

# Real-time Volume Rendering for High Quality Visualization in Augmented Reality

Oliver Kutter<sup>1</sup>, André Aichert<sup>1</sup>, Christoph Bichlmeier<sup>1</sup>, Jörg Traub<sup>1</sup>, Sandro Michael Heining<sup>2</sup>, Ben Ockert<sup>2</sup>, Ekkehard Euler<sup>2</sup>, and Nassir Navab<sup>1</sup>

<sup>1</sup> Chair for Computer Aided Medical Procedures (CAMP)  
Technische Universität München, Germany

{kutter, aichert, bichlmei, traub, navab}@cs.tum.edu

<sup>2</sup> Ludwig-Maximilians-Universität München, Chirurgische Klinik und Poliklinik - Innenstadt AG  
CAS, Germany

{Sandro-Michael.Heining, Ben.Ockert, Ekkehard.Euler}@med.uni-muenchen.de

**Abstract** Recent advances in the field of computer graphics have enabled extremely fast and high quality rendering of volumetric data. However, these algorithms have been developed and optimized for visualization on single view displays, and not for stereoscopic augmented reality systems. In this paper, we present our implementation and results for the integration of a high quality hardware accelerated volume renderer into a medical augmented reality framework using a video see through HMD. The performance and quality of the renderer is evaluated through phantom experiments and an in-vivo experiment. Compared to the literature, our approach allows direct real-time stereo visualization of volumetric medical data on a HMD without prior time consuming pre-processing or segmentation. To further improve the visual perception and interaction of real and virtual objects, the renderer implements occlusion handling with the physicians hands and tracked medical instruments.

## 1 Introduction

Real-time visualization of 2D and 3D image data is one of the key tasks of any augmented reality system. Besides rendering the image data in real-time, the visualization has to provide the user with the appropriate visual cues at the right time. Augmented Reality was introduced as an alternative to monitor based visualization for the presentation of medical image data during surgery. Exemplary systems were proposed within neurosurgery [16], orthopaedic surgery [21], and liver surgery [9]. Existing solutions for medical augmented reality often implement real-time rendering by reducing quality or using simple rendering techniques. Simple wireframe surface models or slice rendering is applied to achieve real-time performance. In addition many methods require time consuming pre-processing of the volume data and manual interaction. More sophisticated methods enhance 3D perception within their stereo systems i.e. by transparent surfaces [6], virtual mirrors [3], or viewing windows [1] into the body. Important visual cues, such as shape from shading, depth from occlusion or motion parallax can help the user of a medical augmented reality system by an improved perception of the 3D data. This paper summarizes efficient implementations for hardware accelerated direct volume rendering (DVR) and proposes smart extensions to get the best visual perception while ensuring the required real time update rate.

## 2 Related Work

The development of GPU<sup>3</sup> ray-casting over the past five years [14,19] has led to flexible and fast hardware accelerated direct volume renderers. The popularity of GPU ray-casting is thanks to its easy and straightforward implementation on nowadays GPUs as well as the great flexibility and adaptability of the core ray-casting algorithm to a specific problem set.

Several recent works deal with enhancing the visual and spatial perception of the anatomy from the 3D data set by advanced DVR, many based on GPU ray-casting. Krüger *et al.*[13] propose ClearView for segmentation free, real-time, focus and context (F+C) visualization of 3D data. ClearView draws the attention of the viewer to the focus zone, in which the important features are embedded. The transparency of the surrounding context layers is modulated based on context layer curvatures properties or distance to the focus layer. Viola *et al.*[23] presented importance driven rendering of volumes based on cut aways allowing fully automatic focus and context renderings. Burns *et al.*[5] extend importance driven volume rendering with contextual cutaway views and combine 2D ultrasound and 3D CT image data based on importance. Beyer *et al.*[2] present a very sophisticated volume renderer, based on GPU ray-casting, for planning in neurosurgery. The renderer supports concurrent visualization of multimodal three-dimensional image, functional and segmented image data, skull peeling, and skin and bone incision removal at interactive frame rates. Kratz *et al.*[12] present their results for the integration of a hardware accelerated volume renderer into a virtual reality environment for medical diagnosis and education. The presented methods and techniques are closely related to our work. However the renderer is integrated within a virtual reality environment. Thus, no fusion of real and virtual data is performed and important augmented reality problems like interaction of real and virtual objects are not addressed.

F+C rendering for medical augmented reality, has been proposed in recent work by Bichlmeier *et al.*[4]. However was never fully realized using the GPU with the same degree of quality and performance as in the field of computer graphics. The presented methods are limited to rendering polygonal surface data, extracted by time consuming segmentation from the volume data set. Although the relevant anatomy and structures can be depicted, the approach suffers from lack of real volumetric visualization of medical image data. Important and standard medical visualization modes are impossible to achieve with these technique, i.e. Maximum Intensity Projection (MIP) and volume rendering of soft tissue. Another limitation is that the visualization is static and fixed to the time of mesh generation. Despite these drawbacks, the proposed video ghosting technique of the skin surface is promising as it enables a natural integration of the focus zone into the real video images.

Our work deals with the integration of a high quality hardware accelerated volume renderer into an augmented reality framework. The current setup is aimed at medical augmented reality using a video see through HMD system. Compared to surface rendering, DVR allows the use of the whole volume information during rendering without prior pre-processing.

---

<sup>3</sup> GPU - Graphics Processing Unit

## 3 Methods

### 3.1 Medical Augmented Reality System

The CAMPAR [20] augmented reality framework takes care of temporal synchronization of the video and tracking data streams and provides basic functionality for the visualization on the HMD. The video see-through HMD system is made of two color cameras for each eye and an inside-out optical tracking system which was originally introduced as RAMP by Sauer et al. [18]. CAMPAR runs on a standard workstation class PC (Dual Intel<sup>TM</sup>Xeon, 3.2Ghz, 1.8Gb Ram, Nvidia<sup>TM</sup>Geforce 8800 Ultra, 768Mb on-board memory). On the same PC runs also a server process for the RAMP inside-out tracking.

Tracking of the objects in the scene is accomplished by two separate optical tracking systems. An outside-in optical tracking system consisting of four infrared (IR) ART-track2<sup>4</sup> cameras mounted to the ceiling of the room and an inside-out optical tracking system using an IR camera mounted directly on the HMD. Position and orientation of the users head are accurately determined by the inside-out tracking system, while the outside in tracking covers a much greater tracking volume and is used for all tracked objects in the scene. A reference system composed of a known arrangement of optical tracking markers, visible by both tracking systems serves as the common reference frame of the scene.

For correct placement of the CT volume data inside the augmented reality scene it is registered to the coordinate frame of a tracking system in the scene. We use 4[mm] Beekley<sup>5</sup> CT-spots fiducials, that are additionally coated with an infrared reflective foil and attached to the surface of the subject. Thereby the same markers are locatable by the optical tracking system and in CT image data. The CT image and tracking space position of the markers are used to compute the rigid transformation from CT image to tracking space as described in [22]. Skin markers are however only suited for rigid anatomy, i.e. in phantom studies. For clinical introduction the concept of dynamic reference base i.e. markers attached to the target (bone) anatomy has to be used. If no combined markers or no markers at all were attached to the subject during CT image acquisition, the data has to be registered by other means. For phantom experiments (cf. Section 4) we therefore attached fiducials to the surface of a phantom and acquired a high resolution CT scan as the original human CT dataset was acquired without CT fiducials. The centroids of the markers are extracted fully automatically from the data set based on intensity thresholding and moments analysis. Finally the real human CT and phantom CT datasets are registered by intensity based registration, using *Mutual Information* and *Best Neighbor* optimization (cf. [11] for a comprehensive overview on medical image registration).

### 3.2 High Quality Volume Rendering

The GPU accelerated ray-casting based volume renderer is implemented as part of a C++ library in OpenGL and OpenGL's built-in shading language (GLSL). Rendering is realized by multipass techniques for ray setup, volumetric ray-casting, and composition of the final image. Various rendering modes, i.e. Non-Polygonal Iso Surface, Volume

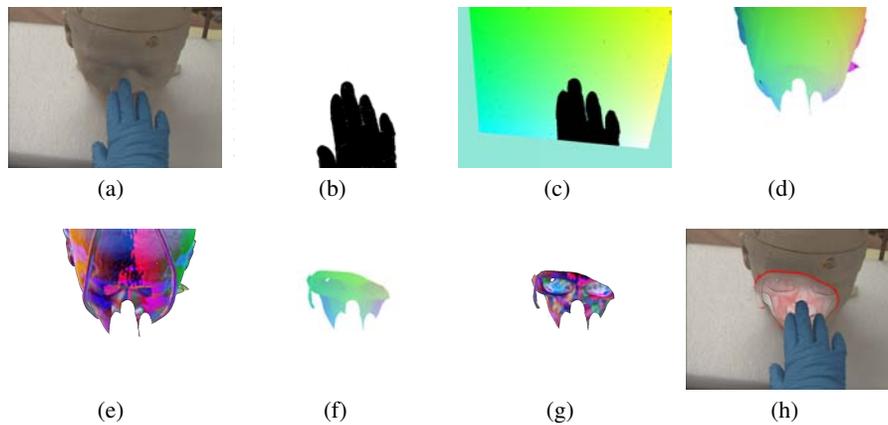
---

<sup>4</sup> A.R.T. GmbH - <http://www.ar-tracking.de/>

<sup>5</sup> Beekley Corp. - <http://www.beekley.com>

Rendering, Maximum Intensity Projection (MIP), Iso Surface Shaded Volume Rendering and Digitally Reconstructed Radiographs (DRR), of 3D medical data are supported.

We employ several techniques to render high quality images in real-time. Floating point render targets are used for full precision throughout the entire rendering pipeline. Methods for clipping of the volume and for correct interaction of the volume and traditional opaque scene geometry similar to those presented by Scharsach *et al.*[19] are implemented. Local illumination is supported by various illumination models and on the fly gradient computation on the GPU. Fast and high quality non-polygonal iso surface rendering is realized by first hit iso surface ray-casting and subsequent iso surface refinement. Higher quality transfer functions, pre-integrated [8] and post color attenuated [24], are employed to retain high image quality for low sampling rates. The performance of the renderer is optimized by empty space leaping of non visible voxels, using an oct-tree representation of the volume for culling of inactive voxels. Early ray termination in the ray-casting shaders, random ray start off-setting, and variable size of pipeline internal render targets furthermore contribute in achieving high performance and high quality rendering. For a detailed treatment of the rendering algorithms and techniques we refer to [7].



**Figure 1.** Stages of Focus and Context Rendering pipeline optimized for Augmented Reality with enabled Hand Occlusion Handling. (a) Video Input feed image from camera. (b) Depthbuffer for ray-casting after hand segmentation (c) Ray Entry Lookup texture (d,e) Skin Context Layer: (d) Surface, (e) Surface Normals. (f,g) Bone Focus Layer: (f) Surface,(g) Surface Normals. (h) Composition of Video Image, Focus and Context Layers.

In addition to the named standard volume rendering techniques the renderer implements ClearView [13] for pre-processing free, real-time F+C visualization (cf. Fig. 1(h)) of 3D data. The implementation is extended by the video ghosting techniques proposed by Bichlmeier *et al.*[4]. We furthermore add an optimization to the original ClearView pipeline greatly accelerating it for real-time stereo visualization. The original ClearView method generates focus and context layers each time the viewpoint changes without considering occlusion of inner layers by other closer to the camera located layers. Thereby, for a large number of pixels never contributing to the resulting image costly computations are executed. For an augmented reality environment, in which the viewpoint changes from frame to frame, this greatly reduces performance. We therefore incor-

porate focus region information already when generating the context and focus layers (cf. Fig 1(c) and 1(f)). Thereby, pixels outside the focus region are masked in an early stage of the pipeline and non-contributing computations are avoided. Especially more computationally expensive rendering techniques for inner layers benefit from this modifications. Additionally, the results of layer  $n - 1$  are used as inputs for layer  $n$ , when generating the temporary hit layer images.

### 3.3 Integration into AR environment

The requirements for volume rendering in medical augmented reality differ greatly from those for visualization on medical workstations, i.e. for diagnostic purposes. The rendering engine has to be optimized for speed first, followed by image quality second. The visualization should at no point result in large latencies of the overall system. In general this task is difficult to achieve as the performance depends on many parameters. Additionally, for in-situ visualization with a HMD as in our setup, the renderer has to be capable of perspectively correct stereo rendering in real-time (30fps). We realize stereo rendering, by rendering the volume for the left and right eye in two separate passes. In each pass the projection matrix for the current eye is loaded into the OpenGL projection matrix and the volume is rendered into a texture on the GPU. To achieve real-time stereo rendering performance we identified some basic optimization techniques that we applied when we adopted the volume renderer to the augmented reality environment.

A general guideline is to avoid computations that do not contribute to the final rendered image as early as possible in the pipeline. By making use of the occlusion of the volume by real and virtual objects further computations are skipped for many pixels of the image. The marking of pixels inside the focus zone for F+C rendering is motivated by the same considerations. Computationally intensive calculations should be deferred in the shaders as long as possible or skipped using conditional statements. A small memory footprint on the GPU is another important factor for high performance rendering. As much memory, textures, render and vertex buffers, should be shared or reused among different rendering techniques and for the per eye render passes. By keeping the memory usage to a minimum all data can be stored in high performance GPU memory. The image quality is another factor that has to be considered for the AR environment. Higher quality rendering techniques are computationally more costly resulting in a reduced framerate. The question is to which level of quality differences can be perceived by the user of the system on the HMD. Naturally, it makes no sense to render high resolution images if the same subjective user perception can be achieved with slightly reduced image resolution, resulting in a higher rendering performance.

Incorrect occlusion of virtual and real objects in the augmented scene is one of the fundamental problems in augmented reality. Without occlusion handling, virtual objects will appear floating over real objects and any impression of depth will be lost. In this work we present a basic yet effective method for occlusion handling in medical augmented reality using a video see through HMD. The operating site visible to the physician is relatively small and access to it is restricted to the treating physician, therefore the main and most disturbing occluders are the physician's own hands and instruments. In Sandor *et al.*[17] the advantages of hand occlusion handling for video see through HMDs were demonstrated. In our approach we handle the occlusion the physician hands by color filtering of the video images. A problem for the stable separation of a physician's hands from the rest of the video image is the large similarity of the standard gloves

white surface color and other features in the operating site image, i.e. patient skin. Therefore we exchange the standard white gloves by blue gloves, which are less common but available in the clinics. The color filter is implemented as a shader that compares color hue to a reference color where value and brightness are not too low for hue to be reliable. Because of natural shading, it is not sufficient to compare RGB color values. Brightness may vary largely, so conversion to HSV is worth the additional effort and gives much better results. The filter is applied at the very first stage of our rendering pipeline. Its result is written to a depth render target, where every pixel not belonging to one of the hands is set to the minimal depth value. The resulting depth buffer image (cf. Fig. 1(b)) is used for all following pipeline stages. Thereby, costly operations for pixels occluded by the hands are avoided by the Depth-Test of the GPU.

Fischer *et al.*[10], presented a method for instrument occlusion handling for medical augmented reality based on offline segmentation and first hit ray-casting. The integration and extension of their approach to real-time occlusion handling is straight forward within our method. It is implemented by running a depth test in a fragment shader, comparing the depth of the first hit isosurface, i.e. skin or bone, extracted by GPU ray-casting and the depth buffer values from the rendering of a polygonal model of a tracked instrument. For every pixel of the rendered instrument model with a smaller depth value than the skin iso-surface the pixel is set to black in the virtual instrument image. Thereby, after composition of all render pass results, the virtual instrument is only visible inside the patient behind the skin surface.

## 4 Experiments and Results

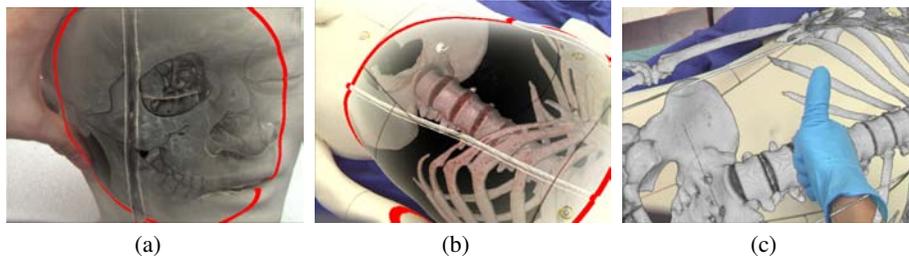
We conducted two experiments for human CT datasets on a realistic phantom and two in-vivo experiments to evaluate the performance of the renderer in a realistic scenario. The performance of the renderer was evaluated in detail for multiple rendering modes on the phantom. For the performance measurements the internal render target resolution was set to the camera resolution (640x480), the viewport resolution was set to the HMD resolution (1024x768). In all experiments empty space leaping and early ray termination optimizations were activated. For the evaluation a volunteer wore the HMD and inspected the phantom by continuous motion around it for two minutes. The minimum, maximum and, average framerate was measured using Fraps<sup>6</sup>, a freely available benchmarking tool. A general note on the framerate before the detailed results are presented. The maximum framerate of the system is limited by the update rate of the slowest component in the system. For the used system the video cameras limit the overall system framerate. Thus, the average framerate can never be more than 30 frames per second, the camera refresh rate, in our experiments.

**Phantom Experiments** Setting up repeatable and realistic experiments for medical augmented reality is a tedious and complex task. A main factor is the availability of a realistic phantom. Within our group a phantom based on the data of the *Visible Korean Human Project* [15]<sup>7</sup> (VKHP) was manufactured<sup>8</sup>. The life-size phantom ranges from the head to the lower hips. The performance of the renderer was tested during several experiments with three datasets, head (296x320x420x8bit), thorax high res.

<sup>6</sup> <http://www.fraps.com/>

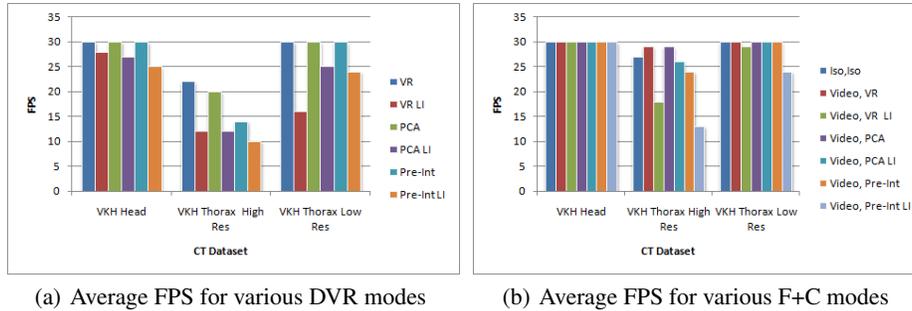
<sup>7</sup> <http://vkh3.kisti.re.kr/new/>

<sup>8</sup> FIT - Fruth Innovative Technologien GmbH, [www.pro-fit.de](http://www.pro-fit.de)



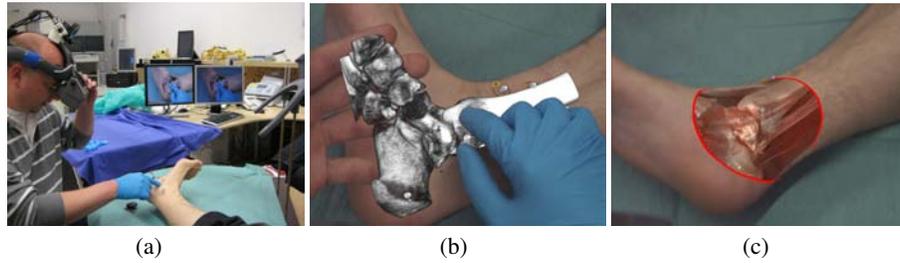
**Figure 2.** (a) Focus and Context rendering of the Korean head dataset. Video image is used as context layer, Focus layer is rendered with volume rendering. (b,c) Renderings of the Korean thorax dataset. (b) Focus and Context rendering with shaded volume rendering for the focus layer (bone). (c) Standard direct volume rendering visualizing the skeleton.

(256x368x522x8bit) and thorax low res. (256x368x261x8bit), extracted from the original *Visible Korean Human* (VKH) CT data. Exemplary result images for the head and thorax datasets of the visible korean human phantom are shown in Fig. 2 (a-c). The measured average framerate for various DVR rendering modes is depicted in Fig 3(a) and Fig. 3(b) for F+C rendering. The better performance of the F+C rendering modes compared to the DVR rendering modes can be explained by the reduced number of pixels for which rays are cast through the dataset when the focus region optimization is enabled.



**Figure 3.** Performance results for VKH datasets measured in average frames per second. (a) Average frames per second for various DVR modes. (b) Average frames per seconds for F+C Rendering. VR - Volume Rendering (VR) with standard post-classification, PCA - VR with Post Color Attenuation, Pre-Int - VR with Pre-Integration. LI - Local Illumination via Blinn-Phong and on-the-fly gradients by central differences. Iso - Skin Surface is rendered as shaded iso surface. Video - Ghosting of skin surface in video enabled.

**In Vivo Experiment** One of our colleagues, who had a sport accident, positioned CT markers at his right foot, using locations that could be identified later on for tracking markers, before a CT scan of the foot was acquired. By attaching tracking fiducials at these distinct landmarks the CT dataset with the CT fiducials at the same relative positions can be registered. In the experiment we evaluated the performance and quality of the rendered image for a volume (256x256x154 voxels), and as well the hand occlusion. Figure 4 depicts some of the results we got for the foot dataset with shaded DVR and enabled F+C rendering.



**Figure 4.** (a) Trauma surgeon inspecting the foot through the HMD. (b) Volume Rendering of bone, with visible Hand Occlusion problem. Hand without glove is beneath rendering. Hand with glove is placed correctly in front. (c) Focus and Context Rendering using shaded volume rendering for focus layer.

For the selected parameters the average framerate was 30fps. The frame rate never dropped below 28 fps, neither for standard VR nor for F+C rendering. We also evaluated the performance for rendering with the full HMD resolution (1024x768). Using this resolution, we noticed a slight performance drop of the average framerate for standard VR. For F+C rendering the framerate remained stable about 30fps. The performance drop for VR is mainly due to the full visibility of the dataset compared to F+C rendering, in which only the volume inside the focus zone is rendered. In the experiment the renderer was also tested by two members of the surgical staff of the trauma surgery department. After the experiments they provided us with valuable feedback considering performance, quality and possible future improvements. Especially the stable real-time performance and high quality of the renderer for several different anatomies, head, torso and the successful in-vivo experiment on the foot motivate further studies from the side of our clinical partners.

## 5 Discussion

Introduction of techniques for hardware accelerated volume rendering for medical augmented reality is an important step towards the integration of the complete 3D data into the medical augmented reality environment in real-time. The positive feedback of the physicians, motivates us for further investigation of the visualization and usability studies with multiple physicians.

The ultimate goal is to provide the best visualization of the 3D image data to the physicians at any point during a medical procedure. However, functionality comes before fancy visualization. Thus, determining the best rendering mode for each medical application is another important issue and interesting subject of research. Future work should focus on evaluation of the already existing rendering modes for a defined set of medical procedures together with our clinical partners. The specific requirements have to be analyzed for each procedure over the complete workflow, from pre-operative imaging till the end of the intervention. For this purpose we plan to record video and tracking data for exemplary procedures for use as offline training datasets. For these dataset the visualization modes can be interactively changed and adjusted based on feedback from our medical partners. Thereby, the visualization can be iteratively refined and optimized for the specific medical procedure.

Concerning the currently implemented occlusion handling, we do not claim that the method is suited for augmentation of all medical procedures, especially not for traditional open surgery. Simple color segmentation is not enough for this task. However, for pre-operative inspection of the patient using augmented reality techniques through the HMD or as mentioned for minimally invasive procedures, with access restricted by ports into the patient, the approach is sufficient. The integration of more elaborate methods for real-time occlusion handling, i.e. depth map reconstruction using the stereo camera system, could be seen as future work.

## 6 Conclusion

This paper presents and summarizes the results for the integration of a high quality hardware accelerated volume renderer into an augmented reality environment. The pipeline of the renderer and advanced rendering techniques were adapted and optimized for stereoscopic rendering on a HMD. The occlusion problem of real and virtual objects in augmented reality scenes was addressed in this work, by simple yet effective methods for diagnosis and minimally invasive procedures using a HMD. The method allows an improved, more natural perception of the augmented reality scene. The performance of the presented techniques has been demonstrated and evaluated in several experiments by rendering real human CT data of the skull and the complete thorax on realistic anatomical phantoms using different rendering techniques. In an in-vivo experiment the renderer was used to augment a human foot by its pre-acquired CT dataset and evaluated by three trauma surgeons.

## References

1. M. Bajura, H. Fuchs, and R. Ohbuchi. Merging virtual objects with the real world: seeing ultrasound imagery within the patient. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 203–210. ACM Press, 1992.
2. J. Beyer, M. Hadwiger, S. Wolfsberger, and K. Bühler. High-Quality Multimodal Volume Rendering for Preoperative Planning of Neurosurgical Interventions. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, pages 1696–1703, 2007.
3. C. Bichlmeier, M. Rustae, and S. H. nad N. Navab. Virtually extended surgical drilling device: Virtual mirror for navigated spine surgery. In *Proc. Int'l Conf. Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2007.
4. C. Bichlmeier, F. Wimmer, H. Sandro Michael, and N. Nassir. Contextual Anatomic Mimesis: Hybrid In-Situ Visualization Method for Improving Multi-Sensory Depth Perception in Medical Augmented Reality. In *Proceedings of the 6th International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 129–138, Nov. 2007.
5. M. Burns, M. Haidacher, W. Wein, I. Viola, and E. Groeller. Feature emphasis and contextual cutaways for multimodal medical visualization. In *EuroVis 2007 Proceedings*, May 2007.
6. P. J. Edwards, L. G. Johnson, D. J. Hawkes, M. R. Fenlon, A. J. Strong, and M. J. Gleeson. Clinical experience and perception in stereo augmented reality surgical navigation. In *Proceedings of Medical Imaging and Augmented Reality: Second International Workshop, MIAR 2004, Beijing, China, August 19-20, 2004.*, pages 369–376, 2004.
7. K. Engel, M. Hadwiger, J. Kniss, and C. Rezk-Salama. *Real-Time Volume Graphics*. AK Peters, Ltd., 2006.
8. K. Engel, M. Kraus, and T. Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware*, pages 9–16, 2001.

9. M. Feuerstein, T. Mussack, S. M. Heining, and N. Navab. Intraoperative laparoscope augmentation for port placement and resection planning in minimally invasive liver resection. *IEEE Trans. Med. Imag.*, 27(3):355–369, March 2008.
10. J. Fischer, D. Bartz, and W. Straßer. Occlusion handling for medical augmented reality using a volumetric phantom model. *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 174–177, 2004.
11. J. Hajnal, D. Hawkes, and D. Hill. *Medical Image Registration*. CRC Press, 2001.
12. A. Kratz, M. Hadwiger, A. Fuhrmann, R. Splechtina, and K. Buhler. GPU-Based High-Quality Volume Rendering For Virtual Environments. In *AMIARCS*, Copenhagen, Denmark, Oct. 2006. MICCAI Society.
13. J. Krüger, J. Schneider, and R. Westermann. ClearView: An interactive context preserving hotspot visualization technique. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization / Information Visualization 2006)*, 12(5), September-October 2006.
14. J. Krüger and R. Westermann. Acceleration Techniques for GPU-based Volume Rendering. *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, 2003.
15. J. Park, M. Chung, S. Hwang, Y. Lee, D. Har, and H. Park. Visible korean human: Improved serially sectioned images of the entire body. 24(3):352–360, March 2005.
16. P. Paul, O. Fleig, and P. Jannin. Augmented virtuality based on stereoscopic reconstruction in multimodal image-guided neurosurgery: Methods and performance evaluation. *IEEE Trans. Med. Imag.*, 24(11):1500–1511, 2005.
17. C. Sandor, S. Uchiyama, and H. Yamamoto. Visuo-Haptic Systems: Half-Mirrors Considered Harmful. *EuroHaptics Conference, 2007 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics 2007. Second Joint*, pages 292–297, 2007.
18. F. Sauer, A. Khamene, B. Bascle, and G. J. Rubino. A head-mounted display system for augmented reality image guidance: Towards clinical evaluation for imri-guided neurosurgery. In *Proc. Int'l Conf. Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 707–716, London, UK, 2001. Springer-Verlag.
19. H. Scharasch, M. Hadwiger, A. Neubauer, S. Wolfsberger, and K. Buhler. Perspective Iso-surface and Direct Volume Rendering for Virtual Endoscopy Applications. *Proceedings of Eurovis/IEEE-VGTC Symposium on Visualization 2006*, pages 315–322, 2006.
20. T. Sielhorst, M. Feuerstein, J. Traub, O. Kutter, and N. Navab. Campar: A software framework guaranteeing quality for medical augmented reality. *International Journal of Computer Assisted Radiology and Surgery*, 1(Supplement 1):29–30, June 2006.
21. J. Traub, P. Stefan, S.-M. M. Heining, T. Sielhorst, C. Riquarts, E. Euler, and N. Navab. Towards a hybrid navigation interface: Comparison of a slice based navigation system with in-situ visualization. In *Proceedings of MIAR 2006*, LNCS, pages 364–372, Shanghai, China, Aug. 2006. Springer.
22. J. Traub, P. Stefan, S.-M. M. Heining, C. R. Tobias Sielhorst, E. Euler, and N. Navab. Hybrid navigation interface for orthopedic and trauma surgery. In *Proceedings of MICCAI 2006*, LNCS, pages 373–380, Copenhagen, Denmark, Oct. 2006. MICCAI Society, Springer.
23. I. Viola, A. Kanitsar, and M. Groller. Importance-driven volume rendering. *Proceedings of IEEE Visualization*, 4:139–145, 2004.
24. Q. Zhang, R. Eagleson, and T. Peters. Rapid Voxel Classification Methodology for Interactive 3D Medical Image Visualization. In *MICCAI 2007 Proceedings*, Lecture Notes in Computer Science. Springer, Oct. 2007.