we show some pictures recorded live using a digital camera (see figure 3). Objects appear to moving viewers floating in the display space and can be manipulated by translating, rotating, and scaling them with a six degree of freedom tracker, as well as by modifying the transfer function. We considered different anatomical human datasets: The Visible Human Male head [†], and the 3D Anatomical Human leg high resolution segmented dataset[‡], with resolution of 220x272x1270 at 16 bits. Our volume ray caster implements a number of composition strategies, that include Direct Volume Rendering with a Phong illumination model, boundary enhancement and view-dependent transparency [BG07]. Various transfer functions can be employed in order to highlight different biological tissues (see figure 4). The specular highlights correctly follow the recording camera's viewpoint, contributing to volume data readability. The perceived image is fully continuous, differently from other contemporary multi-view technologies, which force users into approximately fixed positions, because of the abrupt view-image changes that appear when crossing discrete viewing zones [MP04]. It is important to note that even when a single "static" 3D view is displayed, users can exploit accommodation, stereo and motion parallax to gain understanding of complex shapes. Some of these cues can be also obtained with traditional systems, but only by incorporating interactive manipulation in the rendering system. In that case, users will have to move the object or the viewpoint to provide the visual system with enough depth information. The task is not simple and immediate, and depth information is easily lost when the user stops interacting.

† Source: The National Library of Medicine, USA

‡ Source: 3D Anatomical Human Project, MiraLab, Geneve

5. Conclusions and Future Work

We reviewed our current work in volume visualization systems for human anatomical data. We employ scalable multi-resolution volume ray casting approaches and drive light field display system. A first conclusion than can be drawn from our research is that high quality interactive and scalable volumetric rendering of huge datasets on light field displays is currently achievable even when using single GPU desktop solution for the rendering task. Exploring this domain through the design and implementation of highly interactive techniques that leverage the unique features of an interactive multi-viewer 3D environment is a challenging area for future work.

Acknowledgments. This work is partially supported by the Italian Ministry of Research under the CYBERSAR project and by the EU Marie Curie Program under the 3DANATOMICALHUMAN project (MRTN-CT-2006-035763).

References

[ABGP07] AGUS M., BETTIO F., GOBBETTI E., PINTORE G.: Medical visualization with new generation spatial 3d displays. In *Eurographics Italian Chapter Conference* (Conference held in Trento, Italy, February 2007), Eurographics Association.

[AGI*08] AGUS M., GOBBETTI E., IGLESIAS GUITIÁN J., MARTON F., PINTORE G.: Gpu accelerated direct volume rendering on an interactive light field display. *Computer Graphics Forum 27*, 3 (2008). Proc. Eurographics 2008, to appear.

[BAGZ07] BRELSTAFF G., AGUS M., GOBBETTI E., ZANETTI G.: Pseudo-holographic device elicits rapid depth cues despite random-dot surface masking. In *Perception, ECVP 2007 Abstract Supplement* (Conference held in Arezzo, Italy, 2007), vol. 36, p. 202.

[BFA*05] BALOGH T., FORGÁCS T., AGOCS T., BALET O., BOUVIER E., BETTIO F., GOBBETTI E., ZANETTI G.: A scalable hardware and software system for the holographic display of interactive graphics applications. In *Eurographics Short Papers Proceedings* (2005), pp. 109–112.

[BFB*05] BALOGH T., FORGÁCS T., BALET O., BOUVIER E., BETTIO F., GOBBETTI E., ZANETTI G.: A scalable holographic display for interactive graphics applications. In *Proc. IEEE VR 2005 Workshop on Emerging Display Technologies* (Conference Held in Bonn, Germany, March 13 2005, 2005). CD ROM Proceedings.

[BFG*06] BETTIO F., FREXIA F., GIACHETTI A., GOBBETTI E., PINTORE G., ZANETTI G., BALOGH T., FORGÁCS T., AGOCS T., BOUVIER E.: A holographic collaborative medical visualization system. In *Medicine Meets Virtual Reality 2006*, Westwood J. D., (Ed.). IOS, Amsterdam, The Netherlands, January 2006.

[BG07] BRUCKNER S., GRÖLLER M. E.: Style transfer functions for illustrative volume rendering. *Computer Graphics Forum 26*, 3 (Sept. 2007), 715–724.

[BMY05] BROWN M., MAJUMDER A., YANG R.: Camera-based calibration techniques for seamless multiprojector displays. *IEEE Trans. Vis. Comput. Graph 11*, 2 (2005), 193–206.

[BNS01] BOADA I., NAVAZO I., SCOPIGNO R.: Multiresolution volume visualization with a texture-based octree. *The Visual Computer 17*, 3 (2001), 185–197.

[CNH*07] COSSAIRT O., NAPOLI J., HILL S., DORVAL R., FAVALORA G.: Occlusion-capable multiview volumetric three-dimensional display. *Applied Optics 46*, 8 (Mar. 2007), 1244–1250.

[Dod05] DODGSON N. A.: Autostereoscopic 3D Display. *Computer 38*, 8 (2005), 31–36.

[ea07] ET AL G. B.-R.: Virtual reality 3d echocardiography in the assessment of tricuspid valve function after surgical closure of ventricular septal defect. *Cardiovascular Ultrasound 5*, 8 (2007).

[EHK*06] ENGEL K., HADWIGER M., KNISS J., REZK-SALAMA C., WEISKOPF D.: *Real-time Volume Graphics*. AK-Peters, 2006.

[Fav05] FAVALORA G. E.: Volumetric 3d displays and application infrastructure. *Computer 38*, 8 (2005), 37–44.

[GMG] GOBBETTI E., MARTON F., GUITIÁN J. A. I.: A single-pass gpu ray casting framework for interactive out-of-core rendering of massive volumetric datasets. Submitted for publication.

[HBS98] HAVRAN V., BITTNER J., SÁRA J.: Ray tracing with rope trees. In *14th Spring Conference on Computer Graphics* (1998), Kalos L. S., (Ed.), pp. 130–140.

[HQK05] HONG W., QIU F., KAUFMAN A.: Gpu-based object-order ray-casting for large datasets. In *Eurographics / IEEE VGTC Workshop on Volume Graphics* (June 2005), pp. 177–186.

[JMY*07] JONES A., MCDOWALL I., YAMADA H., BOLAS M. T., DEBEVEC P. E.: Rendering for an interactive 360 degree light field display. *ACM Trans. Graph 26*, 3 (2007), 40.

[KW03] KRUEGER J., WESTERMANN R.: Acceleration techniques for GPU-based volume rendering. In *Proc. Visualization* (2003), pp. 287–292.

[KWAH06] KAEHLER R., WISE J., ABEL T., HEGE H.-C.: Gpu-assisted raycasting for cosmological adaptive mesh refinement simulations. In *Eurographics / IEEE VGTC Workshop on Volume Graphics* (July 2006), pp. 103–110.

[MP04] MATUSIK W., PFISTER H.: 3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 814–824.

[PGSS07] POPOV S., GÜNTHER J., SEIDEL H.-P., SLUSALLEK P.: Stackless kd-tree traversal for high performance gpu ray tracing. *Computer Graphics Forum 26*, 3 (Sept. 2007), 415–424.

[RGW*03] ROETTGER S., GUTHE S., WEISKOPF D., ERTL T., STRASSER W.: Smart hardware-accelerated volume rendering. In *Proc. VISSYM* (2003), pp. 231–238.

[RR05] RELKE I., RIEMANN B.: Three-dimensional multiview large projection system. In *Proc. SPIE* (2005), vol. 5664.

[SSKE05] STEGMAIER S., STRENGERT M., KLEIN T., ERTL T.: A simple and flexible volume rendering framework for graphics-hardware-based raycasting. In *Eurographics/IEEE VGTC Workshop on Volume Graphics* (2005), pp. 187–195.