

# Multiresolution Visualization of Massive Models on a Large Spatial 3D Display

Fabio Bettio, Enrico Gobbetti, Giovanni Pintore, and Fabio Marton

CRS4 Visual Computing Group, Italy  
{Fabio.Bettio|Enrico.Gobbetti|Gianni.Pintore|Fabio.Marton}@crs4.it

---

## Abstract

*We report on a cluster parallel multiresolution rendering system driving a spatial 3D display able to give multiple freely moving naked-eye viewers the illusion of seeing virtual objects floating at fixed physical locations situated in a human-scale working volume. The efficiency of this approach is demonstrated by an application supporting interactive manipulation of colored highly tessellated models on a large (1.6x0.9 meters) 50Mpixel display that allows for a room-size working space.*

Categories and Subject Descriptors (according to ACM CCS): B.4.2 [Input/Output and Data Communications]: Input/Output Devices Image Display

---

## 1. Introduction

The accurate reproduction of three-dimensional light fields requires the generation of a large number of light beams of appropriate origin, direction, intensity, and color. Recent advances in 3D display design have demonstrated that high resolution display technology able to reproduce natural light fields is practically achievable [BFA\*05]. Even though general considerations on the specifics of human vision – and the requirements of targeted applications – drastically reduce the amount of data that needs to be encoded in a reconstructed light field, three-dimensional rendering still remains a complex and computationally intensive task, which has limited until very recently the application domain of spatial 3D displays to presentation of static images, prerecorded movies, or small scale graphics models.

The goal of this paper is to illustrate, with a practical example, that interactive, high quality, 3D multi-user, naked-eye display technology is practically achievable and that state-of-the-art coarse grained multiresolution techniques are able to fully harness current COTS components for providing enough computing power to drive the display even when interactively manipulating massive geometric models. To this end, we report on a parallel multiresolution rendering system driving a spatial 3D display able to give multiple viewers the illusion of seeing virtual objects floating at fixed physical locations situated in a human-scale working volume (see figure 1). Each viewer sees the scene from their point of view and enjoys full, continuous, horizontal parallax without specialized viewing devices. The display design is based



**Figure 1: Massive model visualization on a 50Mpixel spatial 3D display.** The display has a screen dimension of 1.6x0.9 meters, and provides continuous horizontal parallax. Interactive rates in the manipulation of massive geometric models are guaranteed by a cluster-parallel spatial 3D display aware coarse grained multiresolution technique.

on a specially arranged array of optical modules and a holographic screen. The optical modules project light beams of different colors and intensity onto the holographic screen, which then makes the necessary optical transformation to compose these beams into a continuous 3D view. Each point of the holographic screen emits, in a controlled manner, light beams of different colors and intensity in various directions. With proper software control, the light beams coming from the various pixels can be made to propagate in multiple directions, as if they were emitted from physical objects at assigned spatial locations. The display is driven by a cluster parallel multiresolution renderer able to dynamically adapt

model resolution by taking into account the particular spatial accuracy characteristics of the display. The method is a parallel spatial 3D display-aware version of our Adaptive TetraPuzzles technique [CGG\*04]. The efficiency of this approach is demonstrated by an application supporting interactive manipulation of massive colored highly tessellated models on a large (1.6x0.9 meters) 50Mpixel display that allows for a room-size working space.

The rest of this paper is organized as follows. Related work is briefly summarized in section 2. A general overview of the display concept is presented in section 3, while section 4 discusses the design of our 3D display-aware multiresolution renderer. Preliminary results obtained with the current prototype are discussed in section 5.

## 2. Related work

Developing a scalable spatial 3D display system targeting interactive manipulation of large models is a big engineering effort, which requires the combination of state-of-the-art results in a number of technological areas. In the following, we just provide a brief overview of the approaches most closely related to ours.

**Interactive 3D display technology.** A number of approaches have been proposed to support naked-eye stereoscopic visualization. For a recent review on the subject of display technology we refer the reader to [Dod05]. Broadly, 3D display technology might be classified in the following categories: autostereoscopic displays (e.g., [EWO\*95, WEH\*98] and [RS00, PPK00, Dod06]), multi-view displays (e.g., [vBC97, DML\*00, MP04] and [WHJ\*00, KPC\*05, MKL05]), volumetric displays (e.g., [MMMR00, FDHN01, RS00]), and pure holographic displays (e.g., [SHLS\*95, SCC\*00, HMG03, CKLL05]). The display described here uses the distributed image generation approach of projector-based multi-view technology, but removes some of their intrinsic optical limitations, as it offers a fully continuous blend among views. Typical multi-view displays show multiple 2D images in multiple zones in space. They support multiple simultaneous viewers, but at the cost of restricting them to be within a limited viewing angle. Multi-view displays are often based on an optical mask or a lenticular lens array. The Cambridge multi-view display is a classic design in this area [DML\*00]. Optical masks introduce significant light loss when there are more than two views; moreover, the barrier structure becomes visible as the number of views increases. On the other hand, lenticular displays magnify the pixel matrix (of the projecting devices) creating dark zones between viewing slots. Berkel and Clarke [vBC97] demonstrated an improved seven-view display using a LCD panel and a lenticular sheet. Their design results in a 3D display that partially solves the dark-zone problem and offers reasonable resolution in horizontal and vertical dimensions. More recently, Mitsubishi [MP04] demonstrated a prototype

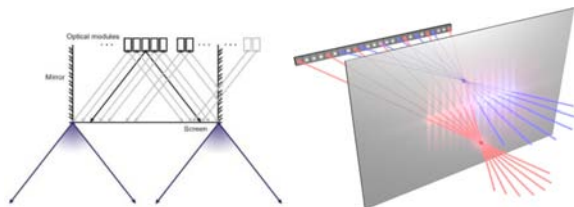
based on this technology and assembled with sixteen 1024x768 projectors and a lenticular screen. A number of manufacturers (Philips [vPF96], Sharp [WHJ\*00], Opticality [RR05], Samsung, Stereographics, Zeiss) produce monitors based on variations of this technology. See also [KPC\*05, MKL05]. Lenticular state-of-the-art displays typically use 8–10 images, i.e., directions, at the expense of resolution. A 3D stereo effect is obtained when the left eye and the right eye see different but matching information. The small number of views produces, however, cross-talks and discontinuities upon viewer's motion [Dod96]. Instead, the solution described here exploits the light shaping capabilities of a holographic screen, and presents a continuous image to many viewers within a large workspace angle, due to the high number of smoothly blended view-dependent pixels that contribute to a single image.

**Large model visualization and multi-view displays.** In recent years, the large demand for entertainment and games has resulted in major investments in commodity graphics chip technology, leading to state-of-the-art programmable graphics units (GPUs) of greater complexity and computational density than current CPUs. In order to fully exploit the capabilities of current graphics hardware, it is necessary to select and send batches of geometric primitives to be rendered with just a few CPU instructions. Following this approach, various GPU-oriented multiresolution structures have been recently proposed. Such structures are based on the idea of moving the granularity of the representation from triangles to triangle patches [CGG\*04, YSGM04, CGG\*05, BGB\*05]. In this work, we adapt one of these methods [CGG\*04] to take into account the 3D display characteristics when producing variable accuracy approximations, and implement it in a cluster-parallel environment rather than a standalone PC solution. As is the case with multi-screen or very high resolution displays, we use a distributed image generation system implemented on a cluster, with a front-end client PC multicasting graphics commands to server PCs. Given the characteristics of the display, we have chosen a sort-last parallel rendering approach. Many other systems obviously exist in this area (see, e.g., cluster-parallel rendering in Chromium [HHN\*02]). However, our system is tailored for a spatial 3D display, in which backends have to render the whole scene using different view parameters. On a smaller scale, the sort-last multiview approach has also been taken for other spatial 3D displays (see, e.g., [CNC\*05]) but not applied to interactive massive model visualization.

## 3. Spatial 3D display technology overview

Our display is based on projection technology and uses a specially arranged array projectors and a holographic screen. It is based on patents held by Holografika ([www.holografika.com](http://www.holografika.com)), who has developed the display hardware and low-level software system used in this work. We summarize here the main concepts behind it. More information is presented elsewhere [BFA\*05]. The projectors are

used to generate an array of pixels of controlled intensity and color onto the holographic screen. Each point of the holographic screen then transmits different colored light beams in different directions in front of the screen. The display system concept makes it possible to produce high pixel-count 3D images by optimally adapting the optical arrangement to the capabilities of the technology and the components applied.



**Figure 2: Schematic diagram.** Left: Each projector emits light beams toward a subset of the points of the holographic screen. Side mirrors increase the available light beams count. Right: A large number of light beams can create a spatial point (voxel).

Our 50Mpixel display uses a specially arranged array of 64 XGA commodity projectors and a holographic screen with a diagonal of 1.8m. The projectors are densely arranged behind the holographic screen, and all of them project their specific image onto the holographic screen to build up the 3D image (see figure 2). By positioning mirrors at the sides of the display, it is possible to reflect the light beams that would otherwise be lost back onto the screen, thus creating virtual projectors that increase the display's field of view. Each projector emits light beams toward a subset of the points of the holographic screen. At the same time, each screen point is hit by more light beams coming from different projectors.

The holographic screen is the key element in this design, as it is the optical element enabling selective directional transmission of light beams. It is a holographically recorded, randomized surface relief structure that enables high transmission efficiency and controlled angular distribution profile. These fully randomized (non periodic) structures are non-wavelength dependent and eliminate moiré, without chromatic effects. The precise surface relief structures provide controlled angular light divergence. The horizontal light diffusion characteristic of the screen is the critical parameter influencing the angular resolution of the system, which is very precisely set in accordance with the system geometry. In the horizontal-parallax only design, the cut of this light distribution is a long rectangle, where the vertical side of the rectangle is the vertical field of view, while the horizontal side corresponds to the neighboring emitting directions. The angular light distribution profile introduced by the holographic screen, with a wide plateau and steep Gaussian slopes precisely overlapped in a narrow region, results in a highly selective, low scatter hat-shaped diffuse characteristic. The result is a homogeneous light distribution and continuous 3D view with no visible crosstalk within the field of depth determined

by the angular resolution. Unlike Fresnel, lenticular and integral screens, where resolution and stray light issues severely limits the operational range of proper optical quality, this approach does not introduce optical roadblocks and it allows large FOV angles. The display's holographic screen provides a horizontal angular diffusion of  $0.8^\circ$ , while the vertical diffusion is  $60^\circ$ . This means that the incident light beam's horizontal divergence is  $0.8^\circ$ , equal by design to the angle under which light beams are arriving from neighboring projectors.

#### 4. Parallel adaptive rendering on the spatial 3D display

With proper software control, the light beams leaving the various pixels can be made to propagate in specific directions, as if they were emitted from physical objects at fixed spatial locations. Reconstructing the light field of a rendered scene amounts to precomputing the projection parameters associated to each of the projectors and to using them for generating multiple perspective views for the same image. Given the high pixel count of the display, and the high triangle count of the target models, appropriate techniques must be employed in order to meet timing constraints in interactive applications. In particular, it is of primary importance to parallelize image generation as well as to dynamically adapt rendering complexity by employing a multiresolution technique, which takes into account the characteristics of the display.

The situation is similar to current parallel view-dependent level-of-detail approaches, with some important differences. First of all, each frame is composed of many very similar views, and it is therefore appropriate to amortize level-of-detail selection costs over all views instead of repeating a view-dependent render call for each of the projector images, and it is also important, for image continuity reasons, that all views agree on the same level-of-detail representation. This leads to an approach in which a common front-end system chooses a single per-frame level-of-detail. Second, nonlinear geometric and color correction must be performed to undo distortions due to lenses and approximate mechanical calibration, as well as to correct the different color, contrast, and intensity response of the projectors. This leads to an approach in which the usual linear pipe-line is fine-tuned by a nonlinear warping and photometric correction by a two-pass approach, as in multi-projector display systems [YGH\*01]. Finally, the multiresolution renderer cannot exploit the position of a particular viewer to select a level of detail, since an unlimited number of viewers must be free to move in a very large workspace in front of the display. In our approach, adaptive rendering exploits the finite spatial resolution of the display to perform adaptation.

As a matter of fact, by analyzing how the display works, it is easy to recognize that the smallest feature (voxel) that the display can reconstruct is not purely dictated by screen resolution as in conventional 2D displays, but depends on the distance  $s$  of the reconstructed voxel center from the screen and the beam angular resolution  $\Phi$ ,

$$\ell(s) = \ell_0 + 2\|s\| \tan(\Phi/2) \quad (1)$$

where  $\ell_0$  is the pixel size on the screen surface (see figure 4(a)). In other words, the achievable spatial resolution decreases with the distance from the screen. This is intuitive because the illusion of the existence of a particular spatial point is generated by pyramidal beams crossing at a specific 3D position. This fact also practically limits the field-of-depth of the display, i.e., the maximum distance from the screen at which objects are faithfully reconstructed. For instance, the accuracy of the display presented here varies from  $\ell_0 = \ell(0) = 1.17mm$  on screen to  $\ell(300mm) = 5.3mm$ . Based on these considerations, we developed a parallel spatial 3D display-aware version of the Adaptive TetraPuzzles technique [CGG\*04]. The method uses a distributed image generation system implemented on a cluster, with a front-end client PC selecting the level of detail from the multiresolution structure and multicasting graphics commands to back-end PCs. The characteristics of multiresolution techniques based on coarse grained adaptation are exploited to efficiently distribute data to back-end nodes as well as to efficiently pass them to the GPU through preferential paths.

The overall architecture of the rendering system is depicted in figure 3. Its main components are discussed in the following sections.

#### 4.1. Multiresolution structure overview

The Tetrapuzzles structure uses a conformal hierarchy of tetrahedra generated by recursive longest edge bisection to spatially partition the model in a preprocessing step. Each tetrahedral cell contains a precomputed simplified version of the original model. The representation is constructed offline during a fine-to-coarse parallel out-of-core simplification of the surface contained in diamonds (sets of tetrahedral cells sharing their longest edge). Appropriate boundary constraints are introduced in the simplification process to ensure that all conforming selective subdivisions of the tetrahedron hierarchy lead to correctly matching surface patches (see [CGG\*04] for more details). The main advantage of the method is its ability to rapidly produce seamless variable accuracy reconstructions by assembling precomputed patches.

Since rendering for the spatial 3D display requires the adaptation of vertices to very small (voxel-sized) triangles, controlling triangle shapes during simplification to reduce triangle counts in nearly flat areas is no longer important. Thus, instead of performing high-quality (quadric based) simplification [CGG\*04], we construct diamonds with a simplification method that produces (roughly) uniformly tessellated meshes, and use edge length as a measure of tessellation accuracy. This approach allows us to manage colored meshes by simply using a color-per-vertex representation.

#### 4.2. Front-end: selection of levels of detail

As in [CGG\*04], the nested subdivision hierarchy is encoded as a forest of binary trees, and we employ a saturation technique [OR98] to extract conforming meshes without requiring neighbor finding. Therefore, each tree is stored as a memory mapped linear array, and each of its nodes, corresponding

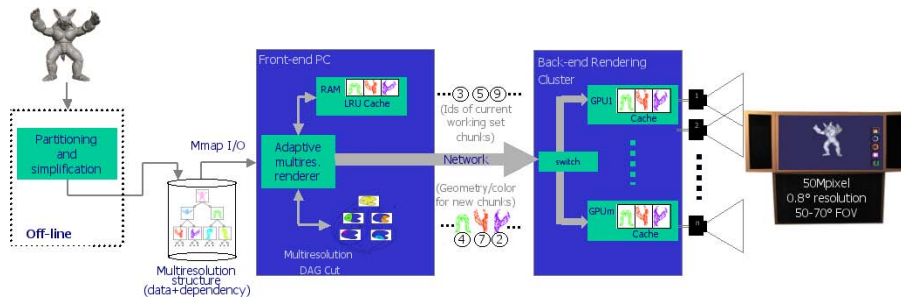
to a particular tetrahedron, contains just the following information: a reference to the associated patch data (vertex attributes and connectivity in stripified form) in a patch repository; the tight bounding sphere for the patch; the saturated model space average edge length and bounding sphere of the neighborhood (maximum among diamond's tight values and saturated values for children); the index of child nodes in the linear arrays, which corresponds to the two tetrahedra generated by bisection. With such a structure, variable resolution rendering is implemented by simple stateless top-down traversals of the binary trees used to encode the tetrahedron hierarchy, which combine view-frustum and contribution culling. The traversal is performed once per spatial 3D frame, and consequently generates as a result the set of patches that needs to be rendered for all the views. The standard view-dependent technique must thus be adapted to become the required spatial accuracy-dependent technique.

As we recurse the hierarchy, we test whether the current node is invisible by checking the tight bounding sphere of the associated patch against the spatial display working volume, determined by screen dimension, viewing angle, as well as achievable field of depth. If a node is found out of the working volume, we simply stop recursion, culling away the entire branch of the tree. If the node is potentially visible, we test whether its patch is an accurate enough representation by measuring its saturated spatial tessellation accuracy, which depends on its position within the volume. If so, we can add the associated patch to the active patch set for the frame, otherwise we continue the recursive refinement with the node's children.

Saturated spatial tessellation accuracy is the quantity that guides refinement, to achieve a target of (no more than) one vertex per voxel. Since the method exploits error saturation to encode dependencies, particular care must be taken to ensure that view-dependent measures are monotonically decreasing as we descend in the hierarchy and produce the same value for all tetrahedra in the same diamond. In our system, we obtain a consistent upper bound on the view-dependent error by measuring the apparent size of a sphere equal in diameter to the saturated average edge lengths of the patch and centered at the saturated bounding sphere point closest to the display's screen (see figure 4(b)). If this value is higher than the display's voxel resolution at that same position, computed from 1), the node needs refinement, otherwise we can safely stop refinement and consider the node for rendering.

At the end of the traversal, all nodes required for holographic rendering have been identified, and rendering can proceed by generating the projector images, which is done in parallel by all back-end PCs. Since all nodes share almost the same view, we use a sort-last distribution approach in which all patches to be rendered are broadcast to all rendering PCs without any sorting or filtering at the source. In order to save bandwidth, a LRU cache maintained in the front-end is exploited by sending only patches not already in cache and referring to already sent patches by patch id. Because of space-

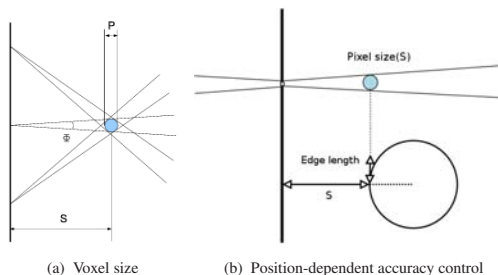




**Figure 3: Scalable rendering architecture.** A front-end client PC selects the level of detail from the multiresolution structure and multicasts graphics commands to back-end PCs that perform the actual rendering. Object-based communication with extensive caching leads to an efficient implementation.

time coherence, only few patches per frame need to be coarsened or refined and therefore caching aggressively reduces bandwidth requirement. Since all back-end nodes receive the same list of patches to be rendered, a multicast protocol can be effectively used to further reduce network load. In the current prototype, communication between front-end and back-end nodes goes through a dual gigabit switch supporting IGMP snooping. At the end of the frame, the front-end synchronizes with the back-end nodes through a barrier operation, performed before calling the hardware swapbuffers on all rendering nodes. Optionally, for control purposes, the front-end node can also render a frame by using the selected patches and showing them in a standard 2D window by taking the picture from a central viewpoint.

It must be noted that the TetraPuzzles structure offers additional front-end parallelization possibilities. In particular, since level of detail selection is implemented through a stateless recursive visit of a forest of binary trees, concurrent traversal is straightforward to implement. However, in our current implementation, we decided to keep the front-end sequential, since the typical size of the traversed coarse grained hierarchy is very small (typically one to few hundreds tetrahedra/frame) and traversal time is negligible in comparison with back-end nodes rendering times.



**Figure 4: Holographic display voxel size and position-dependent accuracy control.** The finite angular size of the light beams determines the voxel dimension as a function of distance from the screen.

### 4.3. Back-end: controlling the display

The rendering back-end consists of an array of PCs, consumer-level 3D graphics cards, and high-speed networking components that drives the 3D display by decoding the stream received by the front-end. Each of the back-end PCs is connected to the display by DVI connections and runs a server agent that controls an OpenGL framebuffer. The server is responsible for generating the images associated to a fixed subset of the display's projectors from the original stream (matrix transforms and patches). Even though in principle it is possible to use, for maximum performance, one PC per projector, benefit/cost analysis leads to a configuration in which multiple projectors are controlled by a single PC. Each back-end node must thus render each of the patches in the working set several times. For each of the back-end PCs, the server agent listens to the network and decodes the stream of multicast commands. Once decoded, the commands are interpreted to generate the images associated to a fixed subset of the display's projectors. When all commands for a given frame have been received, a rendering loop iterates on all associated projectors. All identified patches are then traversed and rendered from the projector's point of view in a projector's viewport. In order to take advantage of spatial and temporal coherency among views also in back-end nodes, each back-end node contains a memory manager, based on the same LRU strategy used in the front-end, which explicitly manages graphics board memory, using OpenGL's *Vertex Buffer Objects* to store patches. Each time we need to render a patch, we reuse the cached version if present, otherwise we render it and cache its representation in place of the oldest one. Least-recently used patches are deleted when the cache becomes full. By making sure that back-end caches are at least as big than the front-end one, we ensure that front-end and back-end caches remain properly synchronized (i.e., the front-end will never refer to a deleted object). Moreover, since all projectors share the same active object set, cache misses for a given patch can happen at most once per spatial frame. Therefore, rendering  $N$  projector views costs less than  $N$  times the rendering of a single view. At end-of-frame, as in all tiled projector displays (see, e.g., [BGH\*02]), non-linear photometric and geometric corrections are applied in



**Figure 5: Interaction sequence.** These images, taken from the accompanying video, show successive instants of interactive manipulation of the multi-million triangles colored datasets on the 50Mpixel display.

a post-pass before synchronizing with the frame barrier and swapping buffers.

## 5. Implementation and Results

We have implemented a prototype hardware and software system based on the design discussed in this paper. The display hardware and software components have been realized by Holografika ([www.holografika.com](http://www.holografika.com)). The multiresolution renderer discussed here has been designed and implemented by the authors. The developed large scale display is already capable of visualizing 50M pixels by composing images generated by 64 XGA commodity projectors. The display provides continuous horizontal parallax with an approximately 45 degrees horizontal field-of-view. The luminance is over 5000 lumen (10000 lumen in high brightness mode) and allows the display to work under almost any kind of ambient lighting conditions.

The parallel multiresolution rendering front-end runs on Linux on an Athlon64 3300+ PC with a NVIDIA7800GT graphics board and a local SATA disk for storing models. The rendering back-end, which drives the 64 projectors, is currently running on an array of 16 Athlon64 3300+ Linux

PCs equipped with two NVIDIA6600 graphics boards running in twin-view mode (i.e., each back-end node controls four projectors through four DVI outputs).

It is obviously impossible to fully convey the impression provided by our system on paper or video. As a simple illustration of our system's current status and capabilities, an accompanying video shows an interactive sequence recorded live using a moving camera. The interactive sequences consist in a short free-hand manipulation of two high resolution laser scan datasets: the Michelangelo's David 1mm model (56M triangles) and a colored wooden statue dataset (6M triangles) acquired by combining a laser scan with high resolution digital photographs. As demonstrated in the video, objects appear floating in the display space and, despite their massive size, can be interactively manipulated by translating, rotating, and scaling them interactively in 3D space using a Logitech 3D mouse input device for direct manipulation. Representative video frames are shown in figure 5. The sequences were recorded with a hand held video camera, freely moving in the display's workspace. Note the parallax effects and the good registration between displayed object space and physical space, which demonstrate the multi-user

capability of the display. As illustrated by the video, the perceived image is fully continuous. This is qualitatively very different from other contemporary multiview display technologies, which force users into approximately fixed positions, because of the abrupt view-image changes that appear at the crossing of discrete viewing zones [MP04].

Only few patches per frame need to be updated when the object is rotated or translated. On the other hand scaling the model typically requires updating most of its representation, since the triangle-size/voxel ratio rapidly changes. This implies the creation and transfer to the back-end GPUs of new patches, which is the most critical work done by the renderer. Both models have about 2000 triangles per patch and each VBO patch takes up about 30KB (1000 vertices with 3 float per position, 3 short per normal and 4 bytes per color, plus about 3500 indices stored as short). Zooming the David model from the minimum scale (85 patches) to the maximum scale (360 patches) requires the creation of about 1000 new patches to go through all intermediate representations from the farthest to the nearest view. We tested the back-end which is able to create up to 1200 VBO patches per second corresponding to a bandwidth of 280Mb/s, but the program is not generally pushed to this limit. If we perform this zooming operation in about 5 seconds, the renderer has to create 200 patches per second, which means that we need roughly 50Mb/s of bandwidth, which is far less than the maximum throughput of our network. During zooming there can be some decay of rendering performance. In order to have a fully interactive rendering another solution is to use a dual queue algorithm, as the one presented in BMT [CGG\*05], where the cut extraction process is interruptible and the renderer has a time budget for this operation. The actual interactive application frame rate ranges from 8 to 25 Hz, depending on the scale of the object and on the number of new patches created, with a mean throughput of 10Mtri/s.

## 6. Conclusions and Future Work

We have presented a practical working implementation of an interactive holographic environment that allows freely moving naked-eye participants to share and manipulate massive 3D geometric models with fully continuous, observer-independent, parallax. The system does not require users to wear any input or output devices. The display is an instance of a novel scalable holographic system design based on a specially arranged array of projectors and a holographic screen, and is driven by a commodity graphics platform. Multiresolution techniques, which take into account the particular display configuration, are employed to dynamically adapt model resolution to display capabilities and timing constraints. The main take home message of this work is that spatial 3D display technology is here, and is here to stay, and that coarse-grained view-dependent multiresolution rendering techniques designed for single user view-dependent rendering can be effectively transformed into 3D display-aware spatially adaptive sort-last cluster parallel renderers.

State-of-the-art 3D display technology can thus be exploited to explore extremely complex and highly detailed datasets.

The prototype discussed here is clearly meant to work as an enabling technology demonstrator, as well as a testbed for integrated 3D interaction, massive model visualization, and display research. While not all techniques are novel in themselves, their combination is definitely pushing forward the state of the art.

**Acknowledgments.** This research is partially supported by the CYBERSAR project. The authors are grateful to all the Holografika team for the design and implementation of display hardware and low-level software. We also acknowledge the key contribution of Eric Bouvier from C-S in the design and implementation of a distributed OpenGL graphics library.

## References

- [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 2005 Short Papers Proceedings* (2005).
- [BGB\*05] BORGEAT L., GODIN G., BLAIS F., MASSI-COTTE P., LAHANIER C.: GoLD: interactive display of huge colored and textured models. *ACM Trans. Graph* 24, 3 (2005), 869–877.
- [BGH\*02] BINNS J., GILL G., HERELD M., JONES D., JUDSON I., LEGGETT T., MAJUMDER A., AMD MICHAEL PAPKA M. M., STEVENS R.: Applying geometry and color correction to tiled display walls. In *IEEE Visualization - Poster sessions* (2002).
- [CGG\*04] CIGNONI P., GANOVELLI F., GOBBETTI E., MARTON F., PONCHIO F., SCOPIGNO R.: Adaptive TetraPuzzles – efficient out-of-core construction and visualization of gigantic polygonal models. *ACM Transactions on Graphics* 23, 3 (August 2004), 796–803. Proc. SIGGRAPH.
- [CGG\*05] CIGNONI P., GANOVELLI F., GOBBETTI E., MARTON F., PONCHIO F., SCOPIGNO R.: Batched multi triangulation. In *Proc. IEEE Visualization* (Oct. 2005), pp. 207–214.
- [CKLL05] CHOI K., KIM J., LIM Y., LEE B.: Full parallax viewing-angle enhanced computer-generated holographic 3d display system using integral lens array. *Optic express* 13, 26 (December 2005), 10494–10502.
- [CNC\*05] CHUN W.-S., NAPOLI J., COSSAIRT O., DORVAL R., HALL D., AMD JAMES SCHOOLER T. P., BANKER Y., FAVALORA G.: Spatial 3-D infrastructure: Display-independent software framework, high-speed rendering electronics, and several new displays. In *Proc. SPIE-IS&T Electronic Imaging* (2005), vol. 5664, SPIE, pp. 302–312.

- [DML\*00] DODGSON N. A., MOORE J. R., LANG S. R., MARTIN G., CANEPA P.: Time-sequential multi-projector autostereoscopic 3D display. *J. Soc. for Information Display* 8, 2 (2000), 169–176.
- [Dod96] DODGSON N. A.: Analysis of the viewing zone of the cambridge autostereoscopic display. *Applied Optics: Optical Technology & Biomedical Optics* 35, 10 (1996), 1705–1710.
- [Dod05] DODGSON N. A.: Autostereoscopic 3d display. *Computer* 38, 8 (2005), 31–36.
- [Dod06] DODGSON N. A.: On the number of views required for head-tracked autostereoscopic display. In *Stereoscopic Displays and Applications XVII* (2006), vol. 6055 of *SPIE proceedings*.
- [EWO\*95] EZRA D., WOODGATE G. J., OMAR B. A., HOLLIMAN N. S., HARROLD J., SHAPIRO L. S.: New autostereoscopic display system. In *Stereoscopic Displays and Virtual Reality Systems II* (1995), vol. 2409 of *SPIE proceedings*, pp. 31–40.
- [FDHN01] FAVALORA G., DORVAL R., HALL D., NAPOLI J.: Volumetric three-dimensional display system with rasterization hardware. In *Stereoscopic Displays and Virtual Reality Systems VII* (2001), vol. 4297 of *SPIE Proceedings*, pp. 227–235.
- [HHN\*02] HUMPHREYS G., HOUSTON M., NG R., FRANK F., AHERN S., KIRCHNER P., KLOSOWSKI K.: Chromium: A stream-processing framework for interactive rendering on clusters. In *Proc. SIGGRAPH* (2002), Annual Conference Series, pp. 693–702.
- [HMG03] HUEBSCHMAN M., MUNJULURI B., GARNER H.: Dynamic holographic 3-d image projection. *Optics Express* 11 (2003), 437–445.
- [KPC\*05] KIM Y., PARK J.-H., CHOI H., KIM J., CHO S.-W., LEE B.: Depth-enhanced integral imaging with multiple central depth planes using multilayered display device. In *Three-Dimensional TV, Video, and Display IV* (2005), vol. 6016 of *Proc. SPIE*, pp. 24–31.
- [MKL05] MIN S.-W., KIM J., LEE B.: Enhanced integral imaging system using image floating technique. In *Optical Information Systems III* (2005), vol. 5908 of *Proc. SPIE*, pp. 73–80.
- [MMMR00] MCKAY S., MAIR G., MASON S., REVIE K.: Membrane-mirrorbased autostereoscopic display for teleoperation and telepresence applications. In *Stereoscopic Displays and Virtual Reality Systems VII* (2000), vol. 3957 of *SPIE Proceedings*, pp. 198–207.
- [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.
- [OR98] OHLBERGER M., RUMPF M.: Adaptive projection operators in multiresolution scientific visualization. *IEEE Transactions on Visualization and Computer Graphics* 4, 4 (1998), 344–364.
- [PPK00] PERLIN K., PAXIA S., KOLLIN J. S.: An autostereoscopic display. In *Proc. SIGGRAPH* (2000), Annual Conference Series, pp. 319–326.
- [RR05] RELKE I., RIEMANN B.: Three-dimensional multiview large projection system. In *Stereoscopic Displays and Virtual Reality Systems XII* (2005), vol. 5664 of *Proc. SPIE*.
- [RS00] ROBERTS J. W., SLATTERY O.: Display characteristics and the impact on usability for stereo. In *Stereoscopic Displays and Virtual Reality Systems VII* (2000), vol. 3957 of *SPIE proceedings*, p. 128.
- [SCC\*00] STANLEY M., CONWAY P., COOMBER S., JONES J., SCAT-TERGOOD D., SLINGER C., BANNISTER B., BROWN C., CROSSLAND W., TRAVIS A.: A novel electro-optic modulator system for the production of dynamic images from giga-pixel computer generated holograms. In *Practical Holography XIV and Holographic Materials VI* (2000), vol. 3956 of *SPIE Proceedings*, pp. 13–22.
- [SHLS\*95] ST.-HILLAIRE P., LUCENTE M., SUTTER J., PAPPU R., SPARRELL C. G., BENTON S.: Scaling up the mit holographic video system. In *Proc. Fifth International Symposium on Display Holography* (1995), SPIE, pp. 374–380.
- [vBC97] VAN BERKEL C., CLARKE J.: Characterisation and optimisation of 3d-lcd module design. In *Stereoscopic Displays and Applications XVII* (1997), vol. 3012 of *SPIE proceedings*, pp. 179–186.
- [vPF96] VAN BERKEL C., PARKER D., FRANKLIN A.: Multiview 3d-lcd. In *Stereoscopic Displays and Virtual Reality Systems III* (1996), vol. 2653 of *SPIE proceedings*, p. 32.
- [WEH\*98] WOODGATE G. J., EZRA D., HARROLD J., HOLLIMAN N. S., JONES G. R., MOSELEY R. R.: Autostereoscopic 3d display systems with observer tracking. *Image Communication - Special Issue on 3D Video Technology (EURASIP - 1998)* (1998), 131.
- [WHJ\*00] WOODGATE G. J., HARROLD J., JACOBS A. M. S., MOSELEY R. R., EZRA D.: Flat-panel autostereoscopic displays: characterisation and enhancement. In *Stereoscopic Displays and Virtual Reality Systems VII* (2000), vol. 3957 of *SPIE proceedings*, p. 153.
- [YGH\*01] YANG R., GOTZ D., HENSLEY J., TOWLES H., BROWN M. S.: PixelFlex: a reconfigurable multi-projector display system. In *Proc. IEEE Visualization* (Oct. 21–26 2001), pp. 167–174.
- [YSGM04] YOON S.-E., SALOMON B., GAYLE R., MANOCHA D.: Quick-vdr: Interactive view-dependent rendering of massive models. In *Proc. IEEE Visualization* (2004), pp. 131–138.