Capturing the Shape of a Dynamic World - Fast !

Marcus Magnor and Hans-Peter Seidel
Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
D-66123 Saarbrücken, Germany
magnor/hpseidel@mpii.de

Abstract

Acquiring on-line the evolving shape of a dynamic scene from a handful of video streams may be considered one of the most challenging, but at the same time also most auspicious tasks in contemporary computer graphics and computer vision research. The anticipation of revolutionary new applications such as interactive 3D television broadcasts motivates the ongoing work on free-viewpoint video rendering. This paper aims at giving a state-of-progress report on this lively research endeavour. Different acquisition setups and on-line reconstruction approaches are exemplified. Yielding interactive frame rates, depth map-based techniques, polyhedral as well as volumetric visual hull reconstruction approaches, and combined methods employing visual hull-guided depth map estimation are presented. The experience gained with these approaches allows identifying future research directions towards real-time analysis and high-quality synthesis of dynamic, real-world scenes.

1. Introduction

The gap between the real and the virtual world is closing rapidly. Advances in imaging devices, reconstruction algorithms, and rendering techniques are paving the way towards incorporating natural objects and entire scenes into computer graphics applications. Work in both computer vision and computer graphics has been converging in recent years to form today a strong foundation for research in photo-realistic rendering [11]. By taking advantage of multiple photographs of a real-world object, image-based rendering (IBR) techniques are capable of generating natural-appearing object views from arbitrary perspective [15, 7]. Image-based modeling approaches, on the other hand, are able to determine object shape from multi-view image data [10, 25, 9]. Shape information is imperative when merging natural objects into virtual environments. Editing of real-world augmented virtual scenes as well as illumination and shadow computations both require 3D geometry models of the natural objects to be incorporated. Finally, by combining shape with imagery, photo-realistic rendering results are attainable from much fewer images than needed in pure IBR [29, 2], illustrating the usefulness and importance of shape information also for image-based rendering.

Work in both image-based rendering as well as image-based modeling has been focusing mainly on static objects and scenes. Only recently, synchronized video camera arrays have been built to provide multi-video data for analyzing *dynamic* scenes. Among these multi-video acquisition systems, two different recording scenarios can be distinguished: To acquire *Dynamic Light Fields*, a large number of synchronized cameras are spaced closely together [21, 28, 31]. The recorded video streams exhibit only small parallax between adjacent cameras, and pure IBR techniques without additional object shape information can be empoyed to render intermediate views. However, dynamic light field acquisition requires considerable hardware effort, custom-built electronics, and a complex recording setup. Multi-video recording systems consisting of only a handful of cameras, on the other hand, are easier to set up, can be built from consumer-market components, and are much less expensive [23, 20, 30, 18, 19]. Here, the cameras are typically spaced far apart all around the scene, so the camera views differ considerably due to parallax, occlusion, and changing visibility. For *Free-Viewpoint Video Rendering* from such sparse sets of camera perspectives, reconstruction of scene shape information is essential.

Multi-view video streams are envisioned to be the interfacing input modality between the real world and various new computer graphics applications: For computer games, a key factor to success is the degree of perceived realism. Adopting character animations and textures from real-world examples will yield an unprecedented sensation of presence in computer games. In sports, athletes will benefit from reviewing their motor activity from arbitrary perspective and from comparing them to their competitors' motion

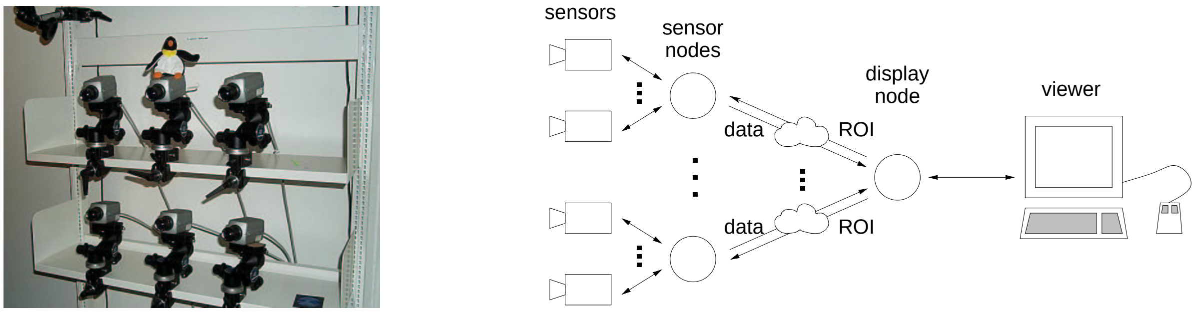

Figure 1. The *Lumi-Shelf* setup records three synchronized stereo-image pairs. Distributed processing of the video data enables interactive frame rendering rates.

sequence. In surveillance applications, sensitive areas may not have to be patrolled by trustworthy watchmen anymore, but a faster and more secure remote “virtual patrol” of a multi-camera observed area will become possible.

One prospective application of free-viewpoint video rendering is *Interactive 3D Television*. Instead of passively watching a sequence of 2D images, the viewer interactively chooses his position and viewing direction in the dynamic 3D scene. Intriguing applications range from watching movies and series from the viewpoint of any actor to sports broadcasts where the soccer fan may decide to experience the game, e.g., from the referee’s point of view. Already, the MPEG consortium has founded the ad-hoc group MPEG-3DAV to investigate the requirements and applications of interactive 3D-TV [26]. The research results compiled in this paper originated in work done in this context [6].

One major challenge in free-viewpoint video rendering for interactive 3D-TV applications consists of the time-critical nature of live television broadcasts. Since, realistically, only a small number of cameras can be placed along the scene perimeter, free-viewpoint video rendering is possible only if the 3D scene shape is available. Consequently, the shape of dynamic scene constituents must be reconstructed from multi-video footage in real-time.

In this paper, several approaches are presented that are capable of reconstructing shape and render from synchronized video streams interactively. From hierarchical depth map estimation in the following Section and polyhedral as well as volumetric visual hull reconstruction techniques in the subsequent Sections to a combined hybrid depth map-visual hull approach, on-line capable shape reconstruction and rendering solutions are evaluated with regard to speed and quality performance. Experience gained with these approaches allow identifying potential future research directions towards a real-time capable, high fidelity, interactive 3D-TV system.

2. Hierarchical Depth Map Estimation for On-Line Warping

The most straightforward way to constructing arbitrary views from multi-view imagery consists of using per-pixel depth information to warp the available images to the desired viewpoint [3]. Estimating depth maps at interactive frame rate on conventional hardware, however, is tantamount to pushing PC computation capabilities to its limits.

The setup shown in Fig. 1 allows investigating how efficiently a depth-map based free-viewpoint system performs using contemporary consumer-market hardware [24]. The *Lumi-Shelf* consists of three calibrated Sony DFWV500 color video camera pairs. Each camera pair is connected via IEEE1394 (FireWire) bus to one PC running Linux. The three sensor host PCs are connected via 100 MBit Ethernet LAN to the display host, Fig. 1. To render a desired view, the display host determines the *region of interest* (ROI) from each camera pair and sends the ROI requests to the sensor nodes. The ROI is an estimate of the subset of the image needed from each sensor, significantly reducing depth-map computation cost as well as transmission load. From its synchronized image pair, each sensor host estimates a dense depth map: Using previously acquired calibration data, both images are rectified, and a block matching-based hierarchical depth-from-stereo algorithm determines the pixel disparity between the images. Being able to trade quality for speed, near-interactive frame rates of up to 2 frames/second can be achieved at 320×240-pixel image resolution. For rendering, RGB color values and associated depth of all pixels inside the respective ROI are transmitted from the sensor nodes to the display host. On the display host, the recorded image regions are warped to the desired viewpoint. To obtain the final rendering result, the warped partial images are composited by taking into account occlusions and by smoothly blending regions visible in more than one camera pair. Rendering frame rates of 9 frames/sec at 256×256 pixels resolution are attainable on a single-processor PC.

Figure 2. 3D-warped intermediate views using depth maps estimated on-line: Missing depth estimates lead to holes, while erroneous pixel depth values introduce isolated pixels.

Performance of depth map-based free-viewpoint video rendering is limited by the speed and the quality of depth-from-stereo estimation. Disparity search range and image resolution determine the algorithm's execution time, while occlusions, featureless scene surfaces and non-Lambertian reflection characteristics lead to missing or erroneous depth estimates, Fig. 2.

To obtain higher-quality rendering results, previously acquired images of the static background can be used to segment the dynamic foreground, restricting shape reconstruction computations to the dynamic object only. This leads to the idea of reconstructing dynamic object shape in terms of *visual hulls*.

3. Real-Time Polyhedral Visual Hulls

Given several outlines of an object corresponding to different viewing directions, the approximate shape of the object can be estimated. When back-projected into space, these outlines intersect to form the object's *visual hull* [10, 20]. Well-calibrated cameras and segmented images of the object are necessary to perform visual hull reconstruction, Fig. 3.

Experiments with on-line visual hull reconstruction and rendering have been performed using six Sony DFWV500 video cameras that are distributed about equally along the perimeter of a stage, Fig. 4 [16]. At 4.8 mm focal length, the observed volume visible in all camera views measures approximately $2 \times 2 \times 2$ meters. Video is captured at 320×240-pixel resolution in RBG color, and the images from two cameras are downstreamed to one PC via IEEE1394 bus. Similar to the network architecture shown in Fig. 1, the three client PCs are connected via Ethernet LAN to one server PC which synchronizes video acquisition at the maximally attainable 13 fps capture frame rate when triggering the Sony DFWV500 cameras externally.

To be able to segment moving foreground objects, the (arbitrary) static background is pre-recorded prior to introducing dynamic scene elements [1]. Statistical measures on the difference in pixel color between the pre-acquired and the current video image are used to classify each pixel as background or foreground, Fig. 5. Undecidable cases typically occur only along the silhouette and for shadows on the floor. Any holes in the silhouette are closed using morphological filters. On the client PCs, object silhouette is extracted, and a 2D polygon is fit to the contour of the silhoutte. By varying the silhouette polygon's number of vertices, visual hull quality can be traded for reconstruction frame rate. The silhouette polygons are transmitted to the server PC where they are back-projected into space from their respective camera positions, and the intersection

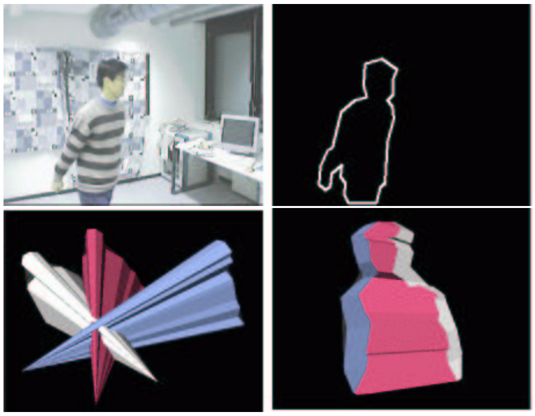

Figure 3. Shape-from-silhouette: The outlines of an object from different viewpoints intersect in space to form an approximation of the object's visual hull.

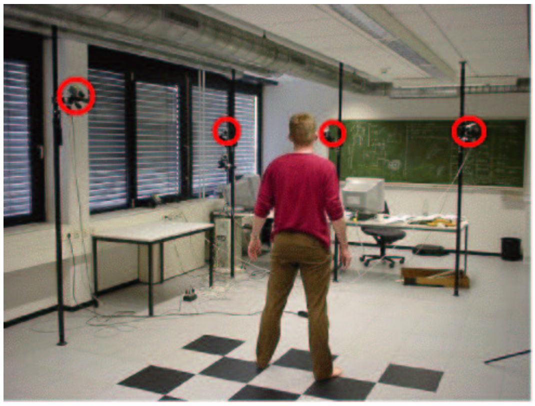

Figure 4. Six externally triggered video cameras record the scene from all around. The checkerboard pattern on the ground aids in camera calibration.

Figure 5. From pre-acquired background images, the dynamic foreground object is segmented exploiting statistical differences in pixel color.

of the silhouette cones is computed. The visual hull is represented by the resulting polyhedron. Object segmentation, silhouette approximation and polyhedral visual hull reconstruction consisting of about 500 triangles is performed in tandem at higher than the maximum camera frame rate of 13 frames per second.

For rendering, the video images of the segmented foreground are packed and sent to the server PC. To multi-video texture the surface of the visual hull polyhedron, projective texture mapping is applied using graphics hardware, Fig. 11 (cf. color plate). View-dependent texturing is achieved by blending overlapping texture regions and weighting the images corresponding to their recording position's angular distance to the current viewing direction [5]. Shadow mapping techniques prevent texturing-through artifacts while applying projective texturing. Using six input camera views, the dynamic object is rendered at 27 fps on an NVidia GeForce3 graphics card, more than twice as fast as the cameras' capture frame rate [16].

4. Fast Shape Reconstruction with Volumetric Visual Hulls

A different way to reconstructing the shape of a dynamic object from its silhouettes consists of performing silhouette intersection in discretized space [27]. Here, the confined volume of space that is visible in all camera views is subdivided into cubic volume elements (voxels), Fig. 7. To reconstruct a multi view-consistent volumetric model, each voxel is re-projected into all camera views. Only if a voxel's footprint falls inside the object silhouette in all images is the voxel classified as belonging to the object's visual hull. All other voxels are deleted. From the calibrated cameras, voxel re-projection locations in all images are pre-computed. Using the distributed client-server architecture described in Sect. 3, silhouette extraction and partial volume reconstruction from two views is performed on each client PC, Fig. 6. The partial volumes are run-length-encoded and transmitted from the three client PCs to the server where the partial volumes need only be intersected by a boolean AND operation. The entire model is reconstructed at 7 fps for a 64 × 64 × 64-voxel volume, Fig. 12 (cf. color plate).

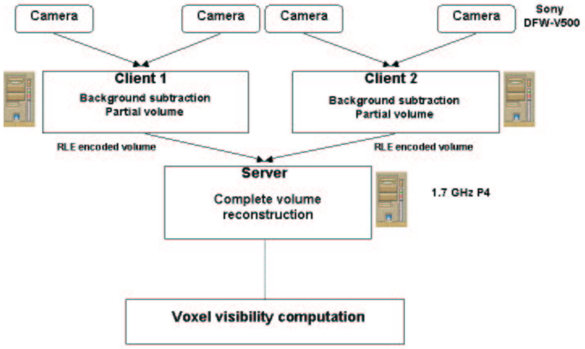

Figure 6. Two cameras are connected to one client PC that extracts the image silhouettes and determines a partial volume by re-projecting all voxels into the segmented images.

To texture the volumetric model, each voxel is rendered as a single geometric primitive called a billboard. Each billboard is a small rectangle parallel to the screen plane centered at the position of its corresponding voxel. From the voxel footprint in the camera images, the re-projection coordinates of the billboard are known and can be used as texture coordinates. However, voxel visibility must be determined prior to texturing in order to render only surface voxels and to texture only from camera views that are visible for each voxel. The mapping is pre-computed and efficiently evaluated using hardware-accelerated dependent

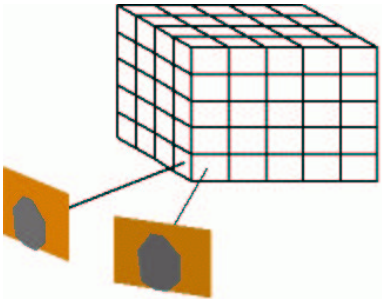

Figure 7. The volume containing the dynamic object is discretized into volume elements (voxels). For reconstruction, each voxel is re-projected into all images, and only voxels falling inside all silhouettes are classified as belonging to the object's visual hull.

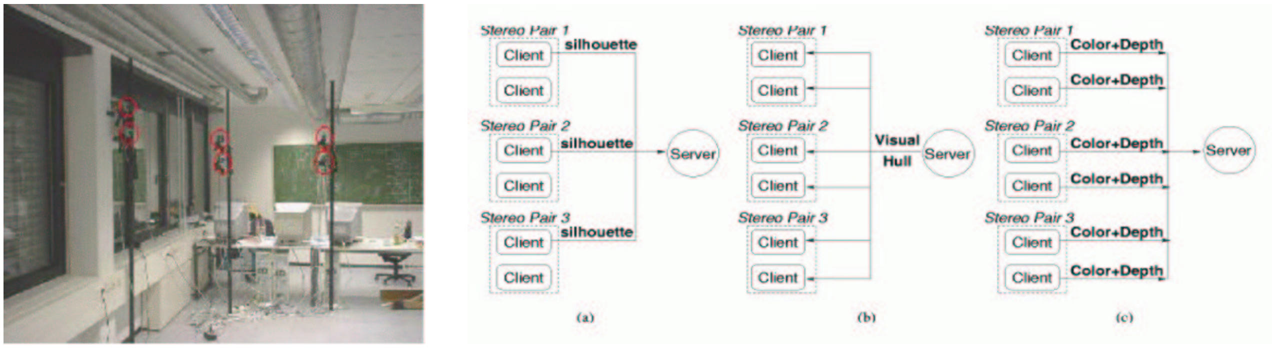

Figure 8. For combining visual hull reconstruction with depth map estimation, six cameras are grouped in pairs of two. The camera pairs are distributed along a 90°-wide arc about the scene. Silhouette extraction is performed on the client PCs. The outlines are sent to the server to compute the polyhedral visual hull that is sent back to the clients. There, the visual hull is used to guide depth estimation for the object pixels. Object pixel color and depth is finally sent back to the server for rendering.

texturing. Graphics hardware is also used to blend the contributions from non-occluded cameras closest to the current viewpoint. Background pixels are rendered transparently, so fine details along the object outline are preserved even for a relatively coarse resolution of the voxel model.

For each frame of the video sequence, the segmented images as well as the partial volumes are transferred to the server PC and on to the graphics card. For a $32 \times 32 \times 32$ voxel model, 30 fps rendering frame rate are attained using an NVidia GeForce4 graphics card, Fig. 12 (cf. color plate). Volumetric visual hull rendering quality is determined by the resolution and the accuracy of the voxel model. By applying view-dependent texturing to render the billboard facets, images from different directions are projected onto the billboard. Even a small error in billboard depth causes the image projections to mismatch slightly, leading to unsharp edges and blurred features. Unfortunately, this cannot be overcome by increasing model resolution or using more video cameras, since the visual hull can in principle not reconstruct *concave* object regions that lie inside any silhouette. However, the fast and elegant visual hull technique can be augmented with the depth map approach of Sect. 2 to obtain improved shape reconstructions at interactive performance rates.

5. Using the Visual Hull for Improved Depth Map Estimation

While the visual hull of an object can be reconstructed fast and robustly from a number of silhouette images, estimating depth is time-consuming and error-prone. However, depth maps may yield accurate per-pixel depth information, whereas visual hulls are in principle only approximations to the object's actual shape, Fig. 9. Fortunately, by combining visual hull reconstruction with depth map estimation, the beneficial properties of both techniques can be preserved to yield a superior shape reconstruction algorithm [17].

Grouping six video cameras into pairs of two and distributing the camera pairs along a 90°-wide arc about the scene, the visual hull as well as pairwise depth maps can be computed, Fig. 8. As in Sect. 3, the three client PCs, each connected to one camera pair, extract the object silhouettes and transmit them to the server where the polyhedral visual hull is recontructed, Fig. 8. The visual hull is not rendered, however, but sent back to the client PCs. Here, the visual hull information is used to initialize the depth maps and to restrict the search range. Visual hull-derived pixel depth represents the lower bound (closest possible distance to camera) for the actual depth of a pixel, Fig. 10. The upper bound (farthest possible distance to camera) can be determined from the visual hull which yields, however, an unnecessarily conservative depth range. Depending on the amount of concavity expected for the object, the depth range is determined from the lower bound by adding a preset maximum range value, speeding up depth map estimation considerably. In addition, the visual hull provides initial depth estimates for all object pixels, so the depth maps exhibit no undefined regions anymore. The considerable overhead of transmitting the silhouettes and the visual hull over the network and synchronizing the calculations currently limit reconstruction frame rate to 4 fps at an image resolution of 320×240 pixels.

Novel views are rendered by 3D warping the source images to the desired view, Fig. 13 (cf. color plate). By decoupling rendering from shape reconstruction and warping the stereo image pair that is closest to the desired viewing

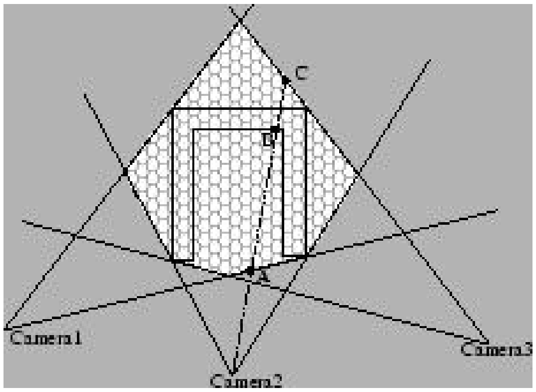

Figure 9. Object concavities can in principle not be reconstructed by shape-from-silhouette methods (point D). However, the visual hull allows constraining the depth search range (from point A to point C).

direction, rendering frame rates of 10 fps are achieved on an NVidia GeForce2 graphics card [17].

6. Outlook on Future Free-Viewpoint Video Research

Reconstructing dynamic shapes from a handful of video streams for interactive viewing applications is a challenging and intriguing research area on the borderline between computer graphics and computer vision. Fast depth map estimation, on-line polyhedral and volumetric visual hull reconstruction, and the combination of both approaches are among the most promising techniques for real-time free-viewpoint video rendering. Despite recent progress, a lot of research remains to be done. First and foremost, temporal coherence may be exploited to obtain continuously evolving shapes. Towards this goal, a generic model of the dynamic object in the scene may be fitted to the extracted object silhouettes, taking advantage of coherence due to object surface contiguity as well as object animation constraints [12, 8]. In addition, a generic model allows for texture extrapolation as well as for the incorporation of a-priori object texture information to render intermittently not visible object regions. However, a suitably parameterized geometry model must be available, and the necessary assumptions about scene content restrict general applicability.

Another direction of future research concerns the reconstruction of object surface reflection properties [14]. To be able to edit the illumination of virtual scenes that have been augmented by real-world objects, non-Lambertian surface characteristics must be known to be considered correctly. On the same note, object surface reflection properties may be exploited to estimate object shape more robustly [13]. For estimating object reflection properties, pre-acquired high dynamic range recordings of the illuminating environment may be necessary. In conjunction with object shape, known illumination also allows detecting shadowed object regions and to remove illumination effects. Future work will be comprised of enhancing existing real-time algorithms for dynamic 3D scene analysis by exploiting additional knowledge about the scene and to incorporate the analysis results into a high quality, interactive free-viewpoint video rendering system.

Figure 10. A dense depth map is obtained for the segmented foreground object at 4 frames per second by using the object's visual hull to intialize and restrict depth estimation.

When comparing the presented results to the image quality of television broadcasts, it must be kept in mind that TV technology has had more than 70 years to mature, while free-viewpoint video rendering is not much older than 2 years. Despite its infancy, however, free-viewpoint video rendering is already drawing considerable commercial attention [4, 26, 22, 32]. This may be taken as the best indictation that research into capturing the shape of a dynamic world is a fascinating and worthwhile research endeavour.

7. Acknowledgments

The results in this paper represent a compilation of research work done by Bastian Goldlücke, Ming Li, Hartmut Schirmacher and Christian Theobalt at the MPI für Informatik. For latest results, please be referred to their web pages at *www.mpi-sb.mpg.de.*

References

[1] M. Bichsel. Segmenting simply connected moving objects in a static scene. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16(11):1138–1142, 1994.

[2] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen. Unstructured lumigraph rendering. *Proc. ACM Confer-*

ence on Computer Graphics (SIGGRAPH-2001), Los Angeles, USA, pages 425–432, Aug. 2001.

[3] S. Chen and L. Williams. View interpolation for image synthesis. *Proc. ACM Conference on Computer Graphics (SIGGRAPH'93),* pages 279–288, 1993.

[4] M. O. de Beeck and A. Redert. Three dimensional video for the home. *Proc. EUROIMAGE International Conference on AUgmented, Virtual Environments and Three-Dimensional Imaging (ICAV3D'01),* Mykonos, Greece, pages 188–191, May 2001.

[5] P. Debevec, C. Taylor, J. Malik, G. Levin, G.Borshukov, and Y. Yu. Image-based modeling and rendering of architecture with interactive photogrammetry and view-dependent texture mapping. *Proc. IEEE International Symposium on Circuits and Systems (ISCAS'98),* Monterey, USA, 5:514–517, June 1998.

[6] B. Goldlücke, M. Magnor, M. Li, and C. Theobalt. EE2 - methods for model-based scene reconstruction and real-time free viewpoint rendering, Dec. 2002. MPEG Doc. M9145, Awaji Island, Japan.

[7] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The lumigraph. *Proc. ACM Conference on Computer Graphics (SIGGRAPH'96),* New Orleans, USA, pages 43–54, Aug. 1996.

[8] K. Klein, W. Cornelius, T. Wiebesiek, and J. Wingbermühle. Creating a "personalised, immersive sports tv experience" via 3D reconstruction of moving athletes. In Abramowitz and Witold, editors, *Proc. Business Information Systems*, 2002.

[9] K. Kutulakos and S. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[10] A. Laurentini. The visual hull concept for sihouette-based image understanding. *IEEE Trans. Pattern Analysis and Machine Vision*, 16(2):150–162, Feb. 1994.

[11] J. Lengyel. The convergence of graphics and vision. *Computer*, 31(7):46–53, July 1998.

[12] H. Lensch, W. Heidrich, and H.-P. Seidel. A silhouette-based algorithm for texture registration and stitching. *Graphical Models*, 63(4):245–262, 2001.

[13] H. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H.-P. Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Transactions on Graphics*, 22(2):27, April 2003.

[14] H. Lensch, J. Kautz, M. Gösele, W. Heidrich, and H.-P. Seidel. Image-based reconstruction of spatially varying materials. *Proc. ACM 12th Eurographics Workshop on Rendering,* London, UK, pages 104–115, 2001.

[15] M. Levoy and P. Hanrahan. Light field rendering. *Proc. ACM Conference on Computer Graphics (SIGGRAPH'96),* New Orleans, USA, pages 31–42, Aug. 1996.

[16] M. Li, M. Magnor, and H.-P. Seidel. Online accelerated rendering of visual hulls in real scenes. *Proc. International Conference in Central Europe on Computer Graphics WSCG'03,* Plzen, Czech Republic, Feb. 2003.

[17] M. Li, H. Schirmacher, M. Magnor, and H.-P. Seidel. Combining stereo and visual hull information for on-line reconstruction and rendering of dynamic scenes. *Proc. IEEE Multimedia Signal Processing MMSP-2002,* St. Thomas, US Virgin Islands, Dec. 2002. ISBN 0-7803-7713-3.

[18] J. Luck and D. Small. Real-time markerless motion tracking using linked kinematic chains. In *Proceedings of the International Conference on Computer Vision, Pattern Recognition and Image Processing 2002 (CVPRIP02), in cooperation with JCIS 2002*, 2002.

[19] T. Matsuyama and T. Takai. Generation, visualization, and editing of 3D video. In *Proc. of 1st International Symposium on 3D Data Processing Visualization and Transmission (3DPVT'02)*, page 234ff, 2002.

[20] W. Matusik, C. Buehler, and L. McMillan. Polyhedral visual hulls for real-time rendering. *Proc. 12th Eurographics Workshop on Rendering,* London, UK, pages 115–126, June 2001.

[21] R. Ooi, T. Hamamoto, T. Naemura, and K. Aizawa. Pixel independent random access image sensor for real time image-based rendering system. *Proc. International Conference on Image Processing (ICIP'01),* Thessaloniki, Greece, pages 193–196, Oct. 2001.

[22] PVI Inc. / EyeVision. Lawrenceville, NJ USA. http://www.pvi-inc.com/eyevision.

[23] H. Saito, S. Baba, M. Kimura, S. Vedula, and T. Kanade. Appearance-based virtual view generation of temporally-varying events from multi-camera images in the 3D room. In *Proceedings of Second International Conference on 3-D Digital Imaging and Modeling*, pages 516 – 525, October 1999.

[24] H. Schirmacher, L. Ming, and H.-P. Seidel. On-the-fly processing of generalized lumigraphs. *Computer Graphics Forum*, 20:C165–C173;C543, 2001.

[25] S. Seitz and C. Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2):151–173, 1999.

[26] A. Smolic and R. Yamashita. Draft requirements for 3DAV. Technical report, ISO/IEC JTC1/SC29/WG11, May 2002. MPEG Doc. N4795, Fairfax, Canada.

[27] C. Theobalt, M. Magnor, P. Schüler, and H.-P. Seidel. Combining 2d feature tracking and volume reconstruction for online video-based human motion capture. *Proc. IEEE Pacific Graphics 2002,* Beijing, China, pages 96–103, Oct. 2002.

[28] B. Wilburn, M. Smulski, K. Lee, and M. Horowitz. The light field video camera. *SPIE Proc. Media Processors 2002*, 4674, Jan. 2002.

[29] D. Wood, D. Azuma, K. Aldinger, B. Curless, T. Duchamp, D. Salesin, and W. Stuetzle. Surface light fields for 3D photography. *Proc. ACM Conference on Computer Graphics (SIGGRAPH-2000),* New Orleans, USA, pages 287–296, July 2000.

[30] S. Wuermlin, E. Lamboray, O. Staadt, and M. Gross. 3D video recorder. *Proc. IEEE Pacific Graphics 2002,* Beijing, China, pages 325–334, Oct. 2002.

[31] J. Yang, M. Everett, C. Buehler, and L. McMillan. A real-time distributed light field camera. *Proc. ACM 13th Eurographics Workshop on Rendering,* Pisa, Italy, pages 77–86, June 2002.

[32] Zaxel Inc. Los Altos Hills, CA USA. http://www.zaxel.com.

Figure 11. The visual hull is represented by the polyhedron resulting when intersecting the back-projected silhouette polygon cones. The surface can be arbitrarily shaded, or the video images can be applied as view-dependent textures by projective texture mapping in conjunction with shadow mapping to avoid through-projection artifacts.

Figure 12. A volumetric model of the dynamic visual hull is reconstructed on-line. The voxel model is view-dependently textured by rendering each voxel as a billboard facet.

Figure 13. Different views rendered by 3D warping from three image pairs using visual hull-enhanced dense depth maps.